## 2.10. Numerical methods

Our aim is to introduce a few fundamental concepts for the numerical solution of linear-quadratic elliptic optimal control problems. We will usually consider the algorithms for the continuous problem, the algorithms still need to be discretised (e.g. using finite differences, finite elements) for the practical implementation.

### Conditioned gradient method

Consider the following optimisation problem:
Let $U$ be a Hilbert space, $U_{ad} \subset U$ a non-empty, bounded, convex and closed subset, $f : U \to \mathbb{R}$ a Gâteaux-differentiable functional. Find

$$\min_{u \in U_{ad}} f(u).$$

The conditioned gradient method is an iterative algorithm to solve this problem, it is given as follows:

0) Choose an initial iterate $u_0 \in U_{ad}$, set $n = 0$.

1) Determine direction $v_n$ as solution of the optimisation problem: $f'(u_n) v_n = \min_{v \in U_{ad}} f'(u_n) v$ $(*)$
("Determination of new descent direction")
If $f'(u_n)(v_n - u_n) \geq 0 \Rightarrow$ STOP, $u_n$ satisfies variational inequality, $u_n$ is optimal.
Otherwise, $f'(u_n)(v_n - u_n) < 0$, i.e. $(v_n - u_n)$ is a descent direction.

2) Determine $s_n \in (0,1]$ from $f(u_n + s_n(v_n - u_n)) = \min_{s \in (0,1]} f(u_n + s(v_n - u_n))$.
Set $u_{n+1} := u_n + s_n(v_n - u_n)$, $n := n+1$, go to 1). ("Line search/step size control")

**Remarks:** i) Optimisation problem $(\ast)$ admits a solution in view of the assumptions.

ii) For convex $f$, the sequence $(f(u_n))_n$ converges monotonically to the optimal solution.

iii) $u_n, v_n \in U_{ad} \implies u_{n+1} = u_n + s_n(v_n - u_n) \in U_{ad}$, i.e. all iterates are admissible.

**Example:** Consider the optimal stationary heating problem:

$$\min J(y, u) := \frac{1}{2}\|y - y_\Omega\|^2_{L^2(\Omega)} + \frac{\lambda}{2}\|u\|^2_{L^2(u)}$$

s.t. $\quad -\Delta y = \beta u$ in $\Omega$, $y = 0$ on $\partial\Omega \qquad (\ast\ast)$

and $\quad u_a(x) \le u(x) \le u_b(x)$ a.e. in $\Omega$.

Let $U = L^2(\Omega)$, $U_{ad} = \{u \in L^2(\Omega) : u_a(x) \le u(x) \le u_b(x) \text{ a.e.}\}$.
Consider the reduced functional $f(u) := J(y(u), u)$,
then:

$$f'(u_n)v = \int_\Omega (\beta p_n + \lambda u_n)v \, dx,$$

where $p_n$ is the solution of the adjoint equation

$$(\ast\ast\ast)\begin{cases} -\Delta p_n = y_n - y_\Omega & \text{in } \Omega \\ p_n = 0 & \text{on } \partial\Omega \end{cases}$$

The conditioned gradient method takes the form:

0) Choose initial iterate $u_0 \in U_{ad}$, $n := 0$.

1) Solve (for given $u = u_n$) $(\ast\ast) \longrightarrow y =: y_n$.
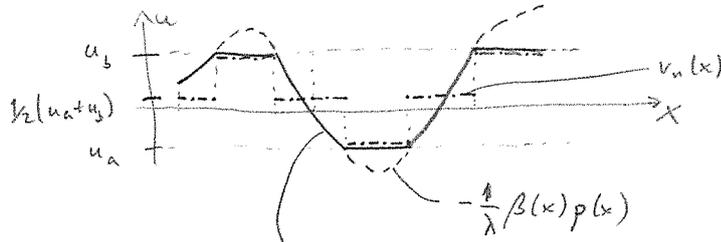
2) Solve (for given $y_n$) $(\ast\ast\ast) \longrightarrow p_n$.

3) Determine $v_n$ from: $\min\limits_{v \in U_{ad}} \int (\beta p_n + \lambda u_n)v \, dx$.
   If $\|v_n - u_n\|_U < \varepsilon$ ($\varepsilon$: given tolerance), then STOP.

4) Determine $s_n \in (0, 1]$ from: $f(u_n + s_n(v_n - u_n)) = \min\limits_{s \in (0,1]}\left(f(u_n + s(u_n - v_n))\right)$
   Set $u_{n+1} := u_n + s_n(v_n - u_n)$, $n := n+1$, go to 1).

Remarks: i) Realisation of step 3):

It follows from the weak minimum principle that a new direction $v_n$ is given by

$$v_n(x) := \begin{cases} u_a(x), & \lambda u_n(x) + \beta(x) p_n(x) > 0, \\ \frac{1}{2}(u_a + u_b)(x), & \lambda u_n(x) + \beta(x) p_n(x) = 0, \\ u_b(x), & \lambda u_n(x) + \beta(x) p_n(x) < 0. \end{cases}$$



$$\bar{u}(x) = \mathbb{P}_{[u_a(x), u_b(x)]}\left(-\frac{1}{\lambda}\beta(x)p(x)\right)$$

Note: The second case is unlikely to occur in practice.

ii) Realisation of step 4):
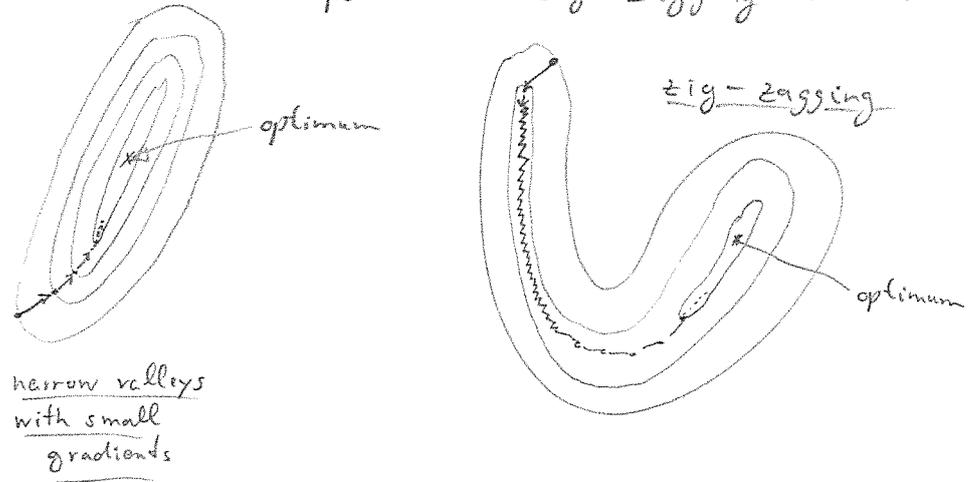
Let $y_n := y(u_n)$, $w_n := y(v_n)$.

Define $g(s) := f(u_n + s(v_n - u_n))$

$$= \frac{1}{2}\|y_n + s(w_n - y_n) - y_\Omega\|^2_{L^2(\Omega)} + \frac{\lambda}{2}\|u_n + s(v_n - u_n)\|^2_{L^2(\Omega)}$$

$$= \frac{1}{2}\|y_n - y_\Omega\|^2_{L^2(\Omega)} + s(y_n - y_\Omega, w_n - y_n)_{L^2(\Omega)} + \frac{s^2}{2}\|w_n - y_n\|^2_{L^2(\Omega)}$$

$$+ \frac{\lambda}{2}\|u_n\|^2_{L^2(\Omega)} + \lambda s(u_n, v_n - u_n)_{L^2(\Omega)} + \frac{\lambda s^2}{2}\|v_n - u_n\|^2_{L^2(\Omega)}$$

$$= g_0 + g_1 s + g_2 s^2,$$

where $g_0, g_1, g_2$ are constants which can be computed easily ($u_n, y_n, v_n$ are known, for $w_n = y(v_n)$, problem (**) has to be solved).

Hence, $s_n := \arg\min_{s \in (0,1]} g(s)$ can be found easily as

the projection of the zero of $g'(s)$ onto $[0,1]$, it is called the 'exact step size'.

(ii) The conditional gradient method usually shows
fast convergence initially, but then becomes slow.
The reason for this being that the gradient often
becomes small near the optimum, the method
exhibits small steps or a 'zig-zagging' course.



optimum

zig-zagging

optimum

narrow valleys
with small
gradients

iv) The method is quite easy to implement. Since
steps 3) & 4) can be solved analytically, only
to elliptic PDE problems have to be solved in
each iteration. It can hence be useful for first
tests. A better method is the so-called
'projected gradient method' which we
explain in the following.

Projected gradient method

Replace steps 3) and 4) in the conditioned gradient method by

3') Choose the anti-gradient as descent direction,
$$v_n = -f'(u_n) = -(\beta p_n + \lambda u_n).$$
If $\|v_n\|_U < \varepsilon$, STOP.
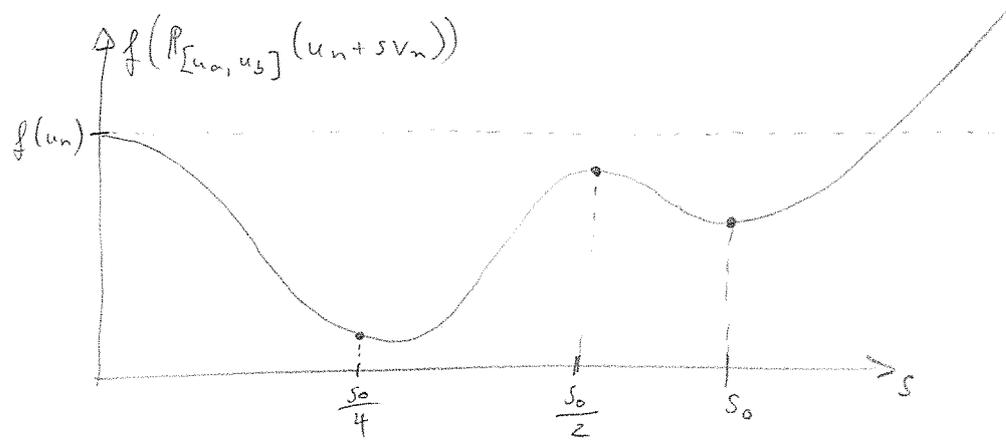
and 4') Determine the step size $s_n$ by
$$f\left(\mathbb{P}_{[u_a, u_b]}(u_n + s_n v_n)\right) = \min_{s>0} f\left(\mathbb{P}_{[u_a, u_b]}(u_n + s v_n)\right).$$
Set $u_{n+1} := \mathbb{P}_{[u_a, u_b]}(u_n + s_n v_n)$, $n := n+1$, go to 1).

Remarks: i) The projection is necessary to guarantee admissibility.

ii) In the linear-quadratic case studied here, the step size in 4') can be determined as the exact step size. If this is not possible, one can use suitable methods of step size control. An example is:

Bisection method: Starting from a suitable small step size $s_0$ (e.g. the step size from the previous iteration), employ successively $s = \frac{s_0}{2^i}$ ($i = 1, 2, 3, 4, ...$), until an $s$ is found such that $f(P_{[u_a, u_b]}(u_n + s v_n))$ is sufficiently smaller than the previous $f(u_n)$. Otherwise, terminate after a maximum, prescribed number of bisections.

Another known method is Armijo's rule from the theory of nonlinear optimisation.



Bisection method

# Transformation into a finite-dimensional quadratic optimisation problem

**Idea:** "Discretise, then optimise"

Consider once more for $\Omega \subset \mathbb{R}^2$:

$$\min J(y,u) := \frac{1}{2} \| y - y_\Omega \|^2_{L^2(\Omega)} + \frac{\lambda}{2} \| u \|^2_{L^2(\Omega)}$$
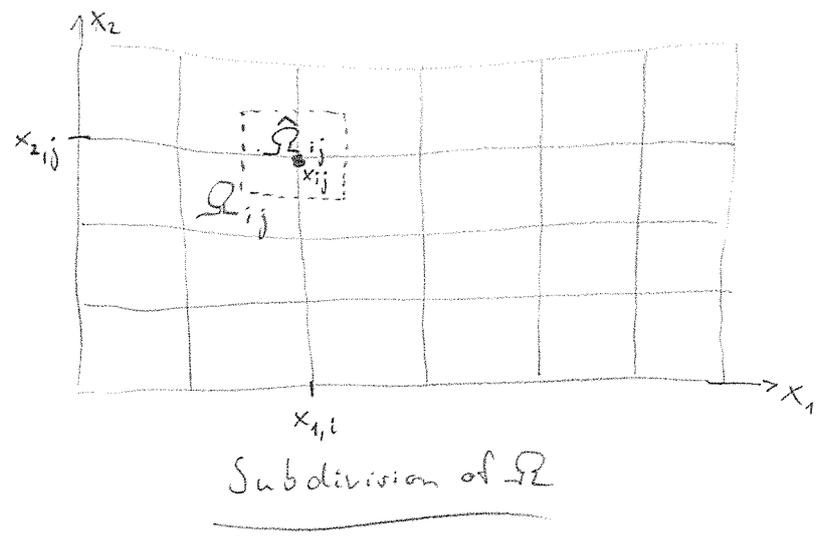
$$\text{s.t.} \quad -\Delta y = \beta u \text{ in } \Omega, \quad y = 0 \text{ on } \partial\Omega$$

$$\text{and} \quad u \in U_{ad} = \{ u \in L^2(\Omega) : u_a(x) \leq u(x) \leq u_b(x) \text{ a.e.} \}$$

We now discretise this problem using finite differences (chosen here for simplicity), and then obtain a <u>discrete optimisation</u> problem.

To this end, let $\Omega \subset \mathbb{R}^2$ be decomposed into squares $\Omega_{ij}$ with corners $x_{ij}$ and side length $h$. Let us assume for the sake of ease of presentation that $x_{ij} = h \binom{i}{j}$, $i,j = 0, \ldots, n$, and let $y_{ij} := y(x_{ij}) = y(x_i, x_j)$. One can see (using Taylor's theorem)

$$-\Delta y(x_{ij}) \approx \frac{4 y_{ij} - (y_{i-1,j} + y_{i,j-1} + y_{i+1,j} + y_{i,j+1})}{h^2}, \quad i,j = 1, \ldots, n-1.$$

Define new squares $\hat{\Omega}_{ij}$ with center $x_{ij}$ and side length $h$. Let $u \approx u_{ij} \equiv$ constant on $\hat{\Omega}_{ij}$, $i,j = 1, \ldots, n-1$. On the boundary, set $u = 0$.



Subdivision of $\Omega$

Now, numbering the quantities $x_{ij}$, $y_{ij}$ and $u_{ij}$ lexicographically, i.e. from the lower left to upper right corner of $\Omega$, we obtain vectors $x = (x_{11}, \dots, x_{n-1,n-1})^T$, $y = (y_{11}, \dots, y_{n-1,n-1})^T$ and $u = (u_1, \dots, u_{n-1,n-1})^T$. Moreover, we define vectors $y_\Omega$, $u_a$, $u_b$ by setting $y_{\Omega,ij} = y_\Omega(x_{ij})$, $u_{a,i,j} = u_a(x_{ij})$, $u_{b,i,j} = u_b(x_{ij})$.

We approximate the integrals appearing in the cost functional by midpoint rules, e.g. $\int_{\hat{\Omega}_{ij}} y(x_{ij}) \, dx \approx h^2 y_{ij}$, and divide the cost functional by $h^2$. We obtain a problem of the form

$$\min \frac{1}{2} \sum_{i,j=1}^{n-1} \left[ (y_{ij} - y_{\Omega,ij})^2 + \frac{\lambda}{2} u_{ij}^2 \right]$$

$$s.t. \quad Ay = Bu$$

$$and \quad u_a \leq u \leq u_b$$

with suitable matrices $A, B \in \mathbb{R}^{(n-1) \times (n-1)}$.

This is a quadratic optimisation problem with linear equality and inequality constraints, and can be solved using existing methods for large scale optimisation (e.g. using quadprog in MATLAB).

# Primal-dual active set strategy

Consider the optimal stationary heating problem as above, and set $\beta \equiv 1$ for simplicity.

We know the optimal control has to satisfy

$$(*) \quad \bar{u}(x) = \mathbb{P}_{[u_a(x), u_b(x)]} \left\{ -\frac{1}{\lambda} p(x) \right\},$$

where $p$ is the adjoint state.

Now setting $\mu := -\left( \frac{1}{\lambda} p + \bar{u} \right) = -\frac{1}{\lambda} f'(\bar{u})$, we obtain the relation which is satisfied by $\bar{u}$:

$$\bar{u}(x) = \begin{cases} u_a(x), & \text{if } \mu(x) < 0 \iff -\frac{1}{\lambda} p(x) < u_a(x), \\ -\frac{1}{\lambda} p(x), & \text{if } \mu(x) = 0 \iff -\frac{1}{\lambda} p(x) \in [u_a(x), u_b(x)], \\ u_b(x), & \text{if } \mu(x) > 0 \iff -\frac{1}{\lambda} p(x) > u_b(x). \end{cases}$$

In the first case, $\mu(x) < 0$ and hence $\bar{u}(x) + \mu(x) < u_a(x)$, since $\bar{u} = u_a$.
In the third case, $\mu(x) > 0$ and hence $\bar{u}(x) + \mu(x) > u_b(x)$, since $\bar{u} = u_b$.
In the second case, $\mu(x) = 0$, and hence $\bar{u}(x) + \mu(x) = -\frac{1}{\lambda} p(x) \in [u_a(x), u_b(x)]$.

$$\Rightarrow u = \bar{u} \text{ satisfies } u(x) = \begin{cases} u_a(x), & \text{if } u(x) + \mu(x) < u_a(x), \\ -\frac{1}{\lambda} p(x), & \text{if } u(x) + \mu(x) \in [u_a(x), u_b(x)], \\ u_b(x), & \text{if } u(x) + \mu(x) > u_b(x). \end{cases} \quad (**)$$

Conversely, if $u \in U_{ad}$ satisfies $(**)$, then $u$ satisfies the projection condition and is therefore optimal (e.g. in the first case, $u = u_a \Rightarrow \mu(x) < 0 \Rightarrow 0 > -\frac{1}{\lambda} p - u = -\frac{1}{\lambda} p - u_a \Rightarrow -\frac{1}{\lambda} p < u_a$, hence $u(x) = \mathbb{P}_{[u_a(x), u_b(x)]} \left\{ -\frac{1}{\lambda} p(x) \right\}$).

Summarizing, the quantity $u + \mu$ is an indicator whether the inequality constraints are active or not.

This motivates the following algorithm.

0) Initialize $u_0, \mu_0 \in L^2(\Omega)$ ($u_0$ not necessarily admissible), $n := 1$

1) Set $A_n^+ = \{x: u_{n-1}(x) + \mu_{n-1}(x) > u_b(x)\}$

$A_n^- = \{x: u_{n-1}(x) + \mu_{n-1}(x) < u_a(x)\}$

$I_n = \Omega \setminus \{A_n^- \cup A_n^+\}$

If $n \geq 2$ and $A_n^+ = A_{n-1}^+$ and $A_n^- = A_{n-1}^-$, then STOP, since iterate is optimal.

2) Solve for $y, p \in H_0^1(\Omega)$ the system

$$-\Delta y = u$$

$$-\Delta p = y - y_\Omega$$

$$u = \begin{cases} u_a & \text{on } A_n^- \\ -\frac{1}{\lambda} p & \text{on } I_n \\ u_b & \text{on } A_n^+ \end{cases}$$

(Note: this is the optimality system of the linear-quadratic optimal control problem as above, where additionally the control $u$ is fixed on $A_n^+, A_n^-$.)

Set $u_n := u$, $p_n := p$, $\mu_n := -(\frac{1}{\lambda} p_n + u_n)$, $n := n+1$, go to step 1).

Remarks: (i) The system in step 2) can be written as

$$\begin{pmatrix} 0 & -\Delta & -1 \\ -\Delta & -1 & 0 \\ (1 - \chi_{A_n^-} - \chi_{A_n^+}) \frac{1}{\lambda} & 0 & +1 \end{pmatrix} \begin{pmatrix} p \\ y \\ u \end{pmatrix} = \begin{pmatrix} 0 \\ -y_\Omega \\ \chi_{A_n^-} u_a + \chi_{A_n^+} u_b \end{pmatrix}$$

where $\chi_{A_n^-}, \chi_{A_n^+}$ are the characteristic functions of $A_n^-, A_n^+$.

(ii) One can replace $\mu$ by $c\mu$ with a constant $c > 0$ in (**), and work with $c\mu_{n-1}$ in place of $\mu_{n-1}$ in step 1), which can be beneficial in numerical computations.

(iii) The method can be interpreted as a "semi-smooth Newton method" which explains that it usually converges superlinearly.

Another approach is to directly solve the non-smooth optimality system:

$$-\Delta y = P_{[u_a, u_b]}\{-\frac{1}{\lambda} p\}, \quad y|_{\partial\Omega} = 0 \text{ and } -\Delta p = y - y_\Omega, \quad p|_{\partial\Omega} = 0.$$