Nonlinear Programming 2

- Unconstrained Optimization 1: line search, search interval, extreme values of convex functions, smooth and nonsmooth line search
- Unconstrained Optimization 2: multivariable unconstrained optimization, steepest descent method
- Solving a constrained nonlinear program using unconstrained optimization: exterior penalty function methods, (interior) barrier function methods, choosing penalty and barrier functions, comparison, examples

Recall: Feasible Directions

- Given a nonlinear program $\min f(\boldsymbol{x}) : \boldsymbol{x} \in X$, where $X = \left\{\boldsymbol{x} : g_i(\boldsymbol{x}) \leq 0, i \in \{1, \ldots, m\}\right\}$ and f and g_i are smooth (continuously differentiable) functions on X
- We use the fact that if $\bar{\boldsymbol{x}}$ is a local minimum then there is no **(improving) feasible direction** $\boldsymbol{d}$ at $\bar{\boldsymbol{x}}$ and $\delta > 0$ so that

$$\forall \lambda \in (0, \delta) : \quad \begin{aligned} f(\bar{\boldsymbol{x}} + \lambda \boldsymbol{d}) &< f(\bar{\boldsymbol{x}}) \\ \bar{\boldsymbol{x}} + \lambda \boldsymbol{d} &\in X \end{aligned}$$

- Only necessary, but for convex programs also sufficient (subject to certain constraint qualifications we do not discuss here)
- We solve the nonlinear program by iteratively moving along feasible directions

Recall: Feasible Directions

- Seek a point where there is no improving feasible direction
- Let $J \subseteq I$ be the set of tight constraints at $\bar{\boldsymbol{x}}$

$$\forall i \in J : g_i(\bar{\boldsymbol{x}}) = 0, \quad \forall i \in I \setminus J : g_i(\bar{\boldsymbol{x}}) < 0$$

- An improving feasible direction is the below (if exists):

$$\boldsymbol{d} : \nabla f(\bar{\boldsymbol{x}})^T \boldsymbol{d} < 0,\ \nabla g_i(\bar{\boldsymbol{x}})^T \boldsymbol{d} < 0 \quad \forall i \in J$$

- We can find such $\boldsymbol{d}$ by solving the below problem:

$$z_{opt} = \min \nabla f(\bar{\boldsymbol{x}})^T \boldsymbol{d} :\ |\boldsymbol{d}| \leq 1,\ \nabla g_i(\bar{\boldsymbol{x}})^T \boldsymbol{d} < 0 \quad \forall i \in J$$

- Linear objective function, linear (linearizable) constraints
- **Successive linear programming**

Recall: Line Search

- If $z_{opt} \sim 0$ then $\bar{\boldsymbol{x}}$ is (probably) a local minimum: in some pathological cases the condition might not be sufficient
- If $z_{opt} < 0$ then $\boldsymbol{d}$ is an improving feasible direction
- **Line search** along the ray $\bar{\boldsymbol{x}} + \lambda \boldsymbol{d} : \lambda > 0$ so that
  - the feasible region is not left and
  - the objective function is minimized along $\boldsymbol{d}$
- Simple nonlinear program with a single unknown, where the feasible region is an interval (or the entire $\mathbb{R}$)

$$\min f(\bar{\boldsymbol{x}} + \lambda \boldsymbol{d}) : \bar{\boldsymbol{x}} + \lambda \boldsymbol{d} \in X, \lambda \geq 0$$

- Interior point methods often work this way: find a “good” direction and perform line search to find the ideal step size
- We need an efficient algorithm to perform line search

Line Search

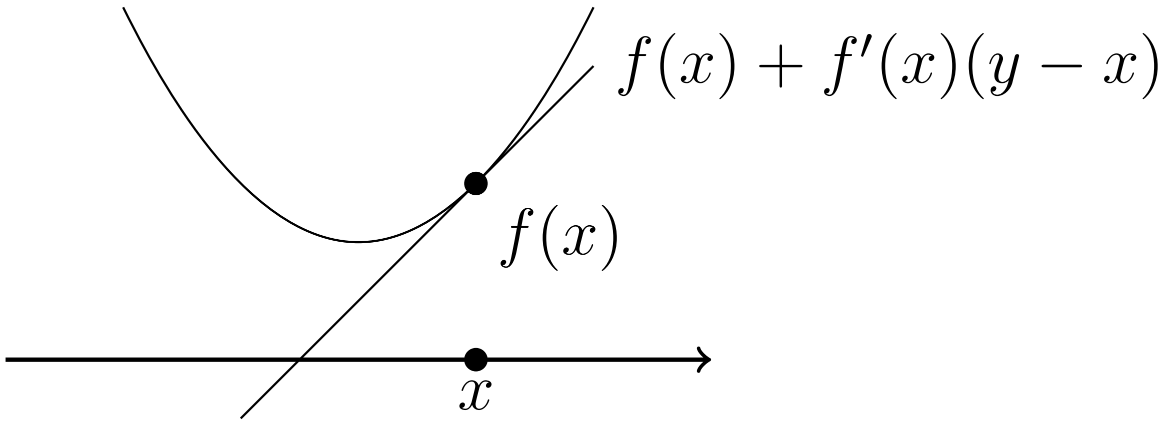
- Find the optimal solution of the optimization problem $\min \theta(\lambda) : \lambda \in [a, b]$ where θ is **convex** and **smooth**
- We concentrate on convex functions: the ideas can be generalized to the nonconvex case as well
- We rely on the assumption that θ is differentiable: for the convex nonsmooth case, one can use dichotomous search for instance
- The interval $[a, b]$ is called the **search interval**

Smooth Convex Line Search

- **Theorem:** a smooth function f is convex on some set X if and only if

$$\forall \boldsymbol{x}, \boldsymbol{y} \in X : \quad f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T (\boldsymbol{y} - \boldsymbol{x}) \tag{1}$$

- Geometrically: the tangent space at $\boldsymbol{x}$ bounds f from below



- Useful for **outer linearization**: approximating a nonlinear constraint system $X = \{\boldsymbol{x} : g_i(\boldsymbol{x}) \leq 0, i = 1, \ldots, m\}$ at $\bar{\boldsymbol{x}}$ with $\bar{X} = \{\boldsymbol{x} : g_i(\bar{\boldsymbol{x}}) + \nabla g_i(\bar{\boldsymbol{x}})^T (\boldsymbol{x} - \bar{\boldsymbol{x}}) \leq 0, i = 1, \ldots, m\}$

Smooth Convex Line Search

- **Proof:** supposing first that f is convex, we show that (1) holds
- f is convex: the line segment between $f(\boldsymbol{x})$ and $f(\boldsymbol{y})$ upper bounds the function of f between points $\boldsymbol{x}$ and $\boldsymbol{y}$

$$\forall \lambda \in [0,1] : f(\lambda \boldsymbol{y} + (1-\lambda)\boldsymbol{x}) \leq \lambda f(\boldsymbol{y}) + (1-\lambda) f(\boldsymbol{x})$$

- Using that $\mu a + (1-\mu)b = b + \mu(a-b)$:

$$f(\boldsymbol{x} + \lambda(\boldsymbol{y} - \boldsymbol{x})) \leq f(\boldsymbol{x}) + \lambda(f(\boldsymbol{y}) - f(\boldsymbol{x}))$$

- Bringing $f(\boldsymbol{x})$ to the left-hand side and dividing by λ:

$$\frac{f(\boldsymbol{x} + \lambda(\boldsymbol{y} - \boldsymbol{x})) - f(\boldsymbol{x})}{\lambda} \leq f(\boldsymbol{y}) - f(\boldsymbol{x})$$

Smooth Convex Line Search

- Taking the limit $\lambda \to 0$ gives the desired inequality:

$$\nabla f(\boldsymbol{x})^T (\boldsymbol{y} - \boldsymbol{x}) \leq f(\boldsymbol{y}) - f(\boldsymbol{x})$$

since the directional derivative of f in the direction $(\boldsymbol{y} - \boldsymbol{x})$:

$$\lim_{\lambda \to 0} \frac{f(\boldsymbol{x} + \lambda(\boldsymbol{y} - \boldsymbol{x})) - f(\boldsymbol{x})}{\lambda} = \nabla^T f(\boldsymbol{x})(\boldsymbol{y} - \boldsymbol{x})$$

- In the other direction: we show that from (1) the convexity of f follows
- Let $\boldsymbol{x}_1, \boldsymbol{x}_2 \in X$ by two arbitrary points in X and let $\boldsymbol{x}$ be a convex combination of $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$

$$\boldsymbol{x} = \lambda \boldsymbol{x}_1 + (1 - \lambda)\boldsymbol{x}_2, \quad \lambda \in [0, 1]$$

Smooth Convex Line Search

- First, write (1) for the case when $\boldsymbol{y} = \boldsymbol{x}_1$

$$f(\boldsymbol{x}_1) \geq f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T(\boldsymbol{x}_1 - \boldsymbol{x}) \qquad (2)$$

- Second, write (1) for the case when $\boldsymbol{y} = \boldsymbol{x}_2$

$$f(\boldsymbol{x}_2) \geq f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T(\boldsymbol{x}_2 - \boldsymbol{x}) \qquad (3)$$

- Multiply (2) by λ and (3) by $1 - \lambda$ and sum the two:

$$\begin{aligned} \lambda f(\boldsymbol{x}_1) &\geq \lambda f(\boldsymbol{x}) + \lambda \nabla f(\boldsymbol{x})^T(\boldsymbol{x}_1 - \boldsymbol{x}) \\ (1-\lambda) f(\boldsymbol{x}_2) &\geq (1-\lambda) f(\boldsymbol{x}) + (1-\lambda) \nabla f(\boldsymbol{x})^T(\boldsymbol{x}_2 - \boldsymbol{x}) \end{aligned}$$

$$\lambda f(\boldsymbol{x}_1) + (1-\lambda) f(\boldsymbol{x}_2) \geq f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T(\lambda \boldsymbol{x}_1 + (1-\lambda)\boldsymbol{x}_2 - \boldsymbol{x})$$

Smooth Convex Line Search

- Since $\boldsymbol{x}$ is a convex combination of $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ we can:
  - on the right-hand side substitute $f(\boldsymbol{x})$ with $f(\lambda \boldsymbol{x}_1 + (1-\lambda)\boldsymbol{x}_2)$
  - rewrite the gradient for the expression in parentheses:

$$\lambda \boldsymbol{x}_1 + (1-\lambda)\boldsymbol{x}_2 - \boldsymbol{x} = \\ \lambda \boldsymbol{x}_1 + (1-\lambda)\boldsymbol{x}_2 - (\lambda \boldsymbol{x}_1 + (1-\lambda)\boldsymbol{x}_2) = 0$$

- We obtain precisely the definition of convexity of f:

$$\lambda f(\boldsymbol{x}_1) + (1-\lambda) f(\boldsymbol{x}_2) \geq f(\lambda \boldsymbol{x}_1 + (1-\lambda)\boldsymbol{x}_2)$$

- Holds for any $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, and $\boldsymbol{x}$, which concludes the proof $\square$
- One of the cornerstone theorems of convex analysis

Smooth Convex Line Search

- **Corollary:** given f be smooth and convex and $\bar{\boldsymbol{x}} \in \mathbb{R}^n$, $f(\bar{\boldsymbol{x}})$ is a global minimum of f if and only if $\nabla f(\bar{\boldsymbol{x}}) = \mathbf{0}$
- **Proof:** earlier, we have seen that $\nabla f(\bar{\boldsymbol{x}}) = \mathbf{0}$ is necessary for $f(\bar{\boldsymbol{x}})$ to be a minimum
- In the general case $\nabla f(\bar{\boldsymbol{x}}) = \mathbf{0}$ is not sufficient though (it also holds at inflection points, saddle points, etc.)
- For convex functions, however, it is also sufficient
- To see this, suppose $\nabla f(\bar{\boldsymbol{x}}) = \mathbf{0}$ and use the previous theorem for $\bar{\boldsymbol{x}}$ and arbitrary $\boldsymbol{y}$:

$$f(\boldsymbol{y}) \geq f(\bar{\boldsymbol{x}}) + \nabla f(\bar{\boldsymbol{x}})^T(\boldsymbol{y} - \bar{\boldsymbol{x}}) = f(\bar{\boldsymbol{x}}) + 0$$

- So $f(\bar{\boldsymbol{x}})$ is a global minimum $\square$

Smooth Convex Line Search

- In single dimension, the finding extreme values is much simpler
- A function $\theta(\lambda) : \mathbb{R} \mapsto \mathbb{R}$ is convex if and only if

$$\forall \mu, \nu : \quad \theta(\mu) \geq \theta(\nu) + \theta'(\nu)(\mu - \nu) \tag{4}$$

- For finding the minimum, simply find $\lambda : \theta'(\lambda) = 0$
- We can find the minimum for a smooth convex function efficiently using the above two results
- We can use simple binary search for finding $\lambda : \theta'(\lambda) = 0$ and it is enough to evaluate θ only once in each iteration

Smooth Convex Line Search

- Solve $\min \theta(\lambda) : \lambda \in [a, b]$, where θ is a smooth convex function on the search interval $[a, b]$
- Take any point $\lambda \in [a, b]$ and evaluate $\theta'(\lambda)$-t
- One of three cases can occur:
  - $\theta'(\lambda) = 0$: by the above, θ attains the minimum at λ
  - $\theta'(\lambda) > 0$: for any $\forall \mu > \lambda : \theta'(\lambda)(\mu - \lambda) > 0$, so by (4) we obtain $\theta(\mu) > \theta(\lambda)$ and therefore the minimum cannot occur in the interval $[\lambda, b]$ and thus the search interval can be narrowed down to $[a, \lambda]$
  - $\theta'(\lambda) < 0$: just the opposite, $\forall \mu < \lambda : \theta(\mu) > \theta(\lambda)$ since $\theta'(\lambda)(\mu - \lambda) > 0$ by $\theta'(\lambda) < 0$ and $\mu - \lambda < 0$, and the new search interval is $[\lambda, b]$

Smooth Convex Line Search

- Choose λ so that the narrowed down search interval $\max\{\lambda - a, b - \lambda\}$ is minimal in the worst-case
- It is easy to see that this occurs at the middle point $\lambda = \frac{1}{2}(a + b)$ of the interval $[a, b]$
- Simple binary search:
  - Initialization: choose search precision $l > 0$, let $a_1 = a$, $b_1 = b$, and let $k = 1$

1. let $\lambda_k = \frac{1}{2}(a_k + b_k)$ and evalutate $\theta'(\lambda_k)$
2. if $\theta'(\lambda) = 0$ or $b_k - a_k < l$ then halt: the minimum occurs in the interval $[a_k, b_k]$
3. if $\theta'(\lambda) > 0$ then $a_{k+1} = a_k$ and $b_{k+1} = \lambda_k$, otherwise $a_{k+1} = \lambda_k$ and $b_{k+1} = b_k$, and go to step (1)

Smooth Convex Line Search

- The size of the search interval at step k: $\frac{1}{2^k}(b-a)$
- The number of steps to reach precision l: $k \geq \log_2 \frac{b-a}{l}$
- Example: find the minimum of the function $\theta(\lambda) = \lambda^2 + 2\lambda$ on the interval $[-5, 15]$ with precision $l = 2 \cdot 10^{-2}$
- We seek λ that satisfies $\theta'(\lambda) = 2\lambda + 2 = 0$, but very often the algebraic equation $\theta'(\lambda) = 0$ cannot be solved directly
- Using our numeric method (binary search) instead:

k	a	b	λ	$\theta'(\lambda)$	$\theta(\lambda)$	$b-a$
1	-5.0000	15.0000	5.0000	12.0000	35.0000	20.0000
2	-5.0000	5.0000	0.0000	2.0000	0.0000	10.0000
3	-5.0000	0.0000	-2.5000	-3.0000	1.2500	5.0000
$\vdots$	...	...	...	...	...	...
11	-1.0156	-0.9961	-1.0059	0.0118	-1.0000	0.0195

Unconstrained Optimization

- In practice we often need to move beyond single-dimension line search and minimize multivariable $\mathbb{R}^n \mapsto \mathbb{R}$ functions
- Given a nonlinear program $\min f(\boldsymbol{x}) : \boldsymbol{x} \in X$, let $X = \mathbb{R}^n$
- Suppose that f is smooth and convex
- The minimum occurs at point $\bar{\boldsymbol{x}}$ where $\nabla f(\bar{\boldsymbol{x}}) = \mathbf{0}$
- Usually this cannot be solved analytically, we need numeric methods and approximations
- We already have a method to optimize convex programs: the method of feasible directions due to Zoutendijk
- Much simpler in our case since now all directions are feasible ($X = \mathbb{R}^n$)
- We only need to take care of ensuring that the direction we find be improving

The Steepest Descent Method

- Some $\boldsymbol{d}$ is an improving direction at $\bar{\boldsymbol{x}}$ if $\nabla f(\bar{\boldsymbol{x}})^T \boldsymbol{d} < 0$
- **Steepest descent:** the unit vector $\hat{\boldsymbol{d}}$ moving along which we experience the largest drop in the value of f
- **Theorem:** $\underset{\hat{\boldsymbol{d}}:\|\hat{\boldsymbol{d}}\|=1}{\operatorname{argmin}} \nabla f(\bar{\boldsymbol{x}})^T \hat{\boldsymbol{d}} = -\dfrac{\nabla f(\bar{\boldsymbol{x}})}{\|\nabla f(\bar{\boldsymbol{x}})\|}$
- **Proof:** using the Cauchy–Schwarz Inequality and $\|\hat{\boldsymbol{d}}\| = 1$:

$$\nabla f(\bar{\boldsymbol{x}})^T \hat{\boldsymbol{d}} \geq -\|\nabla f(\bar{\boldsymbol{x}})\| \|\hat{\boldsymbol{d}}\| \geq -\|\nabla f(\bar{\boldsymbol{x}})\|$$

- Holds with equality if and only if $\hat{\boldsymbol{d}} = \frac{-\nabla f(\bar{\boldsymbol{x}})}{\|\nabla f(\bar{\boldsymbol{x}})\|}$ $\square$
- So the best improving direction is $\hat{\boldsymbol{d}} = \frac{-\nabla f(\bar{\boldsymbol{x}})}{\|\nabla f(\bar{\boldsymbol{x}})\|}$, then we can again use line search to find the ideal step size along $\hat{\boldsymbol{d}}$

The Steepest Descent Method

- Solve the unconstrained nonlinear optimization problem $\min f(\boldsymbol{x}) : \boldsymbol{x} \in \mathbb{R}^n$ where f is smooth and convex
- Initialization: choose precision $\epsilon > 0$ and x_1 initial point, and let $k = 1$

1 Of $\|\nabla f(\boldsymbol{x}_k)\| < \epsilon$ then halt, $\boldsymbol{x}_k$ is minimal

2 Otherwise $\boldsymbol{d}_k = -\nabla f(\boldsymbol{x}_k)$

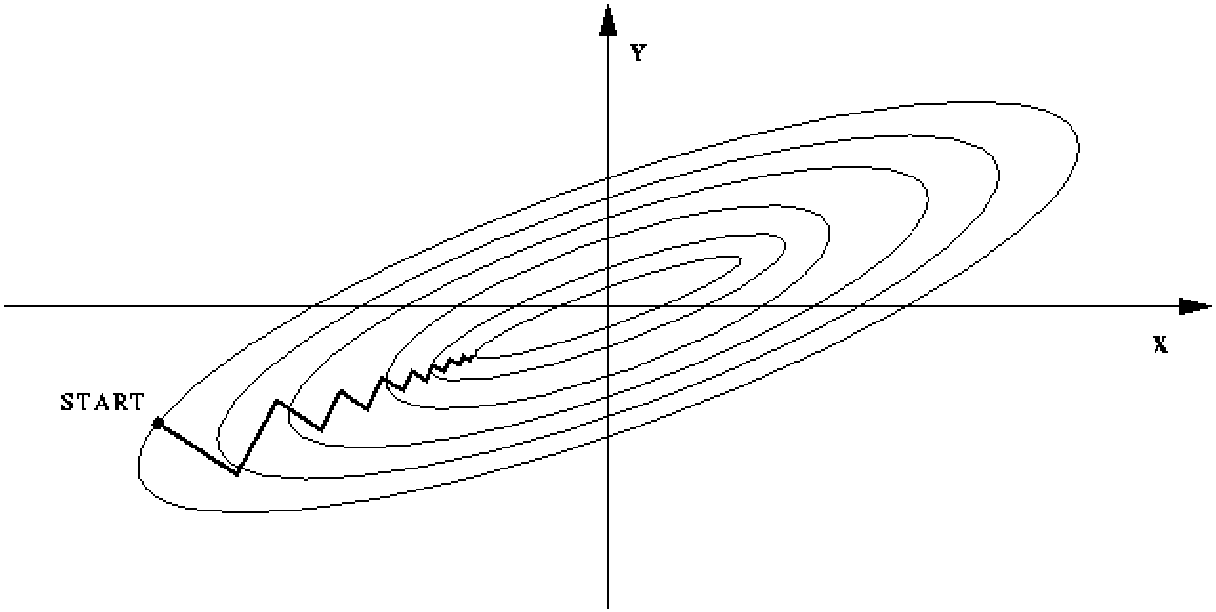
3 Line search: solve the unconstrained search problem

$$\lambda_k = \min f(\boldsymbol{x}_k + \lambda \boldsymbol{d}_k) : \lambda \geq 0$$

4 Let $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \lambda_k \boldsymbol{d}_k$ and go to step (2)

The Steepest Descent Method

- Simple and, theoretically, it converges in finite steps, but hardly usable in practice due to zig-zagging



- Higher-order methods can avoid this: Newton's method, conjugate gradient, etc.

Penalty and Barrier Functions

- We have seen that constraints make the optimization much more difficult: improving directions are easy to find, feasible directions may be more difficult
- It would be nice to trace back constrained nonlinear programming to unconstrained optimization
- The idea is to enforce constraints via the objective function, by adding a term to the objective to penalize any violation of the constraints and to remove the constraints themselves
- Minimization of the resultant unconstrained problem will try to remove the penalty and thereby to satisfy the constraints
  - (exterior) penalty functions: infeasible points are penalized, the further the point from the feasible region the larger the penalty
  - (interior) barrier functions: even approaching the boundary of the feasible region already carries a penalty

Penalty Function Methods

- Consider the simple constrained nonlinear optimization problem
$$\min f(\boldsymbol{x}) : h(\boldsymbol{x}) = 0$$
where f and h are $\mathbb{R}^n \mapsto \mathbb{R}$ functions and $\boldsymbol{x} \in \mathbb{R}^n$
- Replace this constrained problem with the below unconstrained one:
$$\min f(\boldsymbol{x}) + \mu h^2(\boldsymbol{x}) : \boldsymbol{x} \in \mathbb{R}^n$$
where $\mu > 0$ is some large scalar
- Intuitively, at optimum the **(external) penalty function** $\mu h^2(\boldsymbol{x})$ will take a value close to zero, otherwise the penalty would prohibitively increase the objective value
- Choosing μ large enough ensures that the minimum of the original problem and the unconstrained problem coincide

Penalty Function Methods

- Nonlinear programs containing inequality constraints need a different penalty function

$$\min f(\boldsymbol{x}) : g(\boldsymbol{x}) \leq 0$$

- The penalty function $\mu g^2(\boldsymbol{x})$ would not work since it would penalize both $g(\boldsymbol{x}) < 0$ and $g(\boldsymbol{x}) > 0$ whereas we need to penalize only the latter
- Consider the below unconstrained problem instead:

$$\min f(\boldsymbol{x}) + \mu \max\{0, g(\boldsymbol{x})\} : \boldsymbol{x} \in \mathbb{R}^n$$

- Penalty is applied only if $g(\boldsymbol{x}) > 0$, since for $g(\boldsymbol{x}) < 0$ the penalty disappears by $\max\{0, g(\boldsymbol{x})\} = 0$
- Even better penalty function is $\mu(\max\{0, g(\boldsymbol{x})\})^2$: smooth!

Penalty Function Methods

- For the generic case when the nonlinear program contains both equality and inequality constraints:

$$\begin{aligned} \min \quad & f(\boldsymbol{x}) & \\ \text{s.t.} \quad & g_i(\boldsymbol{x}) \leq 0 & i \in \{1, \ldots, m\} \\ & h_i(\boldsymbol{x}) = 0 & i \in \{1, \ldots, n\} \end{aligned}$$

- The external penalty functions:

$$\alpha(\boldsymbol{x}) = \sum_{i=1}^{m} \Phi(g_i(\boldsymbol{x})) + \sum_{i=1}^{n} \Psi(h_i(\boldsymbol{x}))$$

$$\Phi(y) = (\max\{0, y\})^p, \qquad \Psi(y) = |y|^p$$

for some $p > 0$ integer

Penalty Function Methods

- We obtain the below unconstrained nonlinear program:

$$\min f(\boldsymbol{x}) + \mu \left(\sum_{i=1}^{m} (\max\{0, g_i(\boldsymbol{x})\})^p + \sum_{i=1}^{n} |h_i(\boldsymbol{x})|^p \right) :$$

$$\boldsymbol{x} \in \mathbb{R}^n$$

- Can be solved using the unconstrained optimization methods discussed previously
- Can be started from arbitrary, even from an infeasible point
- Drawback is that the search enters into the feasible region only in the vicinity of the optimum, exactly where zig-zagging is most prominent
- Large μ needed for reliable result: numeric instability

Penalty Function Methods: Example

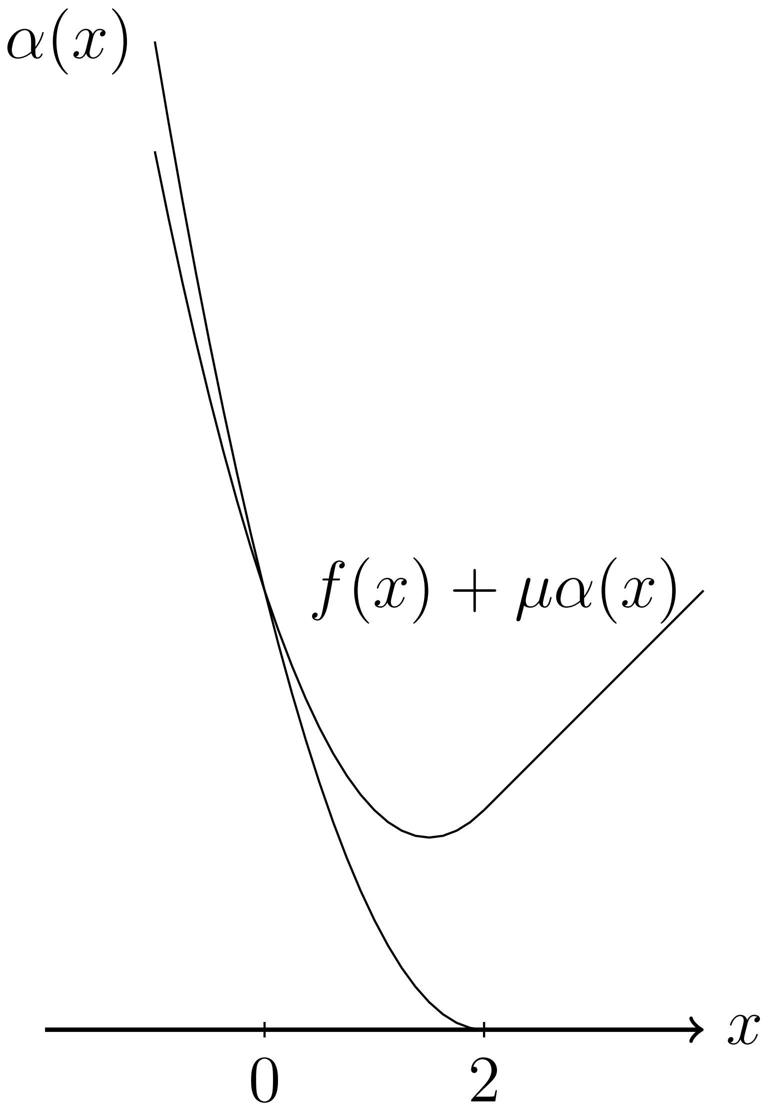
- Consider the constrained (non)linear program

$$\begin{aligned} \min \;& x \\ \text{s.t.} \;& -x + 2 \leq 0 \end{aligned}$$

- Optimum at $\bar{x} = 2$: $f(\bar{x}) = 2$
- Solve the problem using penalty functions
- Using the penalty function $\alpha(x) = (\max\{0, g(x)\})^2$ the constrained problem as an unconstrained nonlinear program:

$$\alpha(x) = \begin{cases} 0 & \text{ha } x \geq 2 \\ (-x+2)^2 & \text{ha } x < 2 \end{cases}$$

Penalty Function Methods: Example



- $f(x) + \mu\alpha(x)$ is convex, so we can find the minimum easily
- The derivative for $x < 2$:

$$(f(x) + \mu\alpha(x))' = 1 + 2\mu(x - 2) = 0$$

- The optimal solution:

$$\bar{x} = 2 - \frac{1}{2\mu}$$

- Approximates the real optimum as $\mu \to \infty$
- In general, guaranteed to converge in finite steps

Penalty Function Methods: Example

- Solve the constrained optimization problem

$$\min x_1^2 + x_2^2 : x_1 + x_2 - 1 = 0$$

- Optimal solution: $\begin{bmatrix} x_1 & x_2 \end{bmatrix}^T = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \end{bmatrix}^T$, optimum: $\frac{1}{2}$
- Apply the penalty function $\alpha(y) = |y|^2$:

$$\min x_1^2 + x_2^2 + \mu(x_1 + x_2 - 1)^2 : x_1, x_2 \in \mathbb{R}$$

- Convex for $\forall \mu \geq 0$, so the minimum occurs where:

$$2x_1 + 2\mu(x_1 + x_2 - 1) = 0$$
$$2x_2 + 2\mu(x_1 + x_2 - 1) = 0$$

- From the two equations: $x_1 = x_2 = \frac{1}{2+\frac{1}{\mu}}$, optimal if $\mu \to \infty$

Barrier Function Methods

- External penalty function methods converge through a sequence of infeasible points
- Internal barrier functions do not allow the search to leave the feasible region in the first place
- Given a linear program with inequality constraints:

$$\begin{aligned} &\min \ f(\boldsymbol{x}) && \\ &\text{s.t.} \ g_i(\boldsymbol{x}) \leq 0 && i \in \{1, \ldots, m\} \end{aligned}$$

- Instead, solve the unconstrained problem

$$\min f(\boldsymbol{x}) + \mu\beta(\boldsymbol{x}) : \boldsymbol{x} \in \mathbb{R}^n$$

Barrier Function Methods

- The internal barrier function β takes nonnegative values in the interior of the feasible region and tends to infinity at the boundary:

$$\beta(\boldsymbol{x}) = \sum_{i=1}^{m} \Phi(g_i(\boldsymbol{x}))$$

where barrier function Φ has the following properties:

$$\Phi(y) \geq 0 \text{ if } y < 0, \qquad \lim_{y \to 0^-} \Phi(y) = \infty$$

- The barrier function prohibits passing the boundary of the feasible region
- Equality conditions are more involving to handle here

Barrier Function Methods

- Typical internal barrier functions:

$$\beta(\boldsymbol{x}) = \sum_{i=1}^{m} \frac{-1}{g_i(\boldsymbol{x})}, \qquad \beta(\boldsymbol{x}) = -\sum_{i=1}^{m} \ln\left(\min\left\{1, -g_i(\boldsymbol{x})\right\}\right)$$

- The augmented objective function can be optimized using unconstrained methods and we never step out of the feasible region
- Drawback is that the penalty appears even at the optimum and the optimum cannot occur at a boundary point
- Plus the iteration must start from a feasible point, otherwise we never get to the other “good” side of the barrier
- Works reliably when μ is small, but too small value for μ again causes numerical instability and badly conditioned problems

Barrier Function Methods: Example

- Consider the nonlinear program

$$\begin{aligned} \min \;& x \\ \text{s.t.} \;& -x+1 \leq 0 \end{aligned}$$

- The optimum is 1, attained by the objective at $\bar{x} = 1$
- Let the internal barrier function be as follows:

$$\beta(x) = \frac{-1}{-x+1} = \frac{1}{x-1} \qquad x \neq 1$$

- The resultant unconstrained problem:

$$\min x + \frac{\mu}{x-1} : x \in \mathbb{R}$$

Barrier Function Methods: Example

- Solving $\min x + \frac{\mu}{x-1} : x \in \mathbb{R}$ only makes sense of $x > 1$, we need to take care of this ourselves
- The objective function is convex, the minimum occurs at $\frac{d}{dx}(f(x) + \mu\beta(x)) = 0$ from which we obtain $\bar{x} = 1 + \sqrt{\mu}$, optimal for $\mu \to 0$