Chapter 1

Introduction

1.1 What is Optimal Control about?

The goal of Optimal Control is to *optimize the operation of a dynamic system* with respect to given optimization criteria (e.g. minimizing costs/maximizing return).
Consider a dynamic system as shown in Figure 1.1:

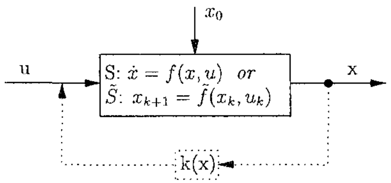


Figure 1.1: Dynamic system (continuous/discrete), optional with feedback $k(x)$

We want to determine the input signal as a function of time (or the feedback $u = k(x)$ respectively)

$$u(t) : 0 \leq t \leq t_f \quad (\text{or } u = k(x))$$

such that

- the given (physical) constraints are met and that
- a performance objective $J(x(\cdot), u(\cdot))$ is minimized.

In general, an optimal control problem consists of three key elements:

1. a mathematical model of the system,
2. the constraints on inputs and states that must be taken into account and
3. an objective/performance index to be minimized.

The following simple example may clarify the concept:

Example 1.1: *Rocket car*

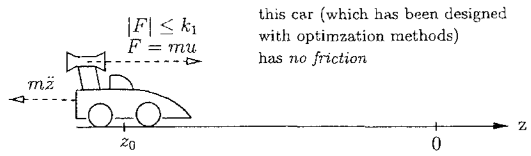


Figure 1.2: Rocket car

- *objective:* the car starting in z_0 shall reach the origin $z = 0$ with minimal "fuel" expense in the fixed time t_f (no friction is considered)
- *mathematical formulation:*

$$\min_{u(\cdot)} \underbrace{\int_0^{t_f} |u(\tau)|\, d\tau}_{\text{performance index J}} \tag{1.1}$$

$$\text{s.t.} \quad \left.\begin{array}{lll} \dot{x}_1 = x_2, & x_1(0) = z_0, & x_1(t_f) = 0 \\ \dot{x}_2 = u, & x_2(0) = 0, & x_2(t_f) = 0 \end{array}\right\} \textit{mathematical model of the system} \qquad \left.\begin{array}{l} \\ |u| \leq k_1 m \end{array}\right\} \textit{constraints}$$

Note that

- the input u is a function of time (not a single value): $\quad u : [0, t_f] \to [-k_1 m, k_1 m]$,
- the performance index $\quad J = J(u(\cdot), x(\cdot)) = \int_0^{t_f} |u(\tau)|\, d\tau$
  is a functional of $u(\cdot)$ (and in general of $x(\cdot)$),
- the solution of the minimization problem is obtained for a pair of optimal control and state trajectories (u^*, x^*) : $\quad J^* = \min_{u(\cdot)} J(u, x) = J(u^*(\cdot), x^*(\cdot))$.

- *in general the following questions arise:*
  - Does a minimum exist and is obtained (*existence*)?
  - Is the minimum *unique*?
  - How can the input as a function of time $u(\cdot)$ be obtained?
  - Could we also obtain a feedback $u = k(x)$?

Answers on these questions will be considered throughout this course ...

1.2 Difference to static/parameter optimization

One might ask what is the difference of an optimal control problem and static optimization or an optimal control problem and the parameter optimization in control.

- In *static optimization* one actually is asking to find a solution for the following problem:

$$\begin{aligned} &\min_q F(q) \qquad (q \in \mathbb{R}^n) \\ &s.t. \quad h(q) \geq 0 \end{aligned} \tag{1.2}$$

Note that

- we do *not* have a dynamic system here,
- we do *not* search for a whole trajectory (in time) but only for an optimal point: $q^* \neq q^*(t)$ and therefore
- the performance index F is *no* functional any more but a conventional function (objective function).

Example 1.2: *Static optimization*
We are looking for the minimum of a quadratic function subject to an inequality constraint:

$$\begin{aligned} &\min_q \; q^2 + 2q + 1 \qquad (q \in \mathbb{R}) \\ &s.t. \quad q \geq -3/2 \end{aligned} \tag{1.3}$$

Drawing the graph of the cost function and the constraint (see Figure 1.3) one can immediately read off the solution $q^* = -1$.

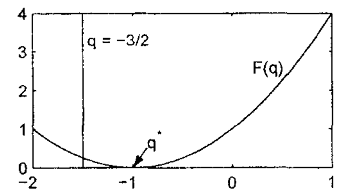


Figure 1.3: Graph of the cost function $F(q) = q^2 + 2q + 1$ and solution q^*

- *Parameter optimization* is a typical problem in control theory dealing with the evaluation of optimal controller parameters. The objective is to minimize the error e in the closed loop (here considered over an infinite time horizon), cmp. Figure 1.4.

  Assume we have a given controller "structure":

$$G_k(s) = \frac{U(s)}{E(s)} = k\frac{1+Ts}{s} \tag{1.4}$$

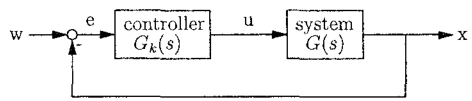


Figure 1.4: Closed loop

and a model of the system $G(s)$ is also given, then we can derive the error e in dependency of the controller parameters k and T: $e(t) = e(t, k, T, w)$. Fixing the input $w(t)$ on a specified test signal (e.g. a unit step), the optimal parameters k^*, T^* are obtained as arguments by solving the following minimization problem:

$$\min_{k,T} \underbrace{\int_0^\infty e(t)dt}_{F(k,T)} \tag{1.5}$$

Note that

- this is pure *static* optimization in k and T ...
- as the "structure" of the controller is predetermined and thus
- the objective F is a *function* of the parameters k and T.

1.3 More examples

Example 1.3: *Rocket car no. 2*
We consider the example above (Example 1.1) with a different objective, namely to bring the car to the origin in *minimal time*. The only thing that is changing is the performance index:

$$J = \int_0^{t_f} d\tau = t_f \tag{1.6}$$

Note that J is now independent of the input (fuel expense) and the final time t_f is free.
The resulting control is shown in Figure 1.5 (we will derive this later in the lecture).

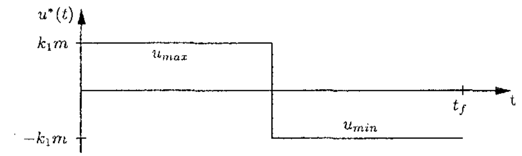


Figure 1.5: Optimal input $u^*(t)$ for the rocket car no. 2, minimizing the final time t_f

Example 1.4: *Rocket car no. 3*
One more variation of the rocket car example: evaluate the input yielding the least square costs.

$$\min_{u(\cdot)} J(u(\cdot), x(\cdot)) = \min_{u(\cdot)} \underbrace{\int_0^{t_f} (x^T Q x + u^2)\, d\tau}_{\text{finite time LQR (R = I)}} \tag{1.7}$$

$$\begin{aligned} s.t. \quad \dot{x}_1 &= x_2, \qquad & x_1(0) &= z_0 \\ \dot{x}_2 &= u, & x_2(0) &= 0 \end{aligned}$$

Note that
- the end state $x(t_f)$ and the input u are free, but
- J now also depends on the state trajectory $x(\cdot)$
- if we let $t_f \to \infty$ we have the well known LQR-controller

Example 1.5: *Managing spending and savings*
The problem is illustrated in Figure 1.6:

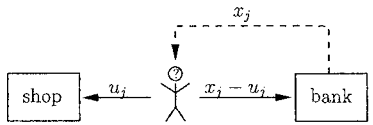


Figure 1.6: Managing spending and savings

Let x_j be the amount of money you possess in year j and which is split in spendings u_j and savings $(x_j - u_j)$ at the beginning of every year. For the saved money you receive some interest from the bank depending on the interest rate p, thus in the following year the amount of money you can decide on is:

$$x_{j+1} = (x_j - u_j) + p(x_j - u_j) \tag{1.8}$$

The *objective* is to maximize the consumption over n years, i.e. the n discrete input values u_j $(j = 1, ..., n)$ labeled with $\{u_n\}$ shall be chosen such that $\sum u_j$ is maximized:

$$\max_{\{u_n\}} J(\{u_n\}, \{x_n\}) = \max_{\{u_n\}} \sum_{j=1}^{n} u_j \tag{1.9}$$

while considering the *constraint* that the spendings can never exceed the available amount of money:

$$u_j \leq x_j. \tag{1.10}$$

Note that
- this is a discrete time problem
- the end time is fixed
- x_{n-1} is free
- in general a maximization problem can be converted into a minimization problem:

$$\max J = \min(-J) \tag{1.11}$$

1.4 Optimal Control - an overview

1.4.1 General mathematical setup

$$\begin{aligned} &\min_{u(\cdot)} J(u(\cdot), x(\cdot)) \\ &s.t. \quad \dot{x} = f(x,u) \qquad x(0) \in \mathcal{S}_0 \subset \mathbb{R}^n \\ &\qquad\qquad\qquad\qquad\quad\; x(t_f) \in \mathcal{S}_f \subset \mathbb{R}^n \\ &\qquad u(\cdot) \in \mathcal{U} \subset \mathbb{R}^m \\ &\qquad x(\cdot) \in \mathcal{X} \subseteq \mathbb{R}^n \end{aligned} \tag{1.12}$$

$$\text{and} \quad J(u(\cdot), x(\cdot)) = E(x(t_f)) + \int_0^{t_f} F(x,u)\, d\tau \tag{1.13}$$

1.4.2 Advantages and disadvantages

+ systematic approach (model, objective, constraint)
+ often an unique solution to the optimization problem is obtained
- an analytic solution is often difficult to derive
++ constraints can be considered
+ nature behaves optimal

1.4.3 Application areas

- economics
- aeronautics
- robotics
- biomathematics

Chapter 2

Review on static optimization

The goal of static optimization is to find a minimum (maximum) of a *scalar objective function* $F(u)$ with respect to $u = [u_1, \ldots, u_n]^T$ subject to constraints, which are defining the feasible region $\mathcal{U}$:

$$\min_{u \in \mathcal{U}} F(u) \tag{2.1}$$

or equivalent:

$$\min_{u \in \mathbb{R}^n} F(u) \quad s.t. \quad H(u) \leq 0, \quad G(u) = 0 \tag{2.2}$$

Notation:

solution, minimum cost: $F^* = \min F(u) = F(u^*)$
corresponding (optimal) input: $u^* = \arg\min F(u)$

Example 2.1: *Ball and spring*
We want to find the position at rest of a ball connected to a spring (stationary point) as shown in Figure 2.1. This is equivalent to finding the (feasible) point of minimal potential energy.

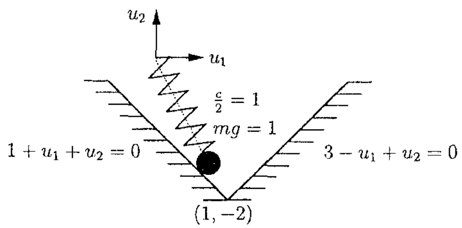


Figure 2.1: Setup of the ball and spring problem

$$\min_{u \in \mathbb{R}^2} \underbrace{u_1^2 + u_2^2}_{spring} + \underbrace{u_2}_{gravity} \tag{2.3}$$
$$\left.\begin{array}{l} s.t. \quad -1 - u_1 - u_2 \leq 0 \\ \qquad -3 + u_1 - u_2 \leq 0 \end{array}\right\} H(u) \leq 0$$

Example 2.2: *Feasibility problem*
Find a *feasible* point, i.e. a point satisfying a given set of constraints.

$$\begin{array}{l} G(u) = 0 \\ H(u) \leq 0 \end{array} \quad \Leftrightarrow \quad \begin{array}{l} \min \tilde{G}(u) \\ s.t. \quad H(u) \leq 0 \end{array} \quad \text{with } \tilde{G}(u) := \begin{cases} 0 & \text{if } G(u) = 0 \\ \infty & \text{otherwise} \end{cases} \tag{2.4}$$

2.1 Local/global minima, convexity

To find a *global* minimum of a function is in general, if there are no restrictions or further informations on the type of function, very difficult. Consider e.g. the function $F(u)$ shown in Figure 2.2:

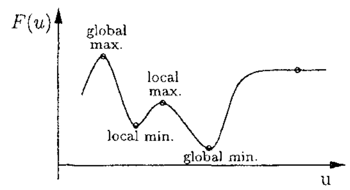


Figure 2.2: Extremal points of an arbitrary function $F(u)$

As it is that hard to make statements about global extremal values for the general case it makes sense to consider a special group of problems:

Theorem 2.1 (Convex problems) *If the performance index $F(u)$ is convex and the corresponding feasible set $\mathcal{U}$ is also convex the optimization problem is called a convex problem*
$\Rightarrow$ *$F(u)$ has one minimum value*
$\Rightarrow$ *every local minimum is also a global minimum.*

Definition 2.1 (Convexity of functions) *A function $F(u): \mathbb{R}^n \to \mathbb{R}$ is convex if*

$$\textit{for all } u^1, u^2 \in \mathbb{R}^n,\ \mu \in [0,1]: \qquad F(\mu u^1 + (1-\mu)u^2) \leq \mu F(u^1) + (1-\mu)F(u^2).$$

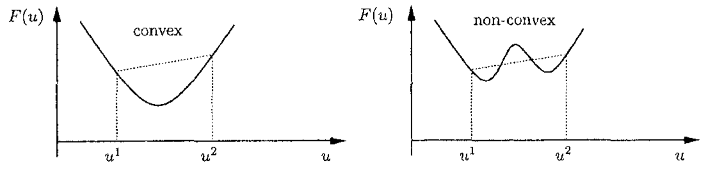


Figure 2.3: Convex and non-convex functions

"i.e. every point on the curve of F is below the connection line between (any) u^1 and u^2" (cmp. Figure 2.3).

Note also:

- F is convex $\Rightarrow \alpha F$ is convex
- F_1, F_2 are convex $\Rightarrow \alpha F_1 + \beta F_2$ is convex
- $F(u_1, u_2)$ is convex $\Rightarrow \inf_{u_2} F(u_1, u_2)$ is convex
- sums and integrals: $F(u_1, u_2)$ is convex in $u_1 \Rightarrow \int F(u_1, u_2)\, du_2$ is convex

Definition 2.2 (Convexity of sets) *A set $\mathcal{U} \subseteq \mathbb{R}^n$ is a convex set if*

$$u^1, u^2 \in \mathcal{U},\ \mu \in [0,1] \quad \Rightarrow \quad \mu u^1 + (1-\mu)u^2 \in \mathcal{U}$$

"i.e. points on all connection lines are in $\mathcal{U}$", see Figure 2.4.

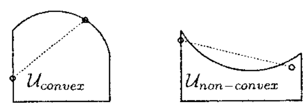


Figure 2.4: Convex and non convex sets

Example 2.3: *Convex set*
If $G(u)$ is *linear* and $H(u)$ is *convex* then the set given by $G(u) = 0$, $H(u) \leq 0$ is convex.

Example 2.4: *Convex minimization problem*
Discrete time LQR:

$$\begin{aligned} \min_u u^T Qu + Ru + V \qquad (Q > 0) \quad s.t. \quad & Gu = 0 \\ & Hu - K \leq 0 \quad (\Leftrightarrow |u_i| \leq \alpha) \end{aligned} \tag{2.5}$$

As $Q > 0$ the problem is *strictly convex* and therefore has a *unique* optimal solution.

We now introduce some terms on hyperplanes which are interrelated to convex sets:

Definition 2.3 (Seperating hyperplane) *If the sets $\mathcal{S}, \mathcal{T} \subseteq \mathbb{R}^n$ are convex and disjoint (i.e. $\mathcal{S} \cap \mathcal{T} = \emptyset$) then $\exists\, a \neq 0, b$ s.t.*

$$\begin{aligned} a^T x \geq b \quad \forall x \in \mathcal{S} \\ a^T x \leq b \quad \forall x \in \mathcal{T} \end{aligned}$$

i.e. the hyperplane $\{x \,|\, a^T x - b = 0\}$ seperates $\mathcal{S}$ and $\mathcal{T}$.

Definition 2.4 (Supporting hyperplane) *The hyperplane $\{x \,|\, a^T x = a^T x_0\}$ supports the set $\mathcal{S}$ at $x_0 \in \partial\mathcal{S}$ if*

$$x \in \mathcal{S} \quad \Rightarrow \quad a^T x \leq a^T x_0$$

(i.e. the halfspace $\{x \,|\, a^T x \leq b\}$ contains $\mathcal{S}$ if $b = a^T x_0$, but not for smaller values of b).

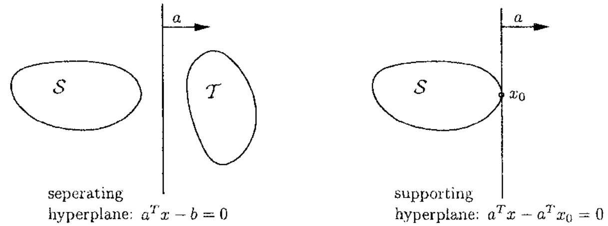


Figure 2.5: Seperating and supporting hyperplanes

Theorem 2.2 (Convex sets - supporting hyperplanes) *If a set $\mathcal{S}$ is convex then there exists a supporting hyperplane for each $x_0 \in \partial\mathcal{S}$.*

2.2 Notation

In the following let $F : \mathbb{R}^n \to \mathbb{R}$, $G : \mathbb{R}^n \to \mathbb{R}^m$. Furthermore, assume that F, J, G, H are sufficiently often continuously differentiable.

- "Gradient":

$$F_u(u) = \frac{\partial F}{\partial u}(u) = \underbrace{\nabla_u F(u)}_{\text{gradient of F}}{}^T = \left[\frac{\partial F}{\partial u_1}, \dots, \frac{\partial F}{\partial u_n}\right] \tag{2.6}$$

- "Jacobian":

$$G_u(u) = \frac{\partial G}{\partial u}(u) = \nabla_u G(u)^T = \begin{bmatrix} \frac{\partial G_1}{\partial u_1} & \cdots & \frac{\partial G_1}{\partial u_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial G_m}{\partial u_1} & \cdots & \frac{\partial G_m}{\partial u_n} \end{bmatrix} \quad \text{for } n = m\text{: "Jacobi matrix"} \tag{2.7}$$

- "Hessian":

$$F_{uu}(u) = \frac{\partial^2 F}{\partial u^2}(u) = \nabla_u^2 F(u)^T = \begin{bmatrix} \frac{\partial^2 F}{\partial u_1^2} & \cdots & \frac{\partial^2 F}{\partial u_n \partial u_1} \\ \frac{\partial^2 F}{\partial u_1 \partial u_n} & \cdots & \frac{\partial^2 F}{\partial u_n^2} \end{bmatrix} = \nabla_u^2 F(u) \tag{2.8}$$

- Taylor-expansion of F around u^0:

$$F(u) = F(u^0) + F_u(u^0)\,\Delta u + \frac{1}{2}\Delta u^T F_{uu}(u^0)\,\Delta u + \mathcal{O}^3(\|\Delta u\|), \qquad \Delta u := u - u^0 \tag{2.9}$$

- Euclidean norm on $\mathbb{R}^n$: $$\|x\| = \sqrt{\sum_{i=1}^{n} x_i^2} \tag{2.10}$$

2.3 Unconstrained static optimization

In unconstrained static optimization we consider the problem:

$$\min_{u \in \mathbb{R}^n} F(u) \tag{2.11}$$

2.3.1 Necessary and sufficient conditions

Theorem 2.3 (Necessary conditions for a local minimum)

first order condition: $\nabla_u F(u^*) = \frac{\partial F}{\partial u}(u^*)^T = 0$ (2.12)

second order condition: $\nabla_u^2 F(u^*) \geq 0$ *(Hessian pos. semidef. at u^*)* (2.13)

Remark: If $F(u)$ is convex these conditions are also *sufficient* for a (in fact global) minimum.

Theorem 2.4 (Sufficient conditions) *If the first order necessary condition (2.12) holds and*

$$\nabla_u^2 F(u^*) > 0 \qquad \textit{(Hessian pos. def. at } u^*) \tag{2.14}$$

then $F(u)$ has a (local) minimum at u^* (since F is locally convex around u^*).

These conditions can be derived from the Taylor-expansion of F around a minimum u^*, e.g. the first order necessary condition is obtained by:

$$F(u^* + \Delta u) = F(u^*) + \nabla_u F(u^*)^T\,\Delta u + \dots \tag{2.15}$$

Within a small neighbourhood of a minimum u^* the value of F must not decrease, i.e. the first order term $\nabla_u F(u^*)^T\,\Delta u \geq 0$ for all possible directions of Δu. Consider the variations $\Delta u = \Delta u^1$ and $\Delta u = -\Delta u^1$: in both cases $\nabla_u F(u^*)^T\,\Delta u \geq 0$ has to hold $\Rightarrow$ (2.12) □

Example 2.5: *Ball and spring no. 2*

We consider the simplest case without any restrictions (walls), see Figure 2.6, and obtain the following unconstrained convex minimization problem:

$$\min_{u \in \mathbb{R}^2} \underbrace{u_1^2 + u_2^2}_{spring} + \underbrace{mgu_2}_{gravity} \tag{2.16}$$

$$\nabla F(u^*) = \begin{bmatrix} 2u_1^* \\ 2u_2^* + mg \end{bmatrix} = 0 \quad \Rightarrow \quad \begin{matrix} u_1^* = 0 \\ u_2^* = -\frac{mg}{2} \end{matrix}$$

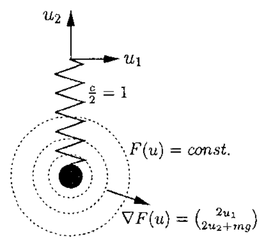


Figure 2.6: Setup of the unconstrained ball and spring problem

2.3.2 Iterative/numerical solution methods

The basic idea of these numerical solution methods is to generate a sequence of inputs u^k which converge to the optimal input vector u^*:

$$u^k \xrightarrow{k\to\infty} u^* \qquad \text{(k: iteration index)}$$

A general algorithm for generating u^{k+1} from u^k looks as follows:

1. Find search direction p^k s.t. the cost is decreasing $\Leftrightarrow$ $\nabla_u F(u^k)^T p^k \leq 0$ (2.17)
2. Pick step length α^k (typically: line search)
3. $u^{k+1} = u^k + \alpha^k p^k$ (2.18)

Remark: While the search direction is fixed by the choice of p^k, the length of the search step can be varied/optimized by the factor α^k, e.g. using a line search algorithm (1-D optimization).

- *Steepest descent/gradient method*

$$\begin{aligned}
\text{based on linear approximation:} \quad & F(u^k + p^k) \approx F(u^k) + \nabla_u F(u^k)^T p^k \\
\text{choose "steepest descent" direction:} \quad & p^k = -\nabla_u F(u^k) && (2.19)\\
\Rightarrow \quad & u^{k+1} = u^k - \alpha^k \nabla_u F(u^k) \\
\textit{linear} \text{ convergence:} \quad & \frac{\|u^{k+1} - u^*\|}{\|u^k - u^*\|} < \beta^k, \quad \beta^k \in (0,1) && (2.20)
\end{aligned}$$

- `Newton-Raphson` *method*

$$\begin{aligned}
\text{based on sec. order approx.:} \quad & F(u^k + p^k) \approx F(u^k) + \nabla_u F(u^k)^T p^k + \frac{1}{2}(p^k)^T \nabla^2 F(u^k)\, p^k \\
\text{min. of this approx. w.r.t. } p^k\text{:} \quad & \frac{\partial F(u^k + p^k)}{\partial p^k} = \nabla F(u^k)^T + (p^k)^T \nabla^2 F(u^k) \stackrel{!}{=} 0^T \\
\Rightarrow \quad & p^k = -\left(\nabla^2 F(u^k)\right)^{-1} \nabla_u F(u^k) && (2.21)\\
\Rightarrow \quad & u^{k+1} = u^k + \alpha^k p^k \\
\textit{quadratic} \text{ convergence:} \quad & \frac{\|u^{k+1} - u^*\|}{\|u^k - u^*\|^2} < \beta^k, \quad \beta^k \in (0,1) && (2.22)
\end{aligned}$$

Remarks:

- steepest descent: simple to calculate, gradient required, only linear convergence
- `Newton-Raphson`: quadratic convergence, gradient and `Hessian` required, need to calculate the inverse, need to store matrices
- derivatives (first and second) can be determined numerically (e.g. finite difference for $\nabla_u F(u)$, approximated `Hessian`, see [])

2.4 Equality constrained static optimization

Problem setup:

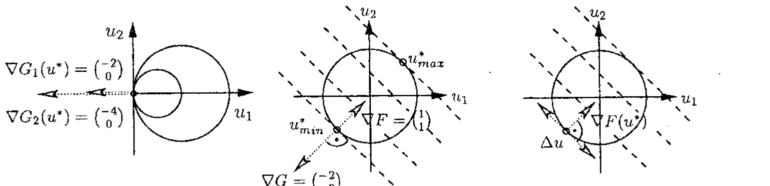
$$\min_{u \in \mathbb{R}^n} F(u) \qquad s.t. \quad G(u) = 0 \qquad (G \in \mathbb{R}^m) \qquad (2.23)$$

Remarks:

- dimensions: normally $m < n$
- often it is necessary to assume that the gradients $\nabla G_i(u^*)$ are linearly independent at the optimum (*regularity condition*), e.g. the following case, illustrated in Figure 2.7(a), would not be allowed:

$$\begin{aligned}
\min_{u \in \mathbb{R}^2} u_1 + u_2 \quad s.t. \quad & G_1(u) = (u_1 - 1)^2 + u_2^2 - 1 = 0 \\
& G_2(u) = (u_1 - 2)^2 + u_2^2 - 4 = 0
\end{aligned} \qquad (2.24)$$

$$\Rightarrow \quad \nabla G_1(u) = \begin{pmatrix} 2(u_1 - 1) \\ 2u_2 \end{pmatrix}, \quad \nabla G_2(u) = \begin{pmatrix} 2(u_1 - 2) \\ 2u_2 \end{pmatrix}$$



(a) Example for non regular gradients $\nabla G_1(u^*)$, $\nabla G_2(u^*)$.

(b) First order condition: $\nabla F(u^*)$ belongs to the subspace of the $\nabla G_i(u^*)$.

(c) First order condition: $\nabla F(u^*)$ is orthogonal to the first order feasible variations Δu.

Figure 2.7: Regularity condition and first order condition for equality constrained problems.

2.4.1 Necessary conditions

Theorem 2.5 (First order necessary condition) *Let u^* be a local minimum of $F(u)$ s.t. $G(u) = 0$ $\quad (G \in \mathbb{R}^m)$. Assume that the $\nabla G_i(u^*)$ are linearly independent. Then there exists a unique vector $\lambda^* = [\lambda_1^*, \ldots, \lambda_m^*]^T$ called Lagrange multiplier vector s.t.*

$$\nabla F(u^*) + \sum_{i=1}^{m} \lambda_i^* \nabla G_i(u^*) = 0 \tag{2.25}$$

Remarks:

- (2.25) $\Leftrightarrow$ *$\nabla F(u^*)$ belongs to the subspace spanned by the $\nabla G_i(u^*)$*, consider e.g.:

$$\min_{u \in \mathbb{R}^2} u_1 + u_2 \qquad s.t. \quad u_1^2 + u_2^2 = 2 \tag{2.26}$$

$$\Rightarrow \nabla F(u) = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \qquad \nabla G(u) = \begin{pmatrix} 2u_1 \\ 2u_2 \end{pmatrix}$$

$$\left.\begin{array}{ll} \overset{(2.25)}{\Rightarrow} & 1 + \lambda^* 2u_1^* = 0 \\ & 1 + \lambda^* 2u_2^* = 0 \\ \overset{G(u)=0}{\Rightarrow} & (u_1^*)^2 + (u_2^*)^2 = 2 \end{array}\right\} \quad \begin{array}{l} u^* = \pm\binom{1}{1} \\ \lambda^* = \pm\frac{1}{2} \end{array}$$

See Figure 2.7(b) and note that we could not distinguish between minima and maxima as we have only considered the first order condition!

- The cost gradient *$\nabla F(u^*)$ is orthogonal to the subspace of first order feasible variations* $\{\Delta u \,|\, \nabla G_i(u^*)^T \Delta u = 0 \quad (i = 1, \ldots, m)\}$, cmp. Figure 2.7(c).

Clarification/derivation of Theorem 2.5:

1. Special case: *linear constraints*

$$\min_{u \in \mathbb{R}^n} F(u) \qquad s.t. \quad Au = b \qquad (A \in \mathbb{R}^{m \times n} \text{ with independent rows}, b \in \mathbb{R}^m) \tag{2.27}$$

- considering the constraint equation we can split up u into two parts:

$$u = [\underbrace{x_1 \ldots x_m}_{\text{m dep. var.}}, \underbrace{\tilde{u}_1 \ldots \tilde{u}_{n-m}}_{\text{(m-n) indep. var.}}]^T = \begin{bmatrix} x \\ \tilde{u} \end{bmatrix} \quad \Rightarrow \quad \underbrace{A}_{m \times n} = [\underbrace{A_x}_{m \times m} \quad \underbrace{A_u}_{m \times (n-m)}] \tag{2.28}$$

- as the rows of A and therefore also of A_x are linearly independent, A_x is invertible and we can eliminate x from the constraint equation:

$$A_x x + A_u \tilde{u} = b \qquad \Rightarrow \quad x = x(\tilde{u}) = A_x^{-1}(b - A_u \tilde{u}) \tag{2.29}$$

- rewriting the minimization problem we obtain:

$$\min_{\tilde{u} \in \mathbb{R}^{n-m}} \tilde{F}(\tilde{u}) \quad \text{with} \quad \tilde{F}(\tilde{u}) := F\left(\begin{bmatrix} A_x^{-1}(b - A_u \tilde{u}) \\ \tilde{u} \end{bmatrix}\right) \tag{2.30}$$

which actually is an *unconstrained* optimization problem.

- the first order optimality condition yields:

$$\begin{array}{lll} \nabla_{\tilde{u}} \tilde{F}(\tilde{u}^*) \overset{!}{=} 0 \quad \Leftrightarrow & \nabla_{\tilde{u}} x \, \nabla_x F(u^*) & + \nabla_{\tilde{u}} F(u^*) = \\ & -A_u^T \underbrace{(A_x^T)^{-1} \nabla_x F(u^*)}_{\lambda^* := -(A_x^T)^{-1} \nabla_x F(u^*)} & + \nabla_{\tilde{u}} F(u^*) \overset{!}{=} 0 \end{array} \tag{2.31}$$

$$\begin{array}{rl} \Rightarrow & \nabla_{\tilde{u}} F(u^*) + A_u^T \lambda^* = 0 \\ \text{per def. } \lambda^*: & \nabla_x F(u^*) + A_x^T \lambda^* = 0 \end{array} \Bigg\} \quad \Leftrightarrow \quad \nabla F(u^*) + A^T \lambda^* = 0 \tag{2.32}$$

as stated in (2.25).

2. *General case:*

$$\min_{u\in\mathbb{R}^n} F(u) \qquad s.t. \quad G(u) = 0 \quad \text{(assume: } \nabla G_i(u^*) \text{ linearly independent)} \tag{2.33}$$

- we again split up into dependent and independent variables: $u^T = [x^T, \tilde{u}^T]$

$$\rightsquigarrow \qquad \min_{\substack{x\in\mathbb{R}^m \\ \tilde{u}\in\mathbb{R}^{n-m}}} F\left(\begin{bmatrix} x \\ \tilde{u} \end{bmatrix}\right) \qquad s.t. \quad G\left(\begin{bmatrix} x \\ \tilde{u} \end{bmatrix}\right) = 0 \tag{2.34}$$

- now we look at the first order variations at a stationary point/minima (which lies in the feasible region $\Leftrightarrow$ G = 0):

$$dF = \quad F_x\, dx + F_{\tilde{u}}\, d\tilde{u} \quad \stackrel{!}{=} 0 \qquad \text{while the constraint variation} \tag{2.35}$$

$$dG = \quad G_x\, dx + G_{\tilde{u}}\, d\tilde{u} \quad \stackrel{!}{=} 0 \qquad \text{(keep feasibilty!)} \tag{2.36}$$

- as G_x^{-1} exists we can solve (2.36) for

$$dx = -G_x^{-1}\, G_{\tilde{u}}\, d\tilde{u} \tag{2.37}$$

and plug it in (2.35), holding $dG = 0$

$$dF = (\underbrace{-F_x\, G_x^{-1}}_{(\lambda^*)^T} G_{\tilde{u}} + F_{\tilde{u}})\, d\tilde{u} \stackrel{!}{=} 0 \qquad \forall d\tilde{u} \tag{2.38}$$

$$\begin{array}{ll} (2.38) \quad \Rightarrow & \nabla_{\tilde{u}} F(u^*)^T + (\lambda^*)^T \nabla_{\tilde{u}} G(u^*)^T = 0^T \\ \text{per def. } \lambda^*: & \nabla_x F(u^*)^T + (\lambda^*)^T \nabla_x G(u^*)^T = 0^T \end{array} \quad \Leftrightarrow \quad (2.25)$$

□

Theorem 2.6 (Second order necessary condition) *Let u^* be a local minimum of $F(u)$ s.t. $G(u) = 0$ $(G \in \mathbb{R}^m)$ and let the gradients $\nabla G_i(u^*)$ be linearly independent. Then*

$$\forall p \;\; s.t. \; \nabla G(u^*)^T p = 0: \qquad p^T \left(\nabla^2 F(u^*) + \sum_{i=1}^{m} \lambda_i^* \nabla^2 G_i(u^*)\right) p \geq 0 \tag{2.39}$$

with the `Lagrange` *multiplier vector $\lambda^* = [\lambda_1^*, \ldots, \lambda_m^*]^T$ of equ. (2.25).*

Note: If the left side of inequation (2.39) is strictly greater zero the condition is also *sufficient* for a local minimum.

2.4.2 Lagrangian and Hamiltonian

To simplify the notation we introduce the

Definition 2.5 (Lagrangian)

$$\mathcal{L}(u,\lambda) = F(u) + \lambda^T G(u) \qquad (u \in \mathbb{R}^n; \quad G, \lambda \in \mathbb{R}^m) \tag{2.40}$$

and summarize Theorem 2.5 and 2.6 in

Theorem 2.7 (Necessary conditions using the Lagrangian) *If u^* is a local minimum of $F(u)$ s.t. $G(u) = 0$ $(G \in \mathbb{R}^m)$ and the gradients $\nabla G_i(u^*)$ are linearly independent then there exists a* `Lagrange` *multiplier vector $\lambda^* \in \mathbb{R}^m$ s.t.*

first order condition $\quad \begin{array}{ll} (n \text{ equ.}) & \nabla_u \mathcal{L}(u^*,\lambda^*) = 0 \\ (m \text{ equ.}) & \nabla_\lambda \mathcal{L}(u^*,\lambda^*) = 0 \end{array} \quad (2.41) \quad \Leftrightarrow \quad \begin{array}{l} \nabla_u F(u^*) + \nabla_u G(u^*)\,\lambda^* = 0 \\ G(u^*) = 0 \end{array}$

second order condition $\quad p^T \nabla_u^2 \mathcal{L}(u^*,\lambda^*)\, p \geq 0 \quad \forall p \;\; s.t. \;\; \nabla G(u^*)^T p = 0 \qquad (2.42)$

Note: The first order condition provides (n+m) equations for the (n+m) unknowns (u^*, λ^*).

Solving strategy:

1. Build the Lagrangian $\mathcal{L}(u,\lambda)$.
2. Calculate the gradients $\nabla_u \mathcal{L}$, $\nabla_\lambda \mathcal{L}$.
3. Solve the first order condition (2.41) for u^*, λ^*.
4. If necessary (e.g. the problem is not convex) check the second order condition (2.42).

As we want to go further from static optimization to optimal control problems it is useful to define the so called

Definition 2.6 (Hamiltonian) *The* `Hamiltonian` *function for the optimization problem*

$$\left.\begin{array}{l} \min\limits_u \displaystyle\int_0^{t_f} F(x,u)\, d\tau \\ \dot{x} = f(x,u), \quad x(0) = x_0 \end{array}\right\} \quad \text{is defined as} \quad \mathcal{H}(x,u,\lambda) = F(x,u) + \lambda^T f(x,u) \tag{2.43}$$

Technically, the set of equality constraints has here been replaced by a set of differential equations with boundary/initial conditions. Through the introduction of the `Hamiltonian` such problems can be handled similar to unconstrained problems, likewise as we used the `Lagrangian` for equality constrained problems.

Example 2.6: *Ball and spring no. 3*

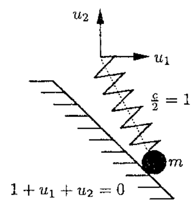


Figure 2.8: Ball and spring problem with one inequality constraint

The problem is illustrated in Figure 2.8. We want to determine the stationary point, i.e. the point of minimal potential energy:

$$\min_{u \in \mathbb{R}^2} u_1^2 + u_2^2 + mgu_2 \quad s.t. \quad -1 - u_1 - u_2 = 0 \tag{2.44}$$

Solution using the Lagrangian:

1. $\mathcal{L}(u, \lambda) = u_1^2 + u_2^2 + mgu_2 - \lambda - \lambda u_1 - \lambda u_2$
2. $\nabla_u \mathcal{L}(u, \lambda) = \binom{2u_1 - \lambda}{2u_2 + mg - \lambda} \qquad \nabla_\lambda \mathcal{L}(u, \lambda) = -1 - u_1 - u_2$
3. $\left.\begin{aligned} \nabla_u \mathcal{L}(u^*, \lambda^*) &\overset{!}{=} 0 \\ \nabla_\lambda \mathcal{L}(u^*, \lambda^*) &\overset{!}{=} 0 \end{aligned}\right\} \quad \begin{aligned} &\lambda^* = \tfrac{mg}{2} - 1 \\ &u_1^* = \tfrac{mg}{4} - \tfrac{1}{2}, \quad u_2^* = -\tfrac{mg}{4} - \tfrac{1}{2} \end{aligned}$
4. problem is convex $\Rightarrow$ u^* is the global minimum.

2.5 Inequality constrained minimization

We consider the problem:

$$\min_{u \in \mathbb{R}^n} F(u) \quad s.t. \quad \begin{aligned} G(u) &= 0 \qquad (G \in \mathbb{R}^m) \\ H(u) &\leq 0 \qquad (H \in \mathbb{R}^r) \end{aligned} \tag{2.45}$$

Definition 2.7 (Active and inactive constraints) *The inequality constraint H_i is called an*

$$\begin{aligned} &\textit{active constraint:} && H_i(u^*) = 0 \quad \textit{or an} \\ &\textit{inactive constraint:} && H_i(u^*) < 0. \end{aligned}$$

Some examples are shown in Figure 2.9.

Figure 2.9: Classification of inequality constraints

Remarks:

- If an inequality constraint is already known to be *inactive* (at the optimum) it can be dropped in further calculations.
- (2.45) can be handled with Lagrange multipliers.
- Motivation for first order conditions (scalar case):

$$\min_{u \in \mathbb{R}} F(u) \quad s.t.\ H(u) = 0 \qquad (H \in \mathbb{R}) \tag{2.46}$$

two cases:

1. H inactive $\rightsquigarrow$ unconstrained minimization
2. H active ($H(u^*) = 0$): consider a small (feasible) perturbation in u

$$dF = \tfrac{\partial F}{\partial u}(u^*)\, du \geq 0 \qquad (u^* \text{ minimum!}) \tag{2.47}$$
$$dH = \tfrac{\partial H}{\partial u}(u^*)\, du \leq 0 \qquad (\text{constraint satisfied!}) \tag{2.48}$$

$\Rightarrow$ two possiblities: either $\quad sgn\, \frac{\partial F}{\partial u}(u^*) = -sgn\, \frac{\partial H}{\partial u}(u^*)$, or $\quad \frac{\partial F}{\partial u}(u^*) = \frac{\partial H}{\partial u}(u^*) = 0$

$$\text{in short} \qquad \tfrac{\partial F}{\partial u}(u^*) + \lambda \tfrac{\partial H}{\partial u}(u^*) = 0 \qquad \lambda \geq 0 \tag{2.49}$$

These considerations are the basis for the first order necessary conditions (KKT conditions) stated below (compare Thm. 2.8).

Definition 2.8 (Generalized Lagrangian) *The generalized Lagrangian for the inequality constrained minimization problem (2.45) is defined as*

$$\mathcal{L}(u, \lambda, \mu) = F(u) + \sum_{i=1}^{m} \lambda_i G_i(u) + \sum_{i=1}^{r} \mu_i H_i(u) \qquad (\lambda \in \mathbb{R}^m, \mu \in \mathbb{R}^r) \tag{2.50}$$

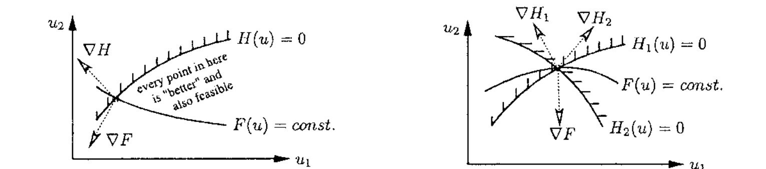
Theorem 2.8 (First order necessary conditions: the KKT[1] conditions) *Let u^* be a regular point (i.e. the $\nabla G_i(u^*)$ and the $\nabla H_i(u^*)$ are linearly independent respectively) and a local minimum for problem (2.45). Then there exists $\lambda^* = [\lambda_1^*, \ldots, \lambda_m^*]^T$, $\mu^* = [\mu_1^*, \ldots, \mu_r^*]^T$ s.t.*

$$\nabla_u \mathcal{L}(u^*, \lambda^*, \mu^*) = 0 \qquad \begin{aligned} &H_i \textit{ active:} && \mu_i^* \geq 0 \\ &H_i \textit{ inactive:} && \mu_i^* = 0 \end{aligned} \tag{2.51}$$

Remarks:

- Of course also $\nabla_\lambda \mathcal{L}(u^*, \lambda^*, \mu^*) = 0$ (i.e. $G(u^*) = 0$) still has to hold.
- If for all active constraints H_i the corresponding μ_i is strictly greater zero the conditions are also *sufficient.* ??

The proof is not presented here, we just consider some graphical explanations, see Figure 2.5.



(a) One inequality constraint, KKT not satisfied

(b) Two inequality constraints, KKT satisfied: $\nabla F(u^*)$ lies here in the subspace spanned by the active constraints $\nabla H_i(u^*)$

Figure 2.10: Graphical explanations to the KKT conditions.

Additionally, the example in Figure 2.11 shows the importance of verifying the constraints, i.e. to make sure that the gradients of all active constraints are linearly independent at the optimum. In Figure 2.11, $\nabla H_1(u^*)$ and $\nabla H_2(u^*)$ are both active but linearly dependent, therefore we cannot apply the KKT conditions (note that u^* is a minimum though ∇F does not belong to the subspace spanned by $\nabla H_1; \nabla H_2$).

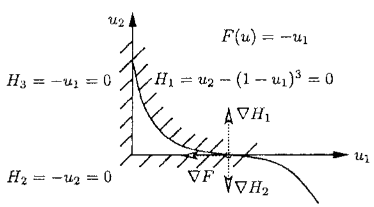


Figure 2.11: Constraint verification: linear dependent gradients $\nabla H_1(u^*), \nabla H_2(u^*)$

[1]Karush Kuhn Tucker

Example 2.7: *Ball and spring no. 4*

Figure 2.12 shows one more modification of the ball and spring problem, now with two active constraints.

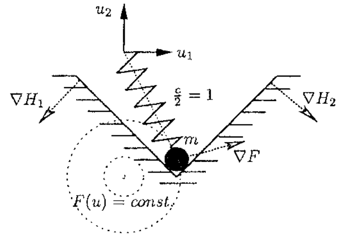


Figure 2.12: Setup of the ball and spring problem no. 4

The resulting problem can be formulated as follows:

$$\min_{u\in\mathbb{R}^2} u_1^2 + u_2^2 + mgu_2 \qquad s.t. \quad \begin{aligned} H_1(u) &= -1 - u_1 - u_2 \leq 0 \\ H_2(u) &= -3 + u_1 - u_2 \leq 0 \end{aligned} \tag{2.52}$$

$$\leadsto \mathcal{L}(u,\mu) = u_1^2 + u_2^2 + mgu_1 + \mu_1 H_1 + \mu_2 H_2 \tag{2.53}$$

$$\nabla_u \mathcal{L}(u^*, \mu^*) = 0 \quad \Leftrightarrow \quad \nabla F(u^*) + \underbrace{\mu_1^* \nabla H_1(u^*) + \mu_2^* \nabla H_2(u^*)}_{\text{"constraint forces"}} = 0 \tag{2.54}$$

Note that in mechanical problems the KKT conditions yield the *eqilibrium of forces*!

Example 2.8: *Calculating the minimal feasible distance to the origin*

$$\min_{u\in\mathbb{R}^3} \frac{1}{2}(u_1^2 + u_2^2 + u_3^2) \qquad s.t. \quad u_1 + u_2 + u_3 \leq -3 \tag{2.55}$$

$$\leadsto \mathcal{L}(u,\mu) = \frac{1}{2}(u_1^2 + u_2^2 + u_3^2) + \mu(u_1 + u_2 + u_3 + 3) \tag{2.56}$$

$$\text{first order cond.:} \quad \nabla_u \mathcal{L}(u^*, \mu^*) = 0 \quad \Leftrightarrow \quad \begin{cases} u_1^* + \mu^* = 0 \\ u_2^* + \mu^* = 0 \\ u_3^* + \mu^* = 0 \end{cases} \tag{2.57}$$

two cases: $\quad H(u) = u_1 + u_2 + u_3 + 3 \quad \ldots$

- *inactive:* $u_1^* + u_2^* + u_3^* < -3 \quad \wedge \quad \mu^* = 0$

  (2.57) $\;\Rightarrow\; u_1^* = u_2^* = u_3^* = 0 \;\Rightarrow\; 0 < -3 \quad \leadsto$ contradiction!

- *active:* $\left.\begin{array}{l} u_1^* + u_2^* + u_3^* = -3 \\ (2.57): \quad \nabla_u \mathcal{L}(u^*, \mu^*) = 0 \end{array}\right\}$ 4 equations for 4 unknowns

  $\Rightarrow \quad \mu^* = 1, \quad u_1^* = u_2^* = u_3^* = -1 \quad$ is local min. (check second order cond.!)

  Note that $\nabla H(u^*) = [1\ 1\ 1]^T$ is regular!

Remark: In general we have to look at each possible combination of active and inactive constraints!

2.6 Lagrange multipliers/shadow price

Besides dividing active and inactive constraints there is an extended meaning of the Lagrange multipliers λ and μ. Consider the inequality constraint $h(u)$ of the following problem:

$$F(u^*) = \min_{u\in\mathbb{R}^n} F(u) \quad s.t. \quad h(u) \le 0 \qquad (h \in \mathbb{R}) \tag{2.58}$$

If we now relax/shift the constraint (as illustrated in Figure 2.13):

$$h(u) - \epsilon \le 0 \quad \Rightarrow \quad \ldots \quad \Rightarrow F(u_\epsilon^*) \approx F(u^*) - \mu^* \epsilon \tag{2.59}$$

the optimal cost is also changing depending on the Lagrange multiplier: μ^* gives the *"hidden cost"* for satisfying the constraint.

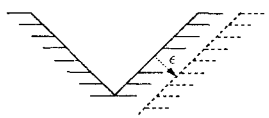


Figure 2.13: Relaxing an inequality constraint

In general:

$$\left.\frac{\partial F}{\partial H_i}\right|_{\mathcal{L}_u,\, \mathcal{L}_x \approx const.} = -\mu_i \tag{2.60}$$

The Lagrange multipliers are therefore also known as *shadow price*:

$|\mu_i| \gg 1 \quad \rightsquigarrow \quad$ constraint H_i "expensive" to satisfy
$\mu_i \approx 0 \quad \rightsquigarrow \quad$ constraint H_i "inexpensive" to satisfy

2.7 Iterative solution methods for constrained optimization

Roughly spoken there are two different approaches to solve a *static constrained optimization problem*:

1. transform the problem directly to an unconstrained problem $\rightsquigarrow$ solve unconstrained problem (cmp. 2.3.2):
   - penalty method
   - barrier method
   - method of augmented Lagrangian
2. apply approximation methods to KKT conditions:
   - SQP method (Newton's method applied to KKT)

2.7.1 Quadratic penalty method

The idea of this method is to approximate the original constrained problem through the unconstrained problem with an additional quadratic penalizing term for constraint violations:

$$\begin{aligned} \min_{u\in\mathbb{R}^n} F(u) \quad & s.t.\ G(u) = 0 \\ & H(u) \le 0 \end{aligned} \quad \rightsquigarrow \quad \overbrace{\min_{u\in\mathbb{R}^n} F(u) + \underbrace{\frac{c}{2}\left(\sum_{i=1}^{m} G_i^2(u) + \sum_{i=1}^{r} \min(0, -H_i^2(u))\right)}_{\text{penalizing term}}}^{Q_c(u)} \tag{2.61}$$

In general the resulting problem (2.61) is solved iteratively increasing the penalization factor $c > 0$ step by step so that the unconstrained minimum is more and more "shifted towards feasibility".

Example 2.9: *Quadratic penalty method*
We are searching for the minimum of the sine function within a given intervall $u \in [0; 1]$:

$$\min_{u\in\mathbb{R}} sin(u) \qquad s.t. \quad \begin{aligned} -u &\le 0 \\ u - 1 &\le 0 \end{aligned} \tag{2.62}$$

Applying the quadratic penalty method we obtain an extended cost function

$$Q_c(u) = \sin(u) + \frac{c}{2}\left(min(0, -u^2) + min(0, -(u-1)^2)\right) \tag{2.63}$$

for the unconstrained problem $\min_{u\in\mathbb{R}} Q_c(u)$ which approximates (2.62) for big penalization factors c. Figure 2.14(a)a shows the graphs of the original cost function $F(u)$ and the cost functions of the transformed problems $Q_c(u)$ for different values of c.

General algorithm for the quadratic penalty method:

1. Choose an initial penalization factor c^0 (small) and an initial guess for u^0.
2. Solve $\min\limits_{u^k\in\mathbb{R}^n} F(u^k) + \frac{c^k}{2}\left(\sum\limits_{i=1}^{m} G_i^2(u^k) + \sum\limits_{i=1}^{r} \min(0, -H_i^2(u^k))\right)$ for $u^k = u^k(c^k)$.
3. Increase c^k: $c^{k+1} > c^k$, use $u^k(c^k)$ as initial guess for $u^{k+1}(c^{k+1})$ and go back to *2*.

$\Rightarrow \quad u^* = \lim\limits_{c^k\to\infty} u^k(c^k)$

Remarks:

+ uses algorithm for unconstrained optimization
+ for an equality constrained problem the cost function of the resulting unconstrained problem $Q_c(u)$ is at least as smooth as $F(u), G(u)$
- in the case of inequality constraints the `Hessian` is not differentiable (*min* function not smooth!)
+ can solve each optimization problem approximately
- guarantie for a minimum only for $c \to \infty$, but then the problem gets *numerically* ill conditioned
- for finite c the method may yield infeasible solutions

2.7.2 Logarithmic barrier method

We consider pure *inequality* constrained problems, see (2.64). As before the constrained problem is approximated by an unconstrained problem, now using the extended cost function $P_\nu(u)$:

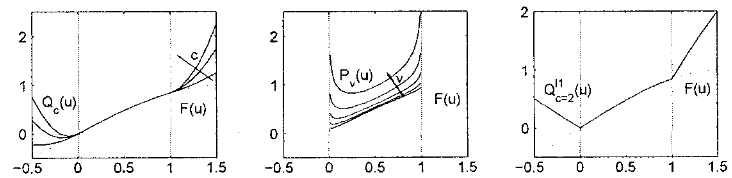
$$\min_{u\in\mathbb{R}^n} F(u) \quad s.t. \quad H(u) \le 0 \quad \rightsquigarrow \quad \min_{u\in\mathbb{R}^n} \underbrace{F(u) - \nu \sum_{i=1}^{r} \log H_i(u)}_{P_\nu(u)} \tag{2.64}$$

The additional logarithmic term $-\nu \sum_{i=1}^{r} \log H_i(u)$ builds up a kind of barrier at the border of the feasible region, cmp. Figure 2.14(b). The algorithm for solving the inequality problem via the logarithmic barier method is similar to the algorithm of the penalty method considered above: Starting from an arbitrary feasible point u^0 and a factor $\nu^0 > 0$ the problem is solved iteratively by decreasing ν^k, each iteration step solving (2.64). If the solution $u^k(\nu^k)$ of the previous step is used as initial guess for the subsequent step the solution finally converges to the minimum u^* for $\nu \to 0$. Note also that $\lim\limits_{\nu\to 0} P_\nu(u) = F(u)$.
Equality constraints can be included via the quadratic penalty approach.

Example 2.10: *Logarithmic barrier method*
We apply the logarithmic barrier method to Example 2.9 and obtain the modified cost function

$$P_\nu(u) = sin(u) - \nu(\,log(-u) + log(u-1)\,) \tag{2.65}$$

for the unconstrained problem $\min\limits_{u\in\mathbb{R}} P_\nu(u)$.
A plot of the cost functions for decreasing values of ν is shown in Figure 2.14(b).



(a) Quadratic penalty method (b) Logarithmic barrier method (c) Exact penalty approach

Figure 2.14: Cost functions for problem (2.62) applying different solution methods.

Remarks on the logarithmic barrier method:

+ intermediate minimizers are feasible
- a feasible initial guess is required
+ the cost function of the resulting unconstrained problem $P_\nu(u)$ is at least as smooth as the constraint functions
- problem of ill conditioning with growing iteration steps ($\nu \to 0$) remains
- optima on the borders are difficult to find ($\nu \to 0$, $log(...) \to \infty$)

2.7.3 Exact penalty approach

Applying one of the methods above we could only guarantie for evaluating the real minimum with the approximated cost functions $Q^c(u)$ and $P^\nu(u)$ in the limit $c \to \infty$ and $\nu \to 0$ respectively. This problem can be avoided by using the l_1-norm instead of the quadratic terms in the quadratic penalty approach (2.61)

$$\begin{aligned}\min_{u\in\mathbb{R}^n} F(u) \quad s.t.\ & G(u) = 0 \\ & H(u) \le 0\end{aligned} \quad \rightsquigarrow \quad \min_{u\in\mathbb{R}^n} \underbrace{F(u) + c\left(\sum_{i=1}^{m} |G_i(u)| + \sum_{i=1}^{r} |\min(0, -H_i(u))|\right)}_{Q_c^{l_1}(u)} \tag{2.66}$$

In Figure 2.14(c) a plot of the resulting unconstrained cost functions is shown, calculated for Example 2.9 with (2.66).

Remarks:

+ we obtain exact solutions for finite values of c
- resulting cost function is non smooth

2.7.4 Method of augmented Lagrangian

We consider the equality constrained case. If we compare the quadratic penalty approach to the Lagrangian approach for constrained problems (see (2.50), (2.51)) we obtain from the first order necessary conditions the relation

$$G_i(u^k) \approx -\frac{1}{c^k}\lambda_i^* \tag{2.67}$$

with a lagrange multiplier λ_i^* for the quadratic penalty method. As in general $\lambda_i^* \neq 0$ we have to increase the penalization factor $c \to \infty$ in order to satisfy the constraints, (2.67) tends to zero. As discussed above we then get the problem of numerical ill conditioning.
To avoid that, we use an augmented Lagrangian function instead:

$$\mathcal{L}_A(u,\lambda,c) = F(u) + \sum_{i=1}^{m} \lambda_i\, G_i(u) + \frac{c}{2}\sum_{i=1}^{m} G_i^2(u) \tag{2.68}$$

$$\Rightarrow \quad \nabla_u \mathcal{L}_A(u,\lambda,c) = \nabla F(u) + \sum_{i=1}^{m}(\lambda_i + c\,G_i)\,\nabla G_i(u) \tag{2.69}$$

The main idea of this method is the iterative adaption of the term $(\lambda_i + c\,G_i)$ in (2.69) which corresponds to the Lagrange multiplier in the first order conditions (2.25). This adaption takes place in step 3 of the following *algorithm:*

1. Fix c to c^k, fix λ to λ^k.
2. Solve the unconstrained problem $\min_{u^k} \mathcal{L}_A(u^k,\lambda^k,c^k)$ using (2.69) $\leadsto$ u^{k+1}.
3. Increase $c: c^{k+1} > c^k$, set $\lambda_i^{k+1} = \lambda_i^k + c^k G_i(u^k)$, increase k and go back to 2.

Perturbation analysis yields again an estimation of the constraint error:

$$G_i(u^k) \approx \frac{1}{c^k}\underbrace{(\lambda_i^* - \lambda_i^k)}_{\approx 0 \text{ for } k \gg 1} \tag{2.70}$$

With increasing k the λ_i^k converge to the λ_i^* and the constraint equations can now be already satisfied for finite values of c, see (2.70).

Remarks:

+ removes ill conditioning
+ convergence for finite penalization factor c
+ smooth objective function
+ delivers an approximation of the Lagrange multipliers

2.7.5 SQP method (Sequential Quadratic Programming)

Different from the methods presented above the SQP method directly starts from the KKT conditions, which are iteratively approximated by solving a quadratic problem each iteration step. We first consider the pure equality constrained case:

$$\min_{u\in\mathbb{R}^n} F(u) \quad s.t.\ G(u) = 0 \tag{2.71}$$

The KKT conditions demand for the $(n+m)$ unknowns (u^*,λ^*) that

$$\begin{array}{ll} \text{(n equ.)} & \nabla_u \mathcal{L}(u^*,\lambda^*) = 0 \\ \text{(m equ.)} & \nabla_\lambda \mathcal{L}(u^*,\lambda^*) = 0 \end{array} \quad \Leftrightarrow \quad \begin{array}{l} \nabla_u F(u^*) + \nabla_u G(u^*)\,\lambda^* = 0 \\ G(u^*) = 0 \end{array} \tag{2.72}$$

We now use the `Newton` method to solve (2.72) iteratively for (u^*,λ^*):

$$\begin{aligned} u^{k+1} &= u^k + \alpha p_u^k \\ \lambda^{k+1} &= \lambda^k + \alpha p_\lambda^k \end{aligned} \tag{2.73}$$

The step length can optionally be adapted by a line search optimization for α. The key issue is to evaluate an appropriate search direction $(p_u^k, p_\lambda^k) \in \mathbb{R}^{n+m}$. Applying `Newton` in order to find a zero of (2.72) (or equivalently: plugging (u^{k+1},λ^{k+1}) of (2.73) in the KKT conditions (2.72) and considering the first order term of the corresponding `Taylor` series expansion) we obtain

$$\mathcal{O}^2??(p^k) + \begin{bmatrix} \nabla_u^2 \mathcal{L}(u^k,\lambda^k) & \nabla G(u^k) \\ \nabla G(u^k)^T & 0 \end{bmatrix} \begin{bmatrix} p_u^k \\ p_\lambda^k \end{bmatrix} + \begin{bmatrix} \nabla F(u^k) + \nabla G(u^k)\,\lambda^k \\ G(u^k) \end{bmatrix} = 0 \tag{2.74}$$

By neglecting the error term and inverting, equation (2.74) can be solved for the search direction (p_u^k, p_λ^k) each iteration step.
An alternative view on this procedure may clarify the naming of the SQP method:
At every iteration step the following *quadratic* approximated problem with linearized constraints around the current point u^k is solved

$$\left.\begin{array}{ll} \min\limits_{p_u^k\in\mathbb{R}^n} & \frac{1}{2}(p_u^k)^T\,\nabla_u^2\mathcal{L}(u^k)\,p_u^k + \nabla F(u^k)^T p_u^k \\ s.t. & \nabla G(u^k)^T p_u^k + G(u^k) = 0 \end{array}\right\} \quad \Rightarrow \quad p_u^k, p_\lambda^k \tag{2.75}$$

which can be handled with widely-used *Q*uadratic *P*rogramming solvers.

For the general case with inequality constraints we obtain the following subproblems

$$\begin{array}{lll} & & \text{iterate } k = 1,\ldots,\hat{k}: \\ \min\limits_{u\in\mathbb{R}^n} F(u) \quad s.t.\ G(u) = 0 & \overset{SQP}{\leadsto} & \min\limits_{p_u^k\in\mathbb{R}^n} \quad \frac{1}{2}(p_u^k)^T\,\nabla_u^2\mathcal{L}(u^k)\,p_u^k + \nabla F(u^k)^T p_u^k \\ \qquad\qquad\qquad\quad H(u) \le 0 & & s.t. \quad \nabla G(u^k)^T p_u^k + G(u^k) = 0 \\ & & \qquad\ \ \nabla H(u^k)^T p_u^k + H(u^k) \le 0 \end{array} \tag{2.76}$$

and the corresponding *algorithm:*

1. Fix initial values $u^0, \lambda^0, \mu^0,\ k = 0$.
2. Evaluate $G(u^k),\ H(u^k),\ \nabla F(u^k),\ \nabla G(u^k),\ \nabla H(u^k), \nabla_u^2 \mathcal{L}(u^k, \lambda^k, \mu^k)$.
3. Solve (2.76) $\Rightarrow$ $p_u^k,\ p_\lambda^k,\ p_\mu^k$.
4. Update $u^{k+1} = u^k + p_u^k,\quad \lambda^{k+1} = \lambda^k + p_\lambda^k,\quad \mu^{k+1} = \mu^k + p_\mu^k,$ increase k and go back to 2. (until stop criteria is fulfilled).

Remarks:

- need to select active and inactive constraints
- additional line search step is possible (but not included above)
- need for efficient QP solvers
- the gradients $\nabla F, \nabla G, \nabla H$ might be calculated by finite differences
- $\nabla_u^2 \mathcal{L}$ is in general calculated using specific update techniques
- huge storage demand for $\nabla_u^2 \mathcal{L}$ $\rightsquigarrow$ reduced `Hessian` approaches
- SQP is widely used for nonlinear problems

Chapter 3

Formal mathematical setup and mathematical preliminaries

We now come back to our original goal formulated in Chapter 1: to influence a *dynamical* system in an optimal way with respect to a given objective. Precisely, assume that

- the system dynamics,
- (physical) constraints and
- a performance objective

are given, then we want to determine an input $u^*(t)$ (or a feedback $u^* = k(x)$) s.t. the constraints are met and the objective is minimized. Before considering different ways of solving such problems, we first introduce a general mathematical frame in this chapter and also investigate some conditions, which are necessary for the *existence*, *uniqueness* and *well posedness* of a solution.

3.1 System dynamics and properties

In principle, there are various ways of formulating the behaviour of dynamical systems mathematically: PDE's, ODE's, integro differential equations, ... In the following we focus on two classes, *(a) continuous time systems* and *(b) discrete time systems.*

- System equations:

$$\dot{x} = f(t,x,u), \quad x(t) \in \mathbb{R}^n, \quad x(0) = x_0 \qquad u(t) \in \mathbb{R}^m \tag{3.1a}$$

Often, f is continuous and also has continuous derivatives in x and t.

Sampling the continuous time system: $t_k = k\,\delta$ (sampling time δ, time index k) we obtain $x(t_{k+1}) = x(t_k) + \int_{k\delta}^{(k+1)\delta} f(t,x,u)\,d\tau$. Using shortcuts for $x(t_k) =: x_k$, $u(t_k) =: u_k$ and redefining f, the system equation for the discrete time case reads:

$$x_{k+1} = \tilde{f}(k, x_k, u_k), \quad x_k \in \mathbb{R}^n, \quad x_0 \textit{ given} \qquad u_k \in \mathbb{R}^m \tag{3.1b}$$

Special cases:

- time invariant: $\dot{x} = f(x,u)$ (3.2a) $\quad x_{k+1} = \tilde{f}(x_k, u_k)$ (3.2b)

- affine in u: $\dot{x} = A(x) + B(x)\,u$ (3.3a) $\quad x_{k+1} = \tilde{A}(x_k) + \tilde{B}(x_k)\,u_k$ (3.3b)

- linear: $\dot{x} = Ax + Bu$ (3.4a) $\quad x_{k+1} = \tilde{A}x_k + \tilde{B}u_k$ (3.4b)

$$\Rightarrow \quad x(t) = e^{A(t-t_0)}x(t_0) + e^{A(t-t_0)} \int_{t_0}^{t} e^{-A(\tau-t_0)} Bu(\tau)\,d\tau \tag{3.5a}$$

$$\Rightarrow \quad x_{k+1} = x(t_{k+1})|_{t_0=t_k} = \underbrace{e^{A\delta}}_{\tilde{A}} x_k + \underbrace{e^{A\delta} \int_0^{\delta} e^{-A\tau} B\,d\tau}_{\tilde{B}}\, u_k \tag{3.5b}$$

- Steady state: $0 = f(x_s, u_s)$ (3.6a) $\quad x_{k_s+1} = x_{k_s} = \tilde{f}(x_{k_s}, u_{k_s})$ (3.6b)

- Stability: $\forall \epsilon > 0 \quad \exists \delta_\epsilon > 0$:

$$\|x(t)\| \le \delta_\epsilon \quad \Rightarrow \quad \|x(\tau)\| \le \epsilon \quad \forall \tau \ge t \tag{3.7a}$$
$$\|x_k\| \le \delta_\epsilon \quad \Rightarrow \quad \|x_{k+l}\| \le \epsilon \quad \forall l \in \mathbb{N} \tag{3.7b}$$

asympt. stab.: stable and convergent, i.e. (3.7) and for

$$\tau \to \infty: \quad \|x(\tau)\| \to 0 \tag{3.8a}$$
$$l \to \infty: \quad \|x_{k+l}\| \to 0 \tag{3.8b}$$

exp. stab.:

$$\|x(\tau)\| \le \alpha \|x(t)\|\, e^{-\lambda(\tau-t)}, \qquad \lambda > 0 \tag{3.9a}$$
$$\|x_{k+l}\| \le \alpha \|x_k\|\, \gamma^k, \qquad \gamma < 1 \tag{3.9b}$$

stability for linear systems:

eigenv. $Re(\lambda_i(A)) < 0 \quad \Rightarrow \quad \dot{x} = Ax(+Bu)$ asympt. stable (3.10a)

eigenv. $|\lambda_i(\tilde{A})| < 1 \quad \Rightarrow \quad x_{k+1} = \tilde{A}x_k(+\tilde{B}u_k)$ asympt. stable (3.10b)

A visualization is shown in Figure 3.1.
Note also, that for linear systems: asympt. stab. $\Leftrightarrow$ exp. stab.

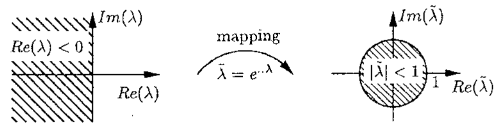


Figure 3.1: Mapping the stable region of a linear continuous time system into the stable region of a corresponding discrete time system (cmp. the definition of $\tilde{A}$, (3.5b)).

- Ljapunow theorems:

We consider the autonome systems with a stationary point in the origin

$$\dot{x} = f(x)\,, \quad f(0) = 0 \qquad (3.11a) \qquad\qquad x(t_{k+1}) = \tilde{f}(x(t_k))\,, \quad \tilde{f}(0) = 0 \qquad (3.11b)$$

Assume that the function $V(x) \in \mathcal{C}^1$ is pos. def. in the domain $\mathcal{D}$:

$$V(x) > 0 \qquad \forall x \in \mathcal{D}/\{0\}, \quad V(0) = 0. \tag{3.12}$$

If $\forall x \in \mathcal{D}$ (eventually we require that $\exists\, \alpha_1, \alpha_2, \alpha_3,\, a > 1$)

$$\frac{\partial V}{\partial x} f(x) \le \begin{cases} 0 & \Rightarrow \text{stability} \\ -W_1(x)\,,\ W_1(x) \text{ pos. def. function} & \Rightarrow \text{asympt. stab.} \\ -\alpha_3 \|x\|^a\,,\ \alpha_1\|x\|^a \le V(x) \le \alpha_2 \|x\|^a\,,\ a > 1 & \Rightarrow \text{exp. stab.} \end{cases} \tag{3.13a}$$

In the discrete case we can relax the requirement for $V(x)$ to be $\mathcal{C}^1$ by demanding local Lipschitz continuouity ??, i.e. $\exists \alpha$ such that:

$$\|V(x) - V(y)\| \;\le\; \alpha\, \|x - y\| \qquad \forall x, y \in \mathcal{D}. \tag{3.14}$$

If we further replace the directional derivative $\frac{\partial V}{\partial x} f(x)$ in (3.13a) by
$\Delta V = V(\tilde{f}(x)) - V(x)$ we obtain the corresponding stability conditions for a discrete system.

Example 3.1: *Ljapunov equation*
We consider linear systems:

$$\dot{x}(t) = A\, x(t) \qquad\qquad x_{k+1} = \tilde{A}\, x_k$$

If $\forall Q = Q^T > 0 \quad \exists P = P^T > 0 \quad s.t.$

$$PA + A^T P = -Q \qquad (3.15a) \qquad\qquad \tilde{A}^T P \tilde{A} - P = -Q \qquad (3.15b)$$

then the linear systems are *asymptotically(?) stable*, which can be derived by choosing a quadratic Ljapunov function $V = x^T P x$. (3.15) are called Ljapunov equations.

- Controllability: A *linear* system (3.4) is controllable if

$\forall x(t) \in \mathbb{R}^n,\ t_1 > t$??	$\forall x(t_k) \in \mathbb{R}^n,\ l > 0$(??) there exist
$\exists u(\tau),\ \tau \in [t, t_1]$	$l > 0$ and $\{u_k, u_{k+1}, \dots, u_{k+l}\}$
s.t. $x(t_1) = 0$.	s.t. $x(t_{k+l}) = 0$.

Check via controllability matrix (in the discrete case replace A, B by $\tilde{A}, \tilde{B}$)

$$rank\,[B, AB, \dots, A^{n-1}B] = n \tag{3.16}$$

or Hautus test (preferably if the eigenvalues are already known)

$$rank\,[\lambda_i I - A \mid B] = n \quad \forall \lambda_i. \tag{3.17}$$

- Observability: A *linear* system (3.4) with output $y = Cx$ is observable if
$\forall x(t) \in \mathbb{R}^n,\ t_1 > t$ it is possible to determine the state trajectory $x(\tau),\ \tau \in [t, t_1]$
by observing the output trajectory $y(\tau)$ (i.e. the inner state of the system is uniquely defined if the output is known).

Check the observability matrix

$$rank \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix} = n \tag{3.18}$$

or use the Hautus test

$$rank \begin{bmatrix} \lambda_i I - A \\ B \end{bmatrix} = n \quad \forall \lambda_i \tag{3.19}$$

(similar for the discrete case).

- Stabilisability: All modes, which are not controllable, are stable.
$\rightsquigarrow$ Hautus test: if $rank[\lambda_i - A \mid B] < n$ then $Re(\lambda_i) \overset{!}{<} 0$ (discrete case: $|\lambda_i| \overset{!}{<} 1$).

Example 3.2: *Stabilisability*
Consider the system

$$\dot{x} = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix} x + \begin{bmatrix} B_1 \\ 0 \end{bmatrix} u. \tag{3.20}$$

The complete system is not controllable, but the subsystem $[A_1, B_1]$ is. So, if A_{22} is stable (i.e. the non controllable modes decay if we wait long enough, or at least do not expand)
$\Rightarrow$ System is stabilisable.

- Detectability: All modes, which are not observable, are stable.

Example 3.3: *Detectability*
Consider the system

$$\dot{x} = \begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} \end{bmatrix} x, \qquad y = [C_1\ 0]\, x. \tag{3.21}$$

To obtain detectability we have to demand that A_{22} is stable.

3.2 Boundary conditions, constraints and control variables

3.2.1 Boundary conditions

Often, the initial state $x(t_0)$ and the final state $x(t_f)$ of a process are fixed, or at least restricted on a certain region:

$$x(t_0) \in \mathcal{S}_0 \ \rightsquigarrow\ x(t_f) \in \mathcal{S}_f \qquad (3.22a) \qquad\qquad x_{k_0} \in \mathcal{S}_0 \ \rightsquigarrow\ x_{k_f} \in \mathcal{S}_f \qquad (3.22b)$$

Example 3.4: *Uncertain boundary conditions*
Consider a semi batch reactor with the reaction scheme: $A \rightarrow C \rightarrow B$.
C is the wanted product, its initial concentration is only approximately known: $c_C(t_0) \in \mathcal{S}_0$.
By controlling the concentration of the substrate A the final concentration of C shall be adjusted on a given intervall: $c_C(t_f) \in [c_{PL}, c_{PH}] = \mathcal{S}_f$.

In general, $\mathcal{S}_0$ and $\mathcal{S}_f$ are formulated as:

$$\mathcal{S} = \{x \in \mathbb{R}^n \,|\, s(x) \leq 0\} \tag{3.23}$$

Remarks:

- the vector valued function $s(x)$ defines a set of inequality constraints
- in the following we assume that $s(x)$ is smooth and all gradients ∇s_i are linearly independent
- typically the initial state $x(t_0), x_{k_0}$ is fixed and the terminal state $x(t_f), x_{k_f}$ is free

3.2.2 Control variables and state/input constraints

In optimal control we are able to consider restrictions to the variables describing and controlling a dynamic process in time. This is of great importance, because in reality almost every physical quantity is in some way limited. In the following, we distinguish between constraints restricting the input signal $u(t)$ and constraints with respect to the states $x(t)$. Finally, we introduce an *admissible control* as a pair of input and state trajectories which are satisfying the respective constraints.

- input constraints:

$$u(\tau) \in \mathcal{U}, \quad \tau \in [t_0, t_f] \qquad (3.24a) \qquad\qquad u_k \in \mathcal{U}, \quad k \in [k_0, k_f] \qquad (3.24b)$$

  - examples: valve saturation, limited force/power
  - often $\mathcal{U}$ is compact (i.e. closed and bounded)
  - in general: $\mathcal{U} = \mathcal{U}(x, \tau)$ (or $\mathcal{U} = \mathcal{U}(x_k, k)$)
  - typical: $\mathcal{U} = \{u \in \mathbb{R}^m \,|\, \underline{u} \leq u \leq \overline{u}\}$

- state constraints:

$$x(\tau) \in \mathcal{X}, \quad \tau \in [t_0, t_f] \qquad (3.25a) \qquad\qquad x_k \in \mathcal{X}, \quad k \in [k_0, k_f] \qquad (3.25b)$$

  - examples: safety constraints (max. temperature, max. acceleration, ...), product quality, hard physical constraints
  - often $\mathcal{X}$ is simply connected
  - $\mathcal{X}$ must be consistent with $\mathcal{S}_0, \mathcal{S}_f$: $\mathcal{S}_0, \mathcal{S}_f \subseteq \mathcal{X}$
  - in general: $\mathcal{X} = \mathcal{X}(\tau)$ (or $\mathcal{X} = \mathcal{X}(k)$)

- admissible control:

Definition 3.1 (Admissibility) *A tuple $(u(\cdot), x(\cdot))$ is admissible if $u \in \mathcal{U}$ and $\forall x(t_0) \in \mathcal{S}_0$ the input $u(\cdot)$ makes $x(\cdot) \in \mathcal{X}$ and $x(t_f) \in \mathcal{S}_f$. $u(\cdot)$ is then called an admissible control.*

Remarks:

- similar for the discrete case ...
- $u(\cdot)$ is used as short notation for a whole trajectory: $u(\tau)$, $\tau \in [t_0, t_f]$, equivalently in the discrete case $\{u\}$ is the short hand form for: $\{u_{k_0}, \dots, u_{k_f}\}$
- $x(\,\cdot\,; x(t_0), u(\cdot))$ may be used in the following for the solution of $\dot{x} = f(t, x, u)$ with initial state $x(t_0)$ under excitation of $u(\cdot)$
- in the following we assume that $u(\cdot)$ is *piecewise continuous*
- admissibility can be regarded as a question of constrained "controllability/reachability"

Example 3.5: *Admissible control for a linear system*
Consider a linear system with fixed boundary conditions

$$\dot{x} = Ax + Bu, \qquad x(t_0 = 0) = x_0, \qquad x(t_f) \stackrel{!}{=} 0. \tag{3.26}$$

There are infinite many solutions to this problem (if the system is controllable):

$$x(t_f) = e^{At_f} x_0 + \int_0^{t_f} e^{A\tau} Bu \, d\tau \stackrel{!}{=} 0 \tag{3.27}$$

$$\dots \Rightarrow \; u(t) = -B^T e^{At} W^{-1}(t)\, e^{At} x_0 + \tilde{u}(t) \quad \text{with} \quad W(t) = \int_0^t e^{A\tau} BB^T (e^{A\tau})^T \, d\tau \quad \text{and } \tilde{u} \text{ such that } \int_0^t e^{A\tau} B\tilde{u} \, d\tau = 0 \tag{3.28}$$

$W(t)$ is connected to the controllability of the system: if $W(t)$ is not invertible, then the system is not controllable. Moreover, if there were constraints on the input which would be violated by each possible solution $u(t)$ in (3.28) then again no admissible control exists.

3.2.3 Cost function

The general formulation of the cost function in optimal control problem reads

$$J(u(\cdot), x(\cdot)) = \int_{t_0}^{t_f} F(t, x, u) \, d\tau + E(x(t_f)) \tag{3.29a}$$

$$J(\{u\}, \{x\}) = \underbrace{\sum_{k=k_0}^{k_f} F(k, x_k, u_k)}_{\text{stage/integral cost}} + \underbrace{E(x_{k_f})}_{\text{terminal penalty}} \,. \tag{3.29b}$$

Remarks:

- the use of a cost function allows to consider "economic" side constraints

- one unique input trajectory $u^*(t)$ of the set of all admissible inputs is selected
- we assume in the following that $F, E \in \mathcal{C}^1$ and often that $F > 0 \quad \forall x, u \neq x_s, u_s$
- examples of specific problems:
  - ⋆ regulation problem: F, E penalize deviations from the steady state (x_s, u_s) ($\rightsquigarrow$ LQR: $J = \int_0^\infty x^T Q x + u^T R u \, d\tau$)
  - ⋆ tracking problem: F, E penalizes deviation from *reference trajectory*
  - ⋆ minimum time problem: t_f, k_f free, $F = 1 \quad \rightsquigarrow \quad J = \int_{t_0}^{t_f} d\tau = T_f$
  - ⋆ minimal control energy: $F = F(u)$

3.2.4 Formal statement of optimal control problems

Find an admissible control $u(\cdot) \in \mathcal{U}$ s.t. for $x(t_0) \in \mathcal{S}_0$ the final state reaches a specified region $x(t_f) \in \mathcal{S}_f$ and the cost $J(u(\cdot), x(\cdot))$ is minimized (similarly for the discrete case):

$$\min_{u(\cdot)} \int_{t_0}^{t_f} F(t,x,u)\, d\tau + E(x(t_f)) \tag{3.30a}$$

$$\begin{aligned} s.t. \quad & \dot{x} = f(t,x,u), \quad x(t_0) \in \mathcal{S}_0 \\ & u(\tau) \in \mathcal{U},\ x(\tau) \in \mathcal{X} \quad (\tau \in [t_0, t_f]) \\ & x(t_f) \in \mathcal{S}_f \\ & u \text{ piecewise continuous} \end{aligned}$$

$$\min_{\{u_k\}} \sum_{k=k_0}^{k_f} F(k, x_k, u_k) + E(x_{k_f}) \tag{3.30b}$$

$$\begin{aligned} s.t. \quad & x_{k+1} = \tilde{f}(t, x_k, u_k), \quad x_{k_0} \in \mathcal{S}_0 \\ & u_k \in \mathcal{U},\ x_k \in \mathcal{X} \quad (k \in [k_0, k_f]) \\ & x_{k_f} \in \mathcal{S}_f \end{aligned}$$

Remarks:

- t_f, k_f can be fixed or free: $t_f > t_0$, $k_f > k_0$, if runtime is free introduce t_f (k_f) as additional minimization parameter (e.g. $\min\limits_{u(\cdot), t_f} \ldots$)
- t_0 is fixed, $\mathcal{S}_0, \mathcal{S}_f$ are optional
- the continuos time problem (3.30a) is *infinite dimensional* (find $u(\cdot)$ as a function of time) while the discrete time problem (3.30b) is *finite dimensional*, provided k_f is finite (find a sequence $\{u_k\}$)
- if t_f $(k_f) \to \infty$ we obtain an *infinite horizon problem*
- if the dynamic system depends on parameters p which we also want to be optimized:
  $$\dot{x} = f(t,x,u,p), \quad p = const. \tag{3.31}$$
  we can add a constraint $\frac{dp}{dt} = 0$ and extend the optimization on these parameters:
  $$\min_{u(\cdot), p} J(u(\cdot), x(\cdot)) \quad s.t. \ \ldots \ p \in \mathcal{S}_p \tag{3.32}$$
- an admissible pair $(x^*(\cdot), u^*(\cdot))$ is optimal if
  $$J(x^*(\cdot), u^*(\cdot)) \leq J(x(\cdot), u(\cdot)) \qquad \text{for } \textit{all} \text{ admissible } (x(\cdot), u(\cdot)) \tag{3.33}$$

The formulation of an optimal control problem using a cost function as in (3.30) is known as Bolza problem. However, there are further formulations which are equivalent:

$$J = \underbrace{\int_{t_0}^{t_f} F(t,x,u)\, d\tau}_{\substack{\text{only integral term:} \\ \text{Lagrange problem}}} + \underbrace{E(x(t_f))}_{\substack{\text{only penalty term:} \\ \text{Mayer problem}}} . \tag{3.34}$$

We can reformulate the Bolza problem as

- pure Mayer problem: introduce a further state $x_0 : \frac{dx_0}{dt} = F(t,x,u)$ and redefine the cost $\tilde{J} = F(x(t_f)) + x_0(t_f)$
- pure Lagrange problem: introduce a further state $x_0 : \frac{dx_0}{dt} = 0,\ x_0(0) = \frac{E(x(t_0+T_f))}{T_f}$

Moreover, a problem with free runtime T_f can be transformed into a problem with fixed (normalized) integration bounds by introducing T_f as a parameter, which is initializing a constant state T:

$$\begin{aligned} t &= T \cdot \tau \quad (\tau \in [0,1]) \\ \frac{dT}{dt} &= 0, \quad T(0) = T_f \in \mathcal{S}_{T_f} \end{aligned} \qquad \rightsquigarrow \qquad \min_{u(\cdot), T_f} \int_0^1 \ldots\, d\tau. \tag{3.35}$$

3.2.5 Well posedness, existence and uniqueness

- Controllability/Solvability
  If no admissible control exists (i.e. an input trajectory $u(\cdot)$ bringing $x(t_0) \in \mathcal{S}_0 \rightsquigarrow x(t_f) \in \mathcal{S}_f, x(\cdot) \in \mathcal{X}$) we define the cost:
  $$J := \infty. \tag{3.36}$$

Example 3.6: *Non feasible linear system*

$$\dot{x} = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} x + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u, \qquad T_f \text{ fixed}, x(t_0 + T_f) = \mathcal{S}_f = 0 \tag{3.37}$$

Note that system (3.37) is not controllable and though the non controllable mode is stable there exists (i.g.) no feasible solution (satisfying the terminal constraint for a fixed and therefore *finite* T_f).

Example 3.7: *Controllable linear system*

$$\min_u \int_{t_0}^{t_0+T_f} u^2\, d\tau \quad s.t. \quad \dot{x} = Ax + Bu, \quad [A,B] \text{ controllable} \\ x(t_0) = x_0, \quad x(t_0 + T_f) = 0 \tag{3.38}$$

Now we have assumed that the linear system is controllable: therefore problem (3.38) always has a solution.

- Well definedness of differential equations (DE)

  A differential equation $\dot{x} = f(t, x, u)$ is well defined over $[t_0, t_f]$ if for a given $u(\cdot) \in \mathcal{U}$ a solution *exists* and also is *unique.* (??)

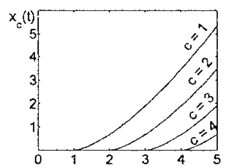
  Example 3.8: *Non unique solutions*

  Consider the differential equation

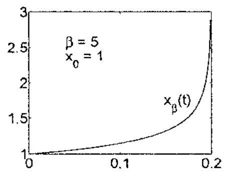
$$\dot{x} = x^{\frac{1}{3}}, \quad x(0) = 0. \tag{3.39}$$

There is an infinite number of feasible solutions, which are below characterized by a parameter c (see also Figure 3.2(a)):

$$x_c(t) = \begin{cases} 0 & \text{for } t \leq c \\ \frac{2}{3}(t-c)^{\frac{3}{2}} & \text{for } t > c \end{cases} \tag{3.40}$$



(a) Solution trajectories $x_c(t)$ of Example 3.8.



(b) Solution $x_\beta(t)$ of Example 3.9: finite escape time $t_{max} = 0.2$.

Figure 3.2: Solution plots for Example 3.8 and 3.9 respectively.

Example 3.9: *Finite escape time*

We consider the system

$$\dot{x} = x^\alpha + u, \quad x(0) = x_0 > 0. \tag{3.41a}$$

The solution for the autonome system ($u = 0$) reads:

$$x_\beta(t) = \frac{x_0}{(1 - \beta x_0^\beta t)^{1/\beta}}, \quad \beta := \alpha - 1 \tag{3.41b}$$

and is plotted in Figure 3.2(b). Note that the solution tends to infinity for a finite time t_{max} and that for $t > t_{max}$ no well defined (real) solution exists.

To make general statements about the existence and uniqueness of a solution for an ordinary differential equation (ODE) it is useful to assume a certain degree of continuity. Therefore, we consider an autonome DE $\dot{x} = f(t, x)$ and define

Definition 3.2 (Lipschitz continuity) *Let $\Omega \subset \mathbb{R}^n$, then $f(t, x)$ is Lipschitz on $[t_0, t_1] \times \Omega$ if $\exists L > 0$ s.t.*

$$\|f(t, x_1) - f(t, x_2)\| \leq L\|x_1 - x_2\| \qquad \forall x_1, x_2 \in \Omega,\ t \in [t_0, t_1]. \tag{3.42}$$

Note that

- a vector valued function f is Lipschitz $\Leftrightarrow$ all components f_i are Lipschitz
- if $f(t, x) \in C^0$ and $\frac{\partial^2 f}{\partial x^2}(t, x) \in C^0$ in a compact set $[t_0, t_1] \times \Omega \Rightarrow f$ is Lipschitz on this set

Theorem 3.1 (Local existence and uniqueness) *Assume that f is* piecewise continuous *in t on $[t_0, t_1]$ and Lipschitz in x on a set $[t_0, t_1] \times \Omega$, Ω open and connected. Then there exists $\delta > 0$, s.t.*

$$\dot{x} = f(t, x), \quad x(t_0) = x_0 \in \Omega \tag{3.43}$$

has a unique solution for $t \in [t_0, t_0 + \delta]$.

Theorem 3.2 (Global existence and uniqueness) *Assume that f is globally Lipschitz in x on a set $[t_0, t_1] \times \mathbb{R}^n$ and that f is bounded: $\|f(t, x)\| \leq M$ for $t \in [t_0, t_1]$, $M = const.$. Then (3.43) has a unique solution for $t \in [t_0, t_1]$, $x(0) \in \mathbb{R}^n$.*

- Existence of *optimal* solutions

  Even if admissible solutions $(u(\cdot), x(\cdot))$ exist, it is not guaranteed that there is also an optimal one.

  Example 3.10: *Rocket car no. 4 - no optimal solution*

  We again want to bring a rocket car from $x(0) = z_0$ to the origin $x(t_f) = 0$ with a minimal amount of energy, but the runtime t_f is now let completely free:

$$J = \int_0^{t_f} u^2 d\tau, \qquad x(0) = z_0,\ x(t_f) = 0,\ t_f \text{ free.} \tag{3.44}$$

Doing some calculations (??) one obtains the relation $J(t_f) = 12\frac{x_0^2}{t_f^3}$, i.e. the total cost decreases with increasing time. For t_f tending to infinity the cost vanishes! If we want to obtain an explicit optimal solution we have to restrict the runtime by including a corresponding constraint.

In general, after stating an optimal control problem one should check whether:

- an admissible control $u(\cdot)$ exists at all and whether
- a solution to the optimal control problem exists.

An optimal control problem is *well posed* if a solution *exists* and is *unique*.

3.2.6 Some existence and uniqueness results

In the following we consider some special cases which allow a quite simple verification of the existence and/or uniqueness of an optimal solution in advance, provided the corresponding assumptions are satisfied.

1. *Fixed end state (terminal constraint)*

$$\begin{aligned} \min_{u(\cdot)} J &= \int_{t_0}^{t_f} F(t,x,u)\,d\tau + E(x(t_f)) \\ s.t.\quad \dot{x} &= f(t,x,u) \qquad x(t_0) = x_0,\ \mathbf{x}(\mathbf{t_f}) \in S_f \\ & \qquad\qquad\qquad\quad u(\cdot) \in \mathcal{U} \end{aligned} \tag{3.45}$$

Assumptions:

(a) $f, F, E \in C^0$

(b) "controllability": the set of admissible controls is not empty and for all admissible controls the states are bounded:

$$\|x(t, x_0, u(\cdot))\| \leq \alpha \qquad \forall t \in [t_0, t_f], \quad \alpha = const. \tag{3.46}$$

(c) $\mathcal{U}$ is compact (i.e. closed and bounded)

(d) "convexity": the set

$$\mathcal{V} = \mathcal{V}(t,x,\mathcal{U}) = \{v \in \mathcal{U} \,|\, \underbrace{(F(t,x,v), f(t,x,v))}_{\text{extended velocity vector}}\} \subset \mathbb{R}^{1+n} \quad \text{is convex } \forall\, t, x \tag{3.47}$$

If these conditions hold then there exists an *unique (??)* optimal solution.

Example 3.11: *Uniqueness conditions - case 1*
Consider the minimization problem

$$\begin{aligned} \min_{u(\cdot)} J &= \int_0^{t_f} |u|^{0.5}\, x\, d\tau \qquad (x, u \in \mathbb{R}) \\ s.t.\quad \dot{x} &= |u|^{0.5}, \qquad x(0) = 0(??), |u| \leq 1. \end{aligned} \tag{3.48}$$

The functions $|u|^{0.5}$ and $|u|^{0.5}x$ are both continuous (a), we assume that there exists an admissible control fulfilling (b) and the input set $\mathcal{U}$ defined by $|u| \leq 1$ is compact (c). We only have to verify the convexity of the sets:

$$\mathcal{V} = \{-1 \leq v \leq 1 \,|\, (|v|^{0.5}x, |v|^{0.5})\} \qquad \forall x \geq 0. \tag{3.49}$$

As shown in Figure 3.3 these sets $\mathcal{V}(x)$ are just lines which are certainly convex. Therefore the upper assumptions are fulfilled and the existence of an optimal solution can be guaranteed.

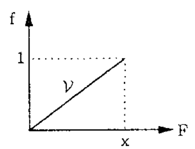


Figure 3.3: Plot of the extended velocity set V of Example 3.11.

2. *Systems which are affine in u*

$$\begin{aligned} \min_{u(\cdot)} J &= \int_{t_0}^{t_f} F_1(x) + F_2(u)\,d\tau + E(x(t_f)) \\ s.t.\quad \dot{x} &= f(x) + B(x)\,u, \qquad x(t_0) = x_0 \end{aligned} \tag{3.50}$$

Assumptions:

(a) no input constraints and no state constraints

(b) $T_f = t_f - t_0$ is fixed

(c) $f, B \in C^1$,
$F_2(u)$ strictly convex and $t \to \infty$ for $\|u\| \to \infty$,
$F_1, E > 0$ and convex

$\Rightarrow$ there exists an *unique* optimal solution

3. *Nonlinear system with input constraints*

$$\begin{aligned} \min_{u(\cdot)} J &= \int_{t_0}^{t_f} F(x,u)\,d\tau + E(x(t_f)) \\ s.t.\quad \dot{x} &= f(x,u), \qquad x(t_0) = x_0 \\ u(\cdot) &\in \mathcal{U} \end{aligned} \tag{3.51}$$

Assumptions:

(a) T_f fixed

(b) $F, E \in C^1$ and $> 0(??)$

(c) $\mathcal{U}$ is convex and compact

(d) there exists a solution of $\dot{x} = f(x,u) \quad \forall u(\cdot) \in \mathcal{U}$

$\Rightarrow$ there exists an optimal solution

3.3 Outlook

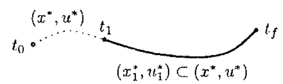
In the end of this chapter a brief outlook on the following chapters is presented, dealing with two different approaches of finding an optimal control $u^*(\cdot)$.

4 Dynamic Programming (DP)

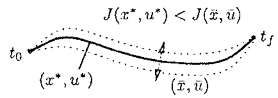
- based on Bellmann's Principle of Optimality (cmp. Figure 3.4(a))
- delivers feedback $u^* = k(t, x)$
- in certain setups $\rightsquigarrow$ *sufficient* conditions
- involves the solution to a PDE, the Hamilton-Jacobi-Bellmann equation
- some smoothness assumptions have to hold

5 Pontryagin Minimum Principle (PMP)

- based on local considerations *around* an optimal trajectory (see Figure 3.4(b))
- often only delivers $u^* = u^*(t)$
- only *necessary* conditions $\rightsquigarrow$ "candidates"
- conditions are easier to check (compared to those of DP)
- might work if DP fails



(a) Principle of Optimality: "endpieces of optimal trajectories must be optimal".



(b) PMP: consider first order variations around (x^*, u^*) $\rightsquigarrow$ "neighbour candidates must be worse".

Figure 3.4: Concepts used for deriving conditions for optimal trajectories.