9. Dual decomposition and dual algorithms

- dual gradient ascent
- example: network rate control
- dual decomposition and the proximal gradient method
- examples with simple dual prox-operators
- alternating minimization method
Dual methods

convex problem with linear constraints and its dual

minimize \( f(x) \)
subject to \( Gx \leq h \)
\( Ax = b \)

maximize \( g(\lambda, \nu) \)
subject to \( \lambda \geq 0 \)

\( g(\lambda, \nu) = \inf_x \left( f(x) + (G^T \lambda + A^T \nu)^T x - h^T \lambda - b^T \nu \right) \)
\( = -h^T \lambda - b^T \nu - f^*(-G^T \lambda - A^T \nu) \)

potential advantages of solving the dual when using 1-st order methods

- dual is unconstrained or has simple constraints
- dual decomposes into smaller problems
(Sub-)gradients of conjugate function

assume \( f : \mathbb{R}^n \rightarrow \mathbb{R} \) is closed, convex with conjugate

\[
    f^*(y) = \sup_x (y^T x - f(x))
\]

- \( x \in \partial f^*(y) \) if and only if \( x \) maximizes \( y^T x - f(x) \) (p. 6-10)

- if \( f \) is strictly convex, then \( f^* \) is differentiable on \( \text{int dom } f^* \) and

\[
    \nabla f^*(y) = \arg\max_x (y^T x - f(x))
\]

- if \( f \) is strongly convex with parameter \( \mu > 0 \), then \( f^* \) is differentiable, \( \text{dom } f^* = \mathbb{R}^n \), and

\[
    \| \nabla f^*(y) - \nabla f^*(x) \|_2 \leq \frac{1}{\mu} \| x - y \|_2
\]

(see p. 8-7)
Dual gradient method

**primal problem:** (for simplicity, only equality constraints)

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad Ax = b
\end{align*}
\]

**dual problem:** maximize \( g(\nu) \) where

\[
g(\nu) = \inf_x (f(x) + (Ax - b)^T \nu)
\]

**dual ascent:** solve dual by (sub-)gradient method (\( t \) is stepsize)

\[
x^+ = \arg\min_x (f(x) + \nu^T Ax), \quad \nu^+ = \nu + t(Ax^+ - b)
\]

- sometimes referred to as Uzawa’s method
- of interest if calculation of \( x^+ \) is inexpensive
Dual decomposition

convex problem with separable objective

\[
\begin{align*}
\text{minimize} & \quad f_1(x_1) + f_2(x_2) \\
\text{subject to} & \quad G_1 x_1 + G_2 x_2 \preceq h
\end{align*}
\]

cstraint is *complicating or coupling* constraint

dual problem (master problem)

\[
\begin{align*}
\text{maximize} & \quad g_1(\lambda) + g_2(\lambda) - h^T \lambda \\
\text{subject to} & \quad \lambda \succeq 0
\end{align*}
\]

where \( g_j(\lambda) = \inf (f_j(x) + \lambda^T G_j x) = -f_j^*(-G_j^T \lambda) \)

can be solved by (sub-)gradient projection (if \( \lambda \succeq 0 \) is the only constraint)
**subproblem:** to calculate $g_j(\lambda)$ and a (sub-)gradient, solve problem

$$\minimize \ (\text{over } x_j) \quad f_j(x_j) + \lambda^T G_j x_j$$

- optimal value is $g_j(\lambda)$
- if $\hat{x}_j$ solves the subproblem, then $-G_j \hat{x}_j$ is a subgradient of $-g_j$ at $\lambda$

**dual subgradient projection method**

- solve two unconstrained (and independent) subproblems

$$x_j^+ = \arg\min_{x_j} (f_j(x_j) + \lambda^T G_j x_j), \quad j = 1, 2$$

- make projected subgradient update of $\lambda$

$$\lambda^+ = (\lambda + t(G_1 x_1^+ + G_2 x_2^+ - h))_+$$

$$\quad (u_+ = \max\{u, 0\}, \text{componentwise})$$
**interpretation:** price coordination

- \( p = 2 \) units in the system; unit \( j \) selects variable \( x_j \)

- constraints are limits on shared resources; \( \lambda_i \) is price of resource \( i \)

- dual update \( \lambda_i^+ = (\lambda_i - ts_i)_+ \) depends on slacks \( s = h - G_1 x_1 - G_2 x_2 \)
  - increases price \( \lambda_i \) if resource is over-used \( (s_i < 0) \)
  - decreases price \( \lambda_i \) if resource is under-used \( (s_i > 0) \)
  - never lets price get negative

**distributed architecture**

- central node 0 sets price \( \lambda \)

- peripheral node \( j \) sets \( x_j \)
Example: network rate control

- $n$ flows (with fixed routes) in a network with $m$ links
- variable $x_j \geq 0$ denotes rate of flow $j$
- utility function for flow $j$ is $U_j : \mathbb{R} \rightarrow \mathbb{R}$, concave, increasing

capacity constraints

- traffic $y_i$ on link $i$ is sum of flows passing through it
- $y = Rx$, where $R$ is the routing matrix

$$R_{ij} = \begin{cases} 
1 & \text{flow } j \text{ passes through link } i \\
0 & \text{otherwise}
\end{cases}$$

- link capacity constraint: $y \leq c$
maximize \[ U(x) = \sum_{j=1}^{n} U_j(x_j) \]
subject to \[ Rx \preceq c \]

a convex problem; dual decomposition gives decentralized method

**Lagrangian** (for minimizing \(-U\))

\[ L(x, \lambda) = -U(x) + \lambda^T (Rx - c) = -\lambda^T c + \sum_{j=1}^{n} (-U_j(x_j) + x_j r_j^T \lambda) \]

- \(\lambda_i\) is the price (per unit flow) for using link \(i\)
- \(r_j^T \lambda\) is the sum of prices along route \(j\) (\(r_j\) is \(j\)th column of \(R\))

**dual function**

\[ g(\lambda) = -\lambda^T c + \sum_{j=1}^{n} \inf_{x_j} (-U_j(x_j) + x_j r_j^T \lambda) = -\lambda^T c - \sum_{j=1}^{n} (-U_j)^* (-r_j^T \lambda) \]
(Sub-)gradients of dual function

\[ g(\lambda) = -\lambda^T c - \sum_{j=1}^{n} \sup_{x_j} (U_j(x_j) - x_j r_j^T \lambda) \]

- subgradient of \(-g(\lambda)\)

\[ c - R\bar{x} \in \partial(-g)(\lambda) \quad \text{where} \quad \bar{x}_j = \arg\max (U_j(x_j) - x_j r_j^T \lambda) \]

if \(U_j\) is strictly concave, this is a gradient

- \(r_j^T \lambda\) is the sum of link prices along route \(j\)

- \(c - R\bar{x}\) is vector of link capacity margins for flow \(\bar{x}\)
Dual decomposition rate control algorithm

given initial link price vector \( \lambda \succ 0 \) (e.g., \( \lambda = 1 \))

repeat

1. sum link prices along each route: calculate \( \Lambda_j = r_j^T \lambda \)

2. optimize flows (separately) using flow prices:

   \[
   x_j^+ := \text{argmax} \left( U_j(x_j) - \Lambda_j x_j \right)
   \]

3. calculate link capacity margins \( s := c - Rx \)

4. update link prices: \( t \) is the step size

   \[
   \lambda := (\lambda - ts)_+
   \]

**decentralized:** links only need to know the flows that pass through them; flows only need to know prices on links they pass through
TCP/AQM congestion control

A large class of internet congestion control mechanisms can be interpreted as distributed algorithms that solve NUM and its dual.

TCP: Reno, Vegas, . . .
AQM: RED, DropTail, . . .

\( x_s \): source rate, updated by TCP (Transmission Control Protocol)
\( \lambda_l \): link congestion measure, or ‘price’, updated by AQM (Active Queue Management)

E.g., TCP Reno uses packet loss as congestion measure, TCP Vegas uses queueing delay.

Refs: [Kelly, et al., '98]; [Low, Lapsley '99]; . . .
Outline

• dual gradient ascent

• example: network rate control

• dual decomposition and dual proximal gradient method

• examples with simple dual prox-operators

• alternating minimization method
First-order dual methods

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad Gx \preceq h \\
& \quad Ax = b
\end{align*}
\]

\[
\begin{align*}
\text{maximize} & \quad -f^*(-G^T\lambda - A^T\nu) \\
\text{subject to} & \quad \lambda \succeq 0
\end{align*}
\]

can apply different algorithms to the dual:

**subgradient method**: slow convergence

**gradient method**: requires differentiable \( f \)

- in many applications \( f^* \) is not differentiable, has a nontrivial domain
- \( f^* \) can be smoothed by adding a small strongly convex term to \( f \)

**proximal gradient method**: dual costs split in two terms

- first term is differentiable; second term has an inexpensive prox-operator
Composite structure in the dual

primal problem with separable objective

\[
\begin{align*}
\text{minimize} & \quad f(x) + h(y) \\
\text{subject to} & \quad Ax + By = b
\end{align*}
\]

(later we consider general problem with inequality constraints)

dual problem

\[
\begin{align*}
\text{maximize} & \quad -f^*(-A^T\nu) - h^*(-B^T\nu) - b^T\nu
\end{align*}
\]

has the composite structure required for the proximal gradient method if

- \(f\) is strongly convex, hence \(\nabla f^*\) is Lipschitz continuous

- prox-operator of \(h^*(-B^T\nu)\) is cheap (closed form or efficient algorithm)
Example: regularized norm approximation

\[ \text{minimize } f(x) + \|Ax - b\| \]

\(f\) is strongly convex with parameter \(\mu\); \(\| \cdot \|\) is any norm

(reformulated) problem and dual

\[ \text{minimize } f(x) + \|y\| \quad \text{maximize } b^T z - f^*(A^T z) \]
\[ \text{subject to } y = Ax - b \quad \text{subject to } \|z\|_* \leq 1 \]

\(\cdot\) gradient of dual cost is Lipschitz continuous with parameter \(\|A\|_2^2/\mu\)

\(\cdot\) for most norms, projection on norm ball is inexpensive
**dual gradient projection step** (with \( C = \{ v \mid \| v \|_* \leq 1 \} \))

\[
z^+ = P_C \left( z + t(b - A \nabla f^*(A^T z)) \right)
\]

where \( \nabla f^*(A^T z) = \arg\min_x (f(x) - z^T A x) \)

**gradient projection algorithm:** choose initial \( z \) and repeat

\[
\hat{x} := \arg\min_x (f(x) - z^T A x)
\]

\[
z := P_C (z + t(b - A \hat{x}))
\]

- step size \( t \): constant or from backtracking line search
- can also use accelerated gradient projection algorithm
Example: regularized nuclear norm approximation

minimize \( \frac{1}{2} \| x - a \|_2^2 + \| A(x) - B \|_* \)

\( \| \cdot \|_* \) is nuclear norm and \( A : \mathbb{R}^n \to \mathbb{R}^{p \times q} \) with \( A(x) = \sum_{i=1}^n x_i A_i \)

**gradient projection:** choose initial \( Z \) and repeat

\[
\hat{x}_i := a_i + \text{tr}(A_i^T Z), \quad i = 1, \ldots, n \\
Z := P_C(Z + t(B - A(\hat{x})))
\]

- \( \hat{x} \) is minimizer of \( (1/2) \| x - a \|_2^2 - \sum_i x_i \text{tr}(A_i^T Z) \)
- \( C \) is unit ball for matrix norm \( \| V \| = \sigma_{\text{max}}(V) \)
- to find \( P_C(V) \), replace \( \sigma_i \) by \( \min\{ \sigma_i, 1 \} \) in SVD of \( V \)
Example: dual decomposition

minimize \( f(x) + \sum_{i=1}^{p} \|B_i x\|_2 \)

with \( f \) strongly convex, \( B_i \in \mathbb{R}^{m_i \times n} \)

reformulated problem

minimize \( f(x) + \sum_{i=1}^{p} \|y_i\|_2 \)
subject to \( y_i = B_i x, \quad i = 1, \ldots, p \)

objective is separable, but not strictly convex

dual problem

maximize \( -f^* \left( \sum_{i=1}^{p} B_i^T z_i \right) \)
subject to \( \|z_i\|_2 \leq 1, \quad i = 1, \ldots, p \)
dual gradient projection step (with \( C_i = \{ v \in \mathbb{R}_{i}^{m} \mid \| v \|_2 \leq 1 \} \))

\[
\hat{z}_i^+ = P_{C_i} \left( z_i - tB_i \nabla f^* \left( \sum_{i=1}^{p} B_i^T z_i \right) \right), \quad i = 1, \ldots, p
\]

algorithm: choose initial \( z_i \) and repeat

\[
z := \sum_{i=1}^{p} B_i^T z_i \\
\hat{x} := \arg\min_x (f(x) - z^T x) \quad (= \nabla f^*(z))
\]

\[
z_i := P_{C_i}(z_i - tB_i \hat{x}), \quad i = 1, \ldots, p
\]

• updates of \( z_i \) are independent

• if \( f \) is separable, primal update decomposes into independent subproblems
Minimization over intersection of convex sets

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad x \in C_1 \cap \ldots \cap C_m
\end{align*}
\]

- \( f \) strongly convex; \( C_i \) closed, convex with inexpensive projector

- example: \( f(x) = \|x - a\|_2^2 \) gives projection of \( a \) on intersection

**reformulation:** introduce auxiliary variables \( x_i \)

\[
\begin{align*}
\text{minimize} & \quad f(x) + I_{C_1}(x_1) + \ldots + I_{C_m}(x_m) \\
\text{subject to} & \quad x_1 = x, \ldots, x_m = x
\end{align*}
\]

**dual problem**

\[
\begin{align*}
\text{maximize} & \quad -f^*(z_1 + \ldots + z_m) - h_1(z_1) - \ldots - h_m(z_m) \\
\end{align*}
\]

\( h_i(z) = \sup_{x \in C_i} (-z^T x) \) is support function of \( C_i \) at \(-z\)
dual proximal gradient step

\[ z_i^+ = \text{prox}_{th_i}(z_i - t\nabla f^*(z_1 + \ldots + z_m)), \quad i = 1, \ldots, m \]

prox-operator of \( h_i \) can be expressed in terms of projection on \( C_i \)

\[ \text{prox}_{th_i}(u) = u + tP_{C_i}(-u/t) \]

dual proximal gradient algorithm: choose initial \( z_1, \ldots, z_m \) and repeat

\[ \hat{x} := \arg\min_x (f(x) - (z_1 + \ldots + z_m)^T x) \]

\[ z_i := z_i + t \left( P_{C_i}(\hat{x} - \frac{1}{t}z_i) - \hat{x} \right), \quad i = 1, \ldots, m \]

can take \( t = \mu/m \) (\( \mu \) is strong convexity parameter of \( f \))
Outline

• dual gradient ascent

• network rate control (utility maximization)

• dual decomposition and dual proximal gradient method

• examples with simple dual prox-operators

• alternating minimization method
Prox-operator of partial dual

\[
\begin{align*}
\text{minimize} & \quad f(x) + h(y) \\
\text{subject to} & \quad Ax + By = b
\end{align*}
\]

\[
\begin{align*}
\text{minimize} & \quad -f^*(-A^T \nu) - F(\nu)
\end{align*}
\]

• \(F\) is negative of a ‘partial dual function’

\[
F(\nu) = b^T \nu + h^*(-B^T \nu)
\]

\[
\quad = - \inf_x (h(y) + \nu^T (By - b))
\]

• prox-operator of \(F\) is defined as

\[
\text{prox}_{tF}(\nu) = \arg\min_\nu \left( F(\nu) + \frac{1}{2t} \|\nu - \nu\|^2 \right)
\]
Primal expression for prox-operator

• by definition, \( v = \operatorname{prox}_{tF}(\nu) \) is the minimizer \( v \) of

\[
b^T v + h^*(-B^T v) + \frac{1}{2t} \|v - \nu\|^2_2
\]

• this is the dual of the problem (with variables \( y, z \))

\[
\text{maximize } -h(y) - \nu^T z - \frac{t}{2} \|z\|^2_2, \quad \text{subject to } By - b = z
\]

• primal and dual optimal solutions are related by \( v = \nu + t(By - b) \)

**conclusion:** primal method for computing \( v = \operatorname{prox}_{tF}(\nu) \)

\[
\hat{y} = \arg\min \left( h(y) + \nu^T (By - b) + \frac{t}{2} \|By - b\|^2_2 \right), \quad v = \nu + t(B\hat{y} - b)
\]

\( \hat{y} \) minimizes **augmented Lagrangian** (Lagrangian + quadratic penalty)
Alternating minimization method

\[
\text{minimize} \quad f(x) + h(y) \quad \text{minimize} \quad -f^*(-A^T\nu) - F(\nu)
\]
\[
\text{subject to} \quad Ax + By = b
\]

\( f \) strongly convex; \( h \) convex, not necessarily strictly

dual proximal gradient step

\[
\nu^+ = \text{prox}_{tF}(\nu + tA\nabla f^*(-A^T\nu))
\]

\( \hat{x} = \nabla f^*(-A^T\nu) \) is minimizer of \( f(x) + \nu^T Ax \)

\( \text{prox}_{tF}(\nu + tA\hat{x}) = \nu + t(A\hat{x} + B\hat{y} - b) \) where \( \hat{y} \) minimizes

\[
h(y) + (\nu + tA\hat{x})^T(By - b) + \frac{t}{2}\|By - b\|_2^2
\]
**algorithm:** choose initial $\nu$ and repeat

\[
\hat{x} := \arg\min_x \left( f(x) + \nu^T A x \right)
\]

\[
\hat{y} := \arg\min_y \left( h(y) + \nu^T B y + \frac{t}{2} \| A \hat{x} + B y - b \|_2^2 \right)
\]

\[
\nu := \nu + t(A \hat{x} + B \hat{y} - b)
\]

- alternating minimization of
  - Lagrangian (step 1)
  - augmented Lagrangian (step 2)

- step 3 is proximal gradient update for the dual problem

- as a variation, can use accelerated proximal gradient method
General problem with separable objective

minimize \( f(x) + h(y) \)
subject to \( Ax + By = b \)
\( Cx + Dy \preceq d \)

\( f \) strongly convex

dual problem

maximize \( -f^*(-C^T \lambda - A^T \nu) - F(\lambda, \nu) \)

where

\[
F(\lambda, \nu) = \begin{cases} 
  d^T \lambda + b^T \nu + h^*(-D^T \lambda - B^T \nu), & \lambda \succeq 0 \\
  +\infty, & \text{otherwise}
\end{cases}
\]

we derive expressions for the prox-operator of \( F \)
Proximal operator of partial dual function

definition: \((u, v) = \text{prox}_t F(\lambda, \nu)\) is the solution of

\[
\minimize F(u, v) + \frac{1}{2t}(\|u - \lambda\|_2^2 + \|v - \nu\|_2^2)
\]

equivalent expression

\[
\begin{bmatrix}
  u \\
  v
\end{bmatrix} = \begin{bmatrix}
  \lambda \\
  \nu
\end{bmatrix} + t \begin{bmatrix}
  D\hat{y} + \hat{s} - d \\
  B\hat{y} - b
\end{bmatrix}
\]

where \(\hat{y}, \hat{s}\) solve

\[
\minimize h(y) + \lambda^T(Dy + s) + \nu^TBy + \frac{1}{2t}(\|Dy + s - d\|_2^2 + \|By - b\|_2^2)
\]
subject to \(s \geq 0\)
proof: follows from the duality between the problems

\[
\begin{align*}
\text{minimize}_{x,s,w,z} & \quad h(y) + \lambda^T w + \nu^T z + \frac{1}{2t}(\|w\|_2^2 + \|z\|_2^2) \\
\text{subject to} & \quad Dy + s - d = w \\
& \quad By - b = z \\
& \quad s \succeq 0
\end{align*}
\]

and

\[
\begin{align*}
\text{maximize}_{u,v} & \quad -d^T u - b^T v - h^*(-D^T u - B^T v) - \frac{1}{2t}(\|u - \lambda\|_2^2 + \|v - \nu\|_2^2) \\
\text{subject to} & \quad u \succeq 0
\end{align*}
\]

- at the optimum,

\[
\lambda + t(Dy + s - d) = u, \quad \nu + t(By - b) = v
\]

- by definition the optimal \((u, v)\) is the proximal operator \(\text{prox}_{tF}(\lambda, \nu)\)
Alternating minimization method

choose initial $\lambda$, $\nu$ and repeat

1. compute the minimizer $\hat{x}$ of the Lagrangian

$$f(x) + (A^T\nu + C^T\lambda)^T x$$

2. compute the minimizers $\hat{y}$, $\hat{s}$ of the augmented Lagrangian

$$h(y) + \lambda^T(Dy + s) + \nu^TBy + \frac{t}{2} (\|C\hat{x} + Dy + s - d\|_2^2 + \|A\hat{x} + By - b\|_2^2)$$

subject to $s \succeq 0$

3. dual update

$$\lambda := \lambda + t(C\hat{x} + D\hat{y} - \hat{s} - d), \quad \nu := \nu + t(A\hat{x} + B\hat{y} - b)$$

as a variation, can use a fast proximal gradient update
References and sources

- L. Vandenberghe, *Lecture notes for EE236C - Optimization Methods for Large-Scale Systems* (Spring 2011), UCLA.
- S. Boyd, course notes for EE364b, Convex Optimization II (the rate control example)