Primal/Dual Decomposition Methods

Daniel P. Palomar
The Hong Kong University of Science and Technology (HKUST)

ELEC5470/IEDA6100A - Convex Optimization
The Hong Kong University of Science and Technology (HKUST)
Fall 2020-21

Outline of Lecture

- Subgradients
- Subgradient methods
- Primal decomposition
- Dual decomposition
- Summary

(Acknowledgement to Stephen Boyd for material for this lecture.)

Gradient and First-Order Approximation

- Recall basic inequality for convex differentiable f:

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^T (\mathbf{y} - \mathbf{x}) \qquad \forall \mathbf{y}$$

- First-order approximation of f at $\mathbf{x}$ is a global underestimator, a supporting hyperplane.

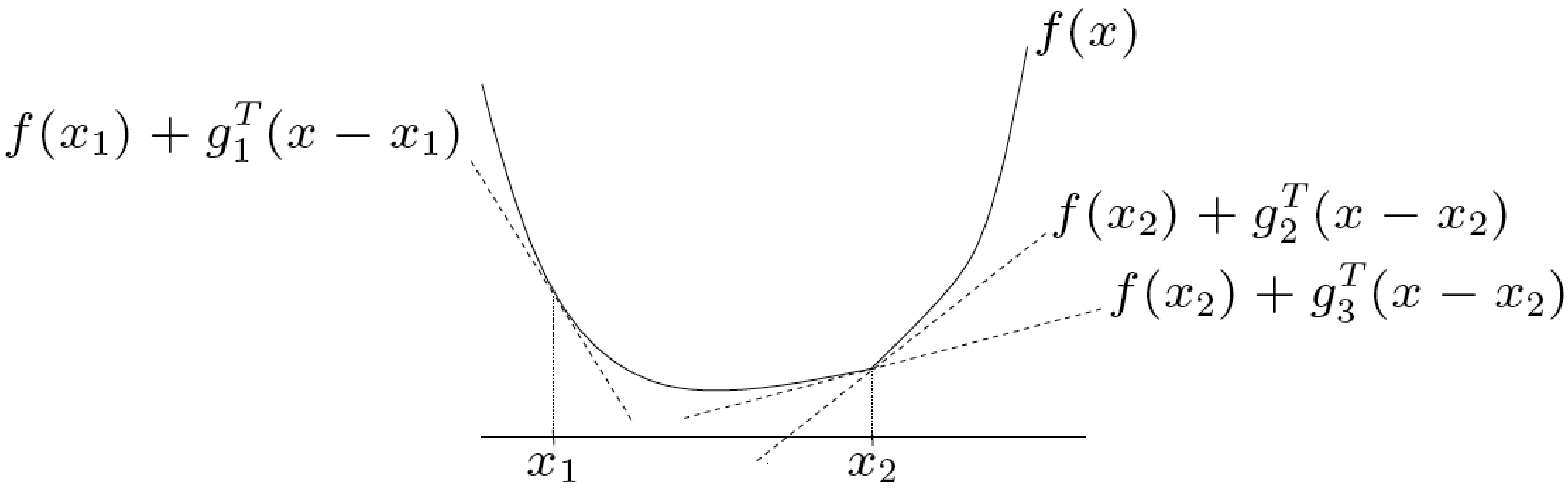
- What if f is not differentiable?

- The answer is given by the concept of $subgradient$.

Subgradient of a Function

- $\mathbf{g}$ is a *subgradient* of f (not necessarily convex) at $\mathbf{x}$ if

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \mathbf{g}^T(\mathbf{y} - \mathbf{x}) \qquad \forall \mathbf{y}$$



Subgradient of a Function

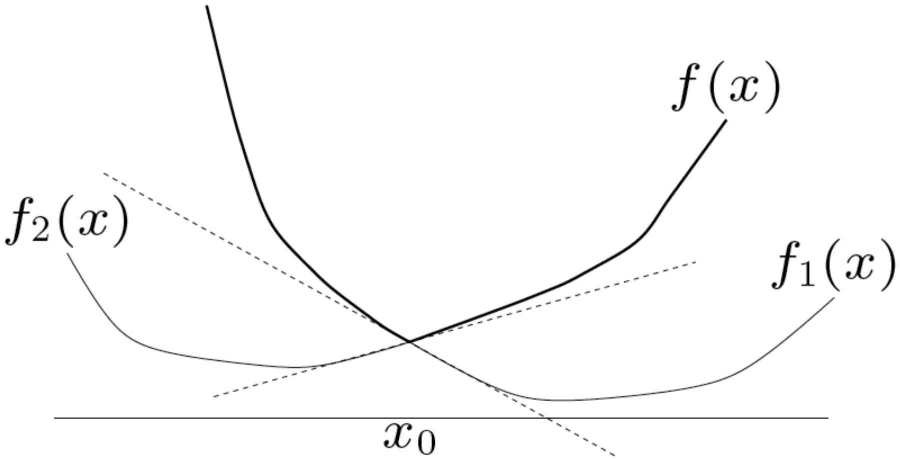
- $\mathbf{g}$ is a subgradient if and only if $f(\mathbf{x}) + \mathbf{g}^T(\mathbf{y} - \mathbf{x})$ is a global affine underestimator of f.

- If f is convex and differentiable, then $\nabla f(\mathbf{x})$ is a subgradient of f at $\mathbf{x}$.

- Subgradients come up in several contexts:
  - algorithms for nondifferentiable convex optimization
  - convex analysis, e.g., optimality conditions and duality for nondifferentiable problems.

Example of Subgradient

- Consider $f = \max\{f_1, f_2\}$ with f_1 and f_2 convex and differentiable



- Subgradient at point $\mathbf{x}$:
  - If $f_1(\mathbf{x}) > f_2(\mathbf{x})$, the subgradient is unique $\mathbf{g} = \nabla f_1(\mathbf{x})$.
  - If $f_2(\mathbf{x}) > f_1(\mathbf{x})$, the subgradient is unique $\mathbf{g} = \nabla f_2(\mathbf{x})$.
  - If $f_1(\mathbf{x}) = f_2(\mathbf{x})$, the subgradients form an interval $[\nabla f_1(\mathbf{x}), \nabla f_2(\mathbf{x})]$.

Subdifferential

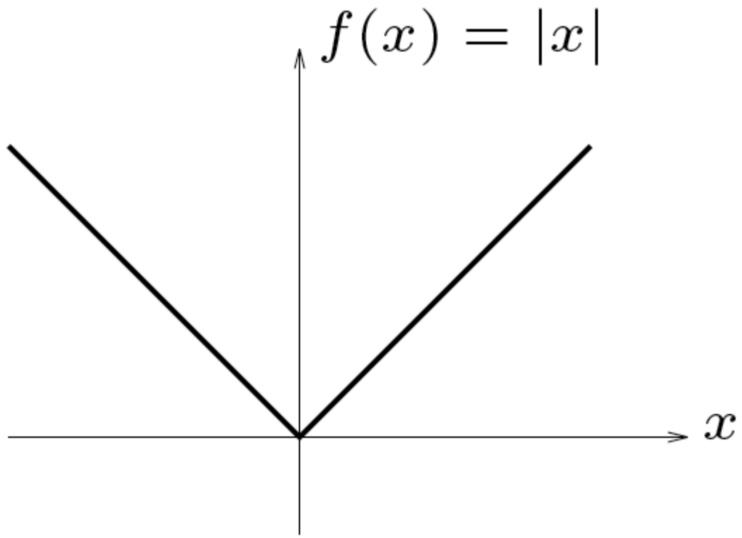
- The set of all subgradients of f at $\mathbf{x}$ is called the *subdifferential* of f at $\mathbf{x}$ and is denoted $\partial f(\mathbf{x})$.

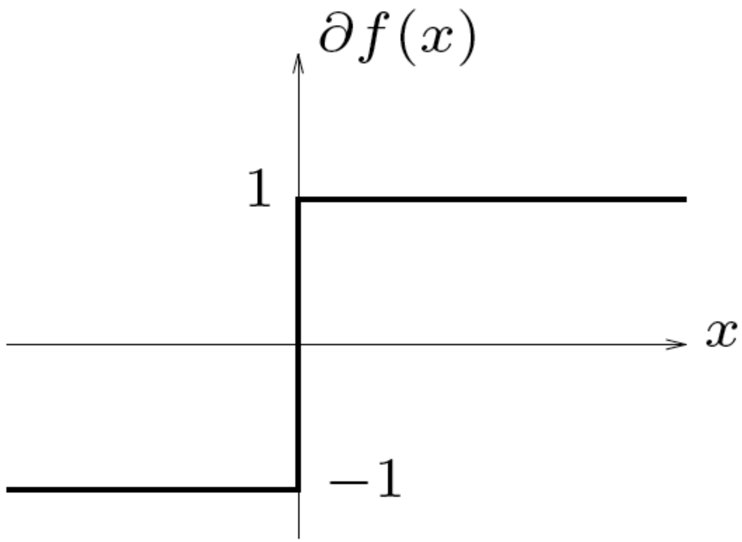
- $\partial f(\mathbf{x})$ is a closed convex set (can be empty).

- If f is convex:
  - $\partial f(\mathbf{x})$ is nonempty for $\mathbf{x} \in \text{relint dom } f$
  - if f is differentiable at $\mathbf{x}$, then $\partial f(\mathbf{x}) = \{\nabla f(\mathbf{x})\}$ (i.e., a singleton).
  - If $\partial f(\mathbf{x}) = \{\mathbf{g}\}$, then f is differentiable at $\mathbf{x}$ and $\mathbf{g} = \nabla f(\mathbf{x})$.

Example of Subdifferential

- Consider $f(x) = |x|$:





Subgradient Calculus

- *Weak subgradient calculus*: formulas for finding one subgradient $\mathbf{g} \in \partial f(\mathbf{x})$.
- *Strong subgradient calculus*: formulas for finding the whole subdifferential $\partial f(\mathbf{x})$, i.e., all subgradients of f at $\mathbf{x}$.
- Many algorithms for nondifferentiable convex optimization require only one subgradient, so weak calculus suffices.
- Some algorithms and optimality conditions need the whole subdifferential.
- In practice, if you can compute $f(\mathbf{x})$, you can usually compute a subgradient $\mathbf{g} \in \partial f(\mathbf{x})$.

Some Basic Rules

(From now on, we will assume that f is convex and $\mathbf{x} \in \text{relint dom } f$.)

- $\partial f(\mathbf{x}) = \{\nabla f(\mathbf{x})\}$ if f is differentiable at $\mathbf{x}$
- Scaling: $\partial(\alpha f) = \alpha \partial f$ (assuming $\alpha > 0$)
- Addition: $\partial(f_1 + f_2) = \partial f_1 + \partial f_2$ (addition of sets)
- Affine transformation of variables: if $g(\mathbf{x}) = f(\mathbf{A}\mathbf{x} + \mathbf{b})$, then $\partial g(\mathbf{x}) = \mathbf{A}^T \partial f(\mathbf{A}\mathbf{x} + \mathbf{b})$
- Finite pointwise maximum: if $f = \max_i f_i$, then

$$\partial f(\mathbf{x}) = \text{Co} \bigcup \{\partial f_i \mid f_i(\mathbf{x}) = f(\mathbf{x})\},$$

i.e., convex hull of union of subdifferentials of active functions at $\mathbf{x}$.

Optimality Conditions: Unconstrained Case

- Recall that for f convex and differentiable, $\mathbf{x}^{\star}$ minimizes $f(\mathbf{x})$ if and only if:

$$\mathbf{0} = \nabla f(\mathbf{x}^{\star}).$$

- The generalization to nondifferentiable convex f is

$$\mathbf{0} \in \partial f(\mathbf{x}^{\star}).$$

Proof. By definition:

$$f(\mathbf{y}) \geq f(\mathbf{x}^{\star}) + \mathbf{0}^{T}(\mathbf{y} - \mathbf{x}^{\star}) \qquad \forall \mathbf{y}.$$

Example: Piecewise Linear Maximization

- We want to minimize $f(\mathbf{x}) = \max_i \left\{\mathbf{a}_i^T\mathbf{x} + b_i\right\}$.

- $\mathbf{x}^\star$ minimizes $f(\mathbf{x}) \Longleftrightarrow \mathbf{0} \in \partial f(\mathbf{x}^\star) = \mathrm{Co}\left\{\mathbf{a}_i \mid \mathbf{a}_i^T\mathbf{x} + b_i = f(\mathbf{x}^\star)\right\} \Longleftrightarrow$ there is a $\boldsymbol{\lambda}$ with

$$\boldsymbol{\lambda} \geq \mathbf{0}, \quad \mathbf{1}^T\boldsymbol{\lambda} = 1, \quad \sum_i \lambda_i \mathbf{a}_i = \mathbf{0}$$

  where $\lambda_i = 0$ if $\mathbf{a}_i^T\mathbf{x}^\star + b_i < f(\mathbf{x}^\star)$.

- Interestingly, these are exactly the KKT conditions for the problem in epigraph form:

$$\begin{array}{ll} \underset{t,\mathbf{x}}{\text{minimize}} & t \\ \text{subject to} & \mathbf{a}_i^T\mathbf{x} + b_i \leq t, \qquad i = 1, \cdots, m\ . \end{array}$$

Optimality Conditions: Constrained Case

$$\begin{array}{ll} \underset{\mathbf{x}}{\text{minimize}} & f_0(\mathbf{x}) \\ \text{subject to} & f_i(\mathbf{x}) \leq 0, \qquad i = 1, \cdots, m \end{array}$$

where each f_i is convex, defined on R^n (hence subdifferentiable) and strict feasibility holds (Slater's condition).

- The KKT necessary and sufficient conditions are

$$\begin{aligned} f_i(\mathbf{x}^\star) \leq 0, & \qquad \lambda_i^\star \geq 0 \\ \mathbf{0} &\in \partial f_0(\mathbf{x}^\star) + \sum_{i=1}^{m} \lambda_i^\star \partial f_i(\mathbf{x}^\star) \\ \lambda_i^\star f_i(\mathbf{x}^\star) &= 0 \quad \text{(complementary slackness).} \end{aligned}$$

Numerical Methods for Nondifferentiable Problems

- In R, we can always use the bisection method for nondifferentiable functions to reduce the interval (uncertainty) by half at each step.

- Can we generalize this to R^n? The problem is that R^n is not ordered, as opposed to R.

- The answer is: yes. These methods are called *localization methods*:
  - *cutting-plane methods* (date back to the 1960s in the Russian literature): the uncertainty set is a polyhedron
  - *ellipsoid method* (goes back to the 1970s in the Russian literature): the uncertainty set is an ellipsoid.

- Another convenient method is the *subgradient method*.

Subgradient Method

- *Subgradient method* is a simple algorithm (looks similar to a gradient method) to minimize a nondifferentiable convex function f:

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha_k \mathbf{g}^{(k)}$$

  where
  - $\mathbf{x}^{(k)}$ is the kth iterate
  - $\mathbf{g}^{(k)}$ is any subgradient of f at $\mathbf{x}^{(k)}$
  - $\alpha_k > 0$ is the kth stepsize

- Note that it is not a descent method (unlike a gradient method), so we need to keep track of the best point so far:

$$f_{\text{best}}^{(k)} = \min_{i=1,\cdots,k} f\left(\mathbf{x}^{(i)}\right).$$

Stepsize Rules

- Different stepsize methods:
  - constant stepsize: $\alpha_k = \alpha$
  - constant step length: $\alpha_k = \gamma / \left\| \mathbf{g}^{(k)} \right\|$ (so $\left\| \mathbf{x}^{(k+1)} - \mathbf{x}^{(k)} \right\|_2 = \gamma$)
  - square summable but not summable: stepsizes satisfying

$$\sum_{k=1}^{\infty} \alpha_k^2 < \infty, \qquad \sum_{k=1}^{\infty} \alpha_k = \infty$$

  - nonsummable diminishing: stepsizes satisfying

$$\lim_{k \to \infty} \alpha_k = 0, \qquad \sum_{k=1}^{\infty} \alpha_k = \infty.$$

Convergence Results

Under some technical conditions (boundedness of optimum value, boundedness of subgradients by G), the limiting value of the subgradient method $\bar{f} = \lim_{k\to\infty} f_{\text{best}}^{(k)}$ satisfies:

- constant stepsize: $\bar{f} - f^\star \leq G^2\alpha/2$, i.e., converges to $G^2\alpha/2$-suboptimal (converges to $f^\star$ if f differentiable and α small enough)

- constant step length: $\bar{f} - f^\star \leq G\gamma/2$, i.e., converges to $G\gamma/2$-suboptimal

- diminishing stepsize rule: $\bar{f} = f^\star$, i.e., converges.

Example: Piecewise Linear Minimization

- Consider the following nondifferentiable optimization problem:

$$\text{minimize} \quad f(\mathbf{x}) = \max_i \left\{ \mathbf{a}_i^T \mathbf{x} + b_i \right\}.$$

- To find a subgradient, simply choose an index j for which

$$\mathbf{a}_j^T \mathbf{x} + b_j = \max_i \left\{ \mathbf{a}_i^T \mathbf{x} + b_i \right\}$$

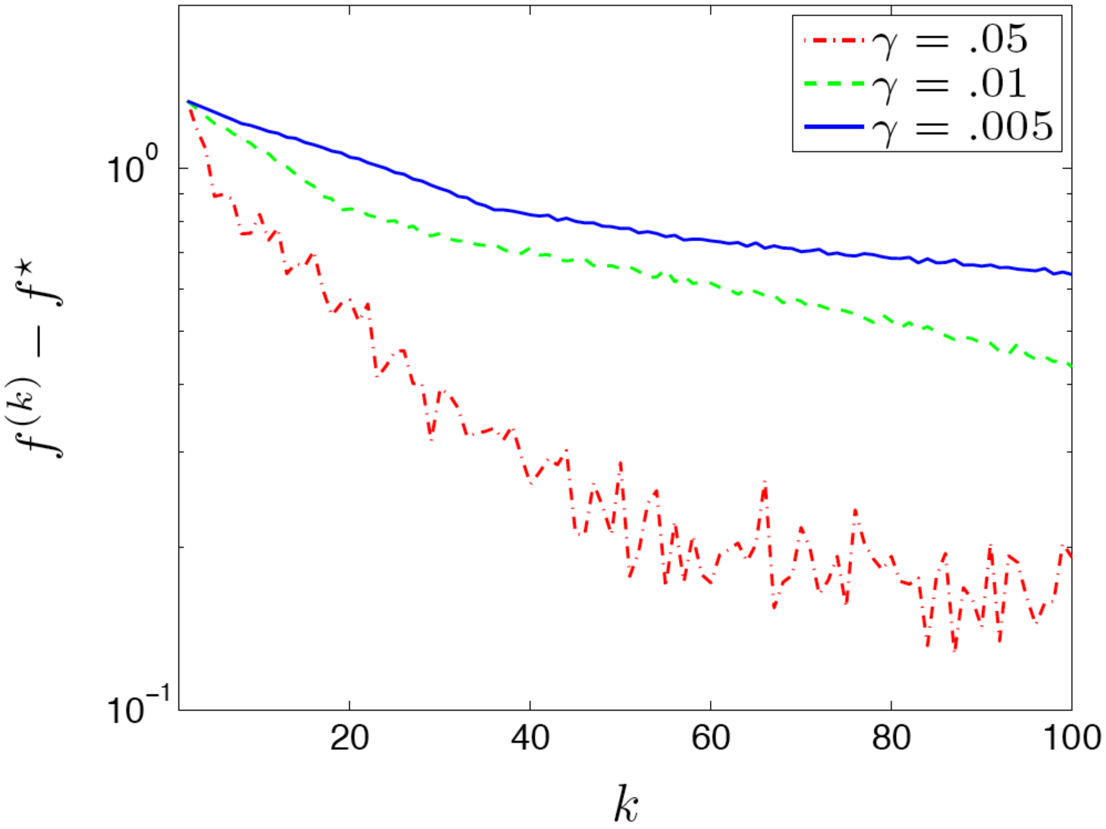
  and take $\mathbf{g} = \mathbf{a}_j$.

- The subgradient update is

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha_k \mathbf{a}_j^{(k)}.$$

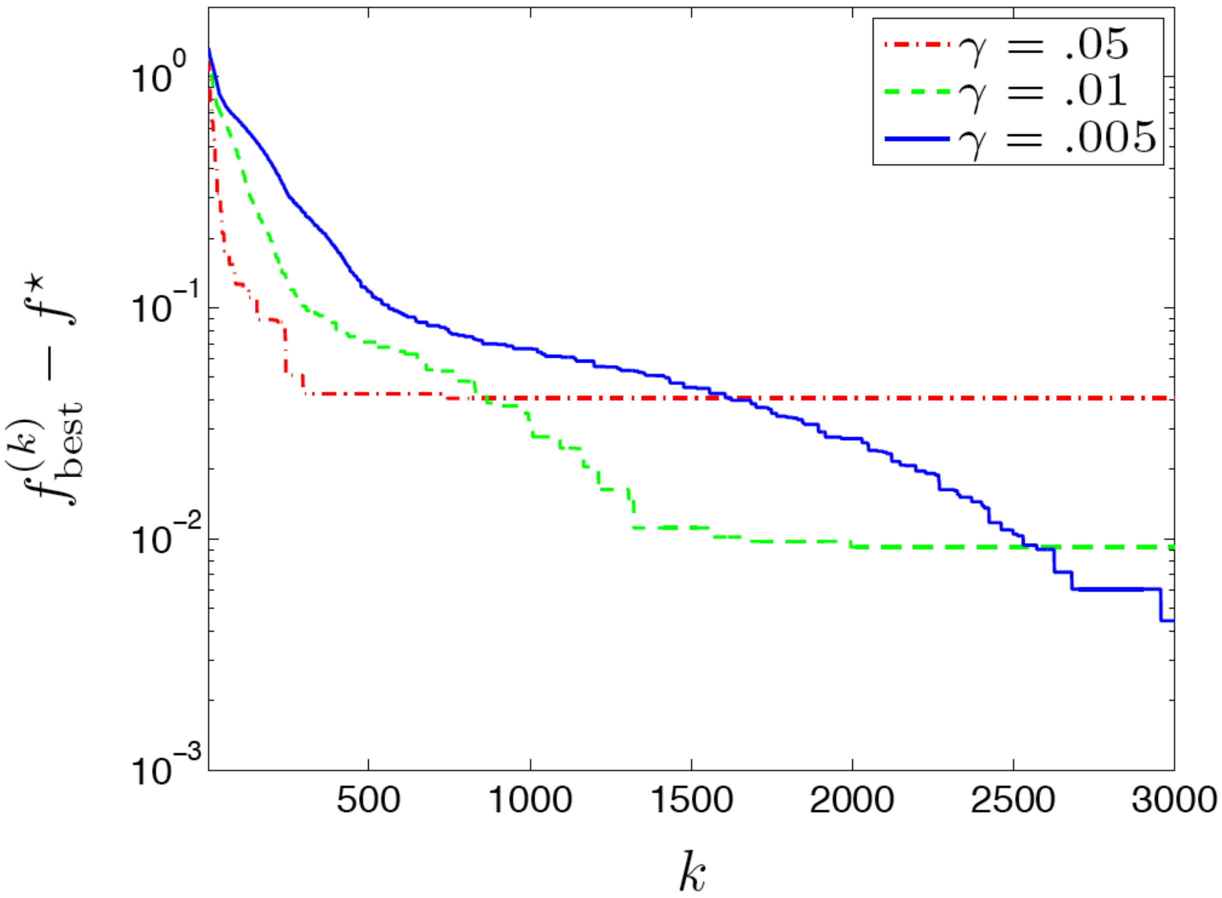
Example: Piecewise Linear Minimization (II)

- Problem instance with $n = 20$ variables, $m = 100$ terms, $f^\star \approx 1.1$
- Constant step length, first 100 iterations, $f^{(k)} - f^\star$:



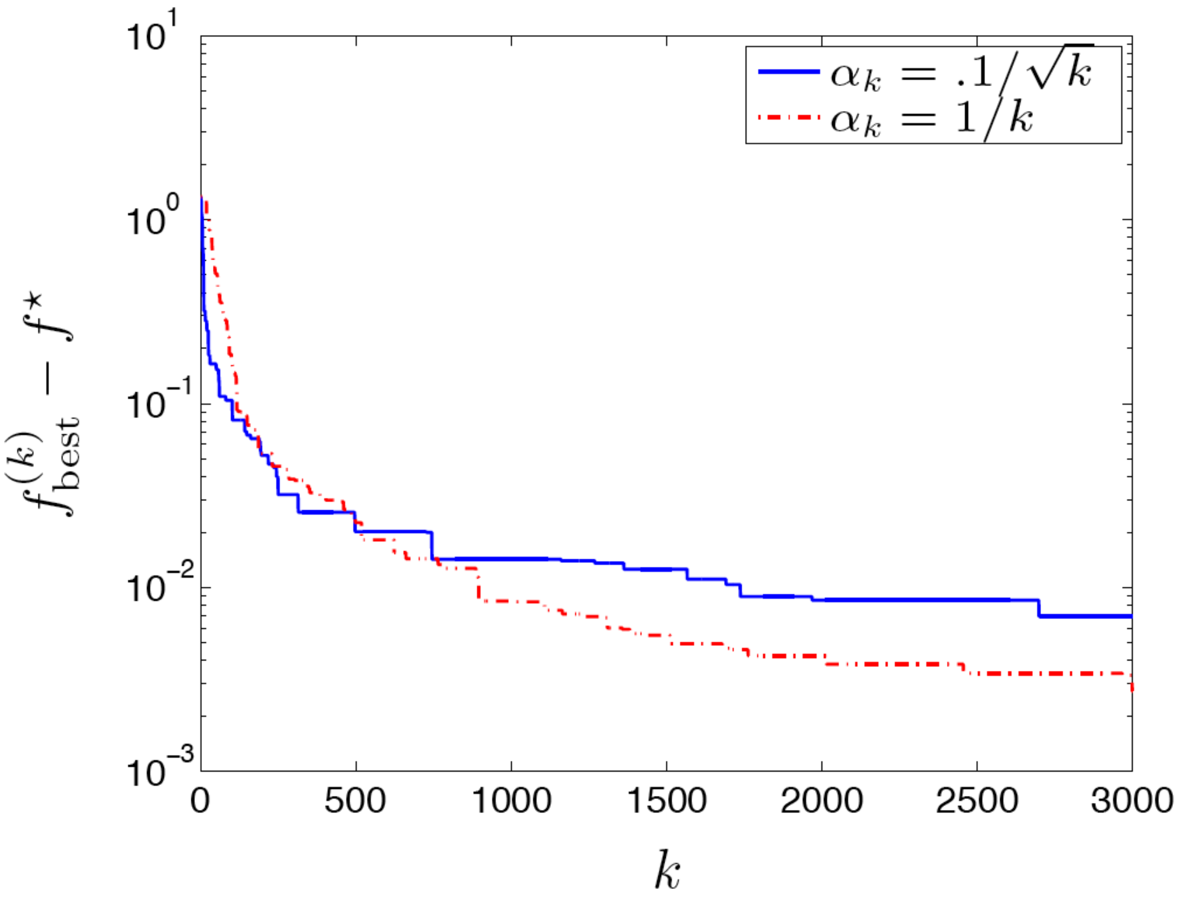
Example: Piecewise Linear Minimization (III)

- Constant step length, $f_{\text{best}}^{(k)} - f^\star$:



Example: Piecewise Linear Minimization (IV)

- Diminishing stepsize rule ($\alpha_k = 0.1/\sqrt{k}$) and square summable stepsize rule ($\alpha_k = 1/k$):



Projected Subgradient Method for Constrained Optimization

- Consider the following constrained optimization problem:

$$\begin{array}{ll} \underset{\mathbf{x}}{\text{minimize}} & f(\mathbf{x}) \\ \text{subject to} & \mathbf{x} \in \mathcal{X}. \end{array}$$

- The *projected subgradient method* guarantees that each update produces a feasible point via a projection:

$$\mathbf{x}^{(k+1)} = \left[\mathbf{x}^{(k)} - \alpha_k \mathbf{g}^{(k)}\right]_{\mathcal{X}}$$

where $[\cdot]_{\mathcal{X}}$ denotes projection onto the convex set $\mathcal{X}$.

Decomposition Methods

- The idea of a decomposition method is to solve a problem by solving smaller subproblems coordinated by a master problem.

- Different reasons to use a decomposition method:
  - to allow the resolution of a problem otherwise unsolvable for memory reasons (useful in areas such as biology or image processing)
  - to speed up the resolution of the problem via parallel computation
  - to solve the problem in a distributed way (desirable for some wireless networks)
  - to derive nice, insightful, and efficient numerical algorithms as alternative to the use of general-purpose interior-point methods.

Decomposition Methods (II)

- In some (few) fortunate cases, problems decouple naturally:

$$\begin{array}{ll} \underset{\mathbf{x}}{\text{minimize}} & f_1\left(\mathbf{x}_1\right) + f_2\left(\mathbf{x}_2\right) \\ \text{subject to} & \mathbf{x}_1 \in \mathcal{X}_1 \ , \qquad \mathbf{x}_2 \in \mathcal{X}_2 \ . \end{array}$$

- We can solve for $\mathbf{x}_1$ and $\mathbf{x}_2$ separately (in parallel).

- Even if they are solved sequentially, there is still the advantage of memory and of speed (if the computational effort is superlinear in problem size).

Decomposition Methods (III)

- In general, however, problems do not decouple so easily and that's when can resort to decomposition methods.

- There are two main types of decompostion methods:
  - *primal decomposition*:
    - deals with complicating variables that couple the subproblems
    - the primal master problem controls directly the resources
  - *dual decomposition*:
    - deals with complicating constraints that couple the subproblems
    - the dual master problem controls the prices of the resources.

Primal Decomposition

- Consider the following problem with a coupling variable:

$$\begin{array}{ll} \underset{\mathbf{x},\mathbf{y}}{\text{minimize}} & f_1\left(\mathbf{x}_1, \mathbf{y}\right) + f_2\left(\mathbf{x}_2, \mathbf{y}\right) \\ \text{subject to} & \mathbf{x}_1 \in \mathcal{X}_1, \qquad \mathbf{x}_2 \in \mathcal{X}_2 \\ & \mathbf{y} \in \mathcal{Y}. \end{array}$$

- $\mathbf{y}$ is the complicating variable or coupling variable.

- When $\mathbf{y}$ is fixed the problem is separable in $\mathbf{x}_1$ and $\mathbf{x}_2$ and then decouples into two subproblems that can be solved independently.

- $\mathbf{x}_1$ and $\mathbf{x}_2$ are local or private variables; and $\mathbf{y}$ can be interpreted as a global or public variable that serves as an interface or boundary variable between the two subproblems.

Primal Decomposition (II)

- For a given fixed $\mathbf{y}$ we define the *subproblems*:

$$\begin{array}{lll} \text{subproblem } 1: & \text{minimize}_{\mathbf{x}_1 \in \mathcal{X}_1} & f_1(\mathbf{x}_1, \mathbf{y}) \\ \text{subproblem } 2: & \text{minimize}_{\mathbf{x}_2 \in \mathcal{X}_2} & f_2(\mathbf{x}_2, \mathbf{y}) \end{array}$$

  with optimal values $f_1^{\star}(\mathbf{y})$ and $f_2^{\star}(\mathbf{y})$.

- Since $\min_{\mathbf{x},\mathbf{y}} f \equiv \min_{\mathbf{y}} \min_{\mathbf{x}} f$, it follows that the original problem is equivalent to the *master primal problem*:

$$\text{minimize}_{\mathbf{y} \in \mathcal{Y}} \quad f_1^{\star}(\mathbf{y}) + f_2^{\star}(\mathbf{y}).$$

Primal Decomposition (III)

Observations:

- subproblems can be solved independently for a given $\mathbf{y}$
- we don't have a closed-form expression for each function $f_i^{\star}(\mathbf{y})$ and its corresponding gradient or subgradient (differentiable?)
- instead, to evaluate each function $f_i^{\star}(\mathbf{y})$ at some point $\mathbf{y}$ we need to solve an optimization problem
- interestingly, we can easily obtain a subgradient of $f_i^{\star}(\mathbf{y})$ "for free" when we evaluate the function.

Primal Decomposition: Solving the Master Problem

- If the original problem is convex, so is master problem.
- To solve the master problem, we can use different methods such as
  - bisection (if y is scalar)
  - gradient or Newton method (if $f_i^\star$ differentiable)
  - subgradient, cutting-plane, or ellipsoid method.
- The subgradient method is very simple and allows itself to distributed implementation; however, its converge is slow in practice.
- Projected subgradient method:

$$\mathbf{y}(k+1) = \left[\mathbf{y}(k) - \alpha_k \left(\mathbf{s}_1(k) + \mathbf{s}_2(k)\right)\right]_{\mathcal{Y}}$$

  where $\mathbf{s}_i(k)$ is a subgradient of $f_i^\star\left(\mathbf{y}\left(k\right)\right)$.

Primal Decomposition Algorithm

repeat

1. Solve the suproblems:

   Find $\mathbf{x}_1(k) \in \mathcal{X}_1$ that minimizes $f_1\left(\mathbf{x}_1(k), \mathbf{y}(k)\right)$, and
   a subgradient $\mathbf{s}_1(k) \in \partial f_1^{\star}\left(\mathbf{y}(k)\right)$.

   Find $\mathbf{x}_2(k) \in \mathcal{X}_2$ that minimizes $f_2\left(\mathbf{x}_2(k), \mathbf{y}(k)\right)$, and
   a subgradient $\mathbf{s}_2(k) \in \partial f_2^{\star}\left(\mathbf{y}(k)\right)$.

2. Update the complicating variable to minimize the primal master subproblem:

$$\mathbf{y}(k+1) = \left[\mathbf{y}(k) - \alpha_k\left(\mathbf{s}_1(k) + \mathbf{s}_2(k)\right)\right]_{\mathcal{Y}}.$$

3. $k = k + 1$

Primal Decomposition: Subgradients

Lemma: Let $f^{\star}(\mathbf{y})$ be the optimal value of the convex problem

$$\begin{array}{ll} \underset{\mathbf{x}}{\text{minimize}} & f_0(\mathbf{x}) \\ \text{subject to} & h_i(\mathbf{x}) \leq y_i, \quad i = 1, \cdots, m. \end{array}$$

A subgradient of $f^{\star}(\mathbf{y})$ is $-\boldsymbol{\lambda}^{\star}(\mathbf{y})$.

Proof. The Lagrangian is $L(\mathbf{x}, \boldsymbol{\lambda}) = f_0(\mathbf{x}) + \boldsymbol{\lambda}^T(\mathbf{h}(\mathbf{x}) - \mathbf{y})$. Then,

$$\begin{array}{rcl} f^{\star}(\mathbf{y}_0) & = & f_0(\mathbf{x}^{\star}(\mathbf{y}_0)) \\ & = & g(\boldsymbol{\lambda}^{\star}(\mathbf{y}_0)) \\ & \leq & L(\mathbf{x}, \boldsymbol{\lambda}^{\star}(\mathbf{y}_0)) = f_0(\mathbf{x}) + \boldsymbol{\lambda}^{\star T}(\mathbf{h}(\mathbf{x}) - \mathbf{y}_0) \\ & = & f_0(\mathbf{x}) + \boldsymbol{\lambda}^{\star T}(\mathbf{h}(\mathbf{x}) - \mathbf{y}) + \boldsymbol{\lambda}^{\star T}(\mathbf{y} - \mathbf{y}_0) \\ & \leq & f_0(\mathbf{x}) + \boldsymbol{\lambda}^{\star T}(\mathbf{y} - \mathbf{y}_0) \qquad\qquad \forall \mathbf{y} \end{array}$$

where the last inequality holds for any $\mathbf{x}$ such that $\mathbf{h}(\mathbf{x}) \leq \mathbf{y}$. In particular,

$$
\begin{aligned}
f^{\star}(\mathbf{y}_0) &\leq \min_{\mathbf{h}(\mathbf{x}) \leq \mathbf{y}} f_0(\mathbf{x}) + \boldsymbol{\lambda}^{\star T}(\mathbf{y} - \mathbf{y}_0) \\
&= f^{\star}(\mathbf{y}) + \boldsymbol{\lambda}^{\star T}(\mathbf{y} - \mathbf{y}_0)
\end{aligned}
$$

or

$$
f^{\star}(\mathbf{y}) \geq f^{\star}(\mathbf{y}_0) - \boldsymbol{\lambda}^{\star T}(\mathbf{y} - \mathbf{y}_0).
$$

$\square$

Lemma: Let $f^{\star}(\mathbf{y})$ be the optimal value of the convex problem

$$
\begin{array}{ll}
\underset{\mathbf{x}}{\text{minimize}} & f_0(\mathbf{x}, \mathbf{y}) \\
\text{subject to} & h_i(\mathbf{x}) \leq y_i, \quad i = 1, \cdots, m.
\end{array}
$$

A subgradient of $f^{\star}(\mathbf{y})$ is $(\mathbf{s}_0(\mathbf{x}^{\star}(\mathbf{y}), \mathbf{y}) - \boldsymbol{\lambda}^{\star}(\mathbf{y}))$.

Dual Decomposition

- Consider the following problem with a coupling constraint:

$$\begin{array}{ll} \underset{\mathbf{x}}{\text{minimize}} & f_1(\mathbf{x}_1) + f_2(\mathbf{x}_2) \\ \text{subject to} & \mathbf{x}_1 \in \mathcal{X}_1, \qquad \mathbf{x}_2 \in \mathcal{X}_2 \\ & \mathbf{h}_1(\mathbf{x}_1) + \mathbf{h}_2(\mathbf{x}_2) \leq \mathbf{h}_0 \end{array}$$

- $\mathbf{h}_1(\mathbf{x}_1) + \mathbf{h}_2(\mathbf{x}_2) \leq \mathbf{h}_0$ is the complicating or coupling constraint.

- If the coupling constraint is relaxed with a Lagrange multiplier $\boldsymbol{\lambda}$, then the problem decouples into two subproblems that can be solved independently.

- $\mathbf{x}_1$ and $\mathbf{x}_2$ are local or private variables; and $\boldsymbol{\lambda}$ can be interpreted as a global or public price variable that serves as an interface or boundary variable between the two subproblems.

Dual Decomposition (II)

- The partial Lagrangian after relaxing the coupling constraint is:

$$L(\mathbf{x}, \boldsymbol{\lambda}) = f_1(\mathbf{x}_1) + f_2(\mathbf{x}_2) + \boldsymbol{\lambda}^T(\mathbf{h}_1(\mathbf{x}_1) + \mathbf{h}_2(\mathbf{x}_2) - \mathbf{h}_0).$$

- The dual function is

$$\begin{aligned} g(\boldsymbol{\lambda}) &= \inf_{\mathbf{x} \in \mathcal{X}} L(\mathbf{x}, \boldsymbol{\lambda}) \\ &= \inf_{\mathbf{x}_1 \in \mathcal{X}_1} \left\{ f_1(\mathbf{x}_1) + \boldsymbol{\lambda}^T \mathbf{h}_1(\mathbf{x}_1) \right\} \\ &\quad + \inf_{\mathbf{x}_2 \in \mathcal{X}_2} \left\{ f_2(\mathbf{x}_2) + \boldsymbol{\lambda}^T \mathbf{h}_2(\mathbf{x}_2) \right\} - \boldsymbol{\lambda}^T \mathbf{h}_0 \end{aligned}$$

which clearly decouples.

Dual Decomposition (III)

- For a given fixed $\boldsymbol{\lambda}$ we define the *subproblems*:

$$\begin{array}{lll} \text{subproblem } 1: & \text{minimize}_{\mathbf{x}_1 \in \mathcal{X}_1} & f_1(\mathbf{x}_1) + \boldsymbol{\lambda}^T \mathbf{h}_1(\mathbf{x}_1) \\ \text{subproblem } 2: & \text{minimize}_{\mathbf{x}_2 \in \mathcal{X}_2} & f_2(\mathbf{x}_2) + \boldsymbol{\lambda}^T \mathbf{h}_2(\mathbf{x}_2) \end{array}$$

  with optimal values $g_1(\boldsymbol{\lambda})$ and $g_2(\boldsymbol{\lambda})$.

- From strong duality, the original problem is equivalent to the *master dual problem*:

$$\text{maximize}_{\boldsymbol{\lambda} \geq \mathbf{0}} \quad g_1(\boldsymbol{\lambda}) + g_2(\boldsymbol{\lambda}) - \boldsymbol{\lambda}^T \mathbf{h}_0.$$

Dual Decomposition (IV)

Observations:

- subproblems can be solved independently for a given $\boldsymbol{\lambda}$
- we don't have a closed-form expression for each function $g_i(\boldsymbol{\lambda})$ and its corresponding gradient or subgradient (differentiable?)
- instead, to evaluate each function $g_i(\boldsymbol{\lambda})$ at some point $\boldsymbol{\lambda}$ we need to solve an optimization problem
- as in the primal case, we can easily obtain a subgradient of $g_i(\boldsymbol{\lambda})$ "for free" when we evaluate the function.

Dual Decomposition: Solving the Master Problem

- The dual master problem is always convex regardless of the original problem. However, we still need convexity to have strong duality (under some constraint qualifications like Slater's condition).

- To solve the master problem, we can use different methods such as
  - bisection (if λ is scalar)
  - gradient or Newton method (if g_i differentiable)
  - subgradient, cutting-plane, or ellipsoid method.

- Projected subgradient method:

$$\boldsymbol{\lambda}(k+1) = \left[\boldsymbol{\lambda}(k) + \alpha_k \left(\mathbf{s}_1(k) + \mathbf{s}_2(k) - \mathbf{h}_0\right)\right]^+$$

  where $\mathbf{s}_i(k)$ is a subgradient of $g_i\left(\boldsymbol{\lambda}\left(k\right)\right)$.

Dual Decomposition Algorithm

repeat

1. Solve the suproblems:

   Find $\mathbf{x}_1(k) \in \mathcal{X}_1$ that minimizes $f_1\left(\mathbf{x}_1(k)\right) + \boldsymbol{\lambda}(k)^T \mathbf{h}_1\left(\mathbf{x}_1(k)\right)$,
   $\quad\quad$ $\text{and a subgradient } \mathbf{s}_1(k) \in \partial g_1\left(\boldsymbol{\lambda}(k)\right)$.

   Find $\mathbf{x}_2(k) \in \mathcal{X}_2$ that minimizes $f_2\left(\mathbf{x}_2(k)\right) + \boldsymbol{\lambda}(k)^T \mathbf{h}_2\left(\mathbf{x}_2(k)\right)$,
   $\quad\quad$ $\text{and a subgradient } \mathbf{s}_2(k) \in \partial g_2\left(\boldsymbol{\lambda}(k)\right)$.

2. Update the complicating variable to minimize the primal master subproblem:

$$\boldsymbol{\lambda}(k+1) = \left[\boldsymbol{\lambda}(k) + \alpha_k\left(\mathbf{s}_1(k) + \mathbf{s}_2(k) - \mathbf{h}_0\right)\right]^+.$$

3. $k = k + 1$

Dual Decomposition: Subgradients

Lemma: Let $g(\boldsymbol{\lambda})$ be the dual function corresponding to the problem

$$\begin{array}{ll} \underset{\mathbf{x}}{\text{minimize}} & f_0(\mathbf{x}) \\ \text{subject to} & h_i(\mathbf{x}) \leq 0, \quad i = 1, \cdots, m. \end{array}$$

A subgradient of $g(\boldsymbol{\lambda})$ is $\mathbf{h}(\mathbf{x}^\star(\boldsymbol{\lambda}))$.

Proof. The Lagrangian is $L(\mathbf{x}, \boldsymbol{\lambda}) = f_0(\mathbf{x}) + \boldsymbol{\lambda}^T \mathbf{h}(\mathbf{x})$ and the dual function $g(\boldsymbol{\lambda}) = \inf_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda})$. Then,

$$\begin{aligned} g(\boldsymbol{\lambda}) &= \inf_{\mathbf{x}} f_0(\mathbf{x}) + \boldsymbol{\lambda}^T \mathbf{h}(\mathbf{x}) \\ &\leq f_0(\mathbf{x}^\star(\boldsymbol{\lambda}_0)) + \boldsymbol{\lambda}^T \mathbf{h}(\mathbf{x}^\star(\boldsymbol{\lambda}_0)) \\ &= f_0(\mathbf{x}^\star(\boldsymbol{\lambda}_0)) + \boldsymbol{\lambda}_0^T \mathbf{h}(\mathbf{x}^\star(\boldsymbol{\lambda}_0)) + (\boldsymbol{\lambda} - \boldsymbol{\lambda}_0)^T \mathbf{h}(\mathbf{x}^\star(\boldsymbol{\lambda}_0)) \\ &= g(\boldsymbol{\lambda}_0) + (\boldsymbol{\lambda} - \boldsymbol{\lambda}_0)^T \mathbf{h}(\mathbf{x}^\star(\boldsymbol{\lambda}_0)). \end{aligned}$$

$\square$

Summary

- We have described the concept of subgradient as a generalization of gradient.

- We have considered subgradient methods as formally similar to gradient methods.

- Finally, we have derived the two basic decomposition techniques:

  – primal decomposition
  – dual decomposition.