

# 14. Dual methods II

- single commodity network flow
- augmented Lagrangian method

14-1

## Single commodity network flow

### network

- connected, directed graph with  $n$  links,  $p$  nodes
- node incidence matrix  $A \in \mathbf{R}^{p \times n}$  is

$$A_{ij} = \begin{cases} 1 & \text{arc } j \text{ enters } i \\ -1 & \text{arc } j \text{ leaves node } i \\ 0 & \text{otherwise} \end{cases}$$

### flow vector and external sources

- variable  $x_j$  denotes flow (traffic) on arc  $j$
- given external source (or sink) flow  $b_i$  at node  $i$ ,  $\mathbf{1}^T b = 0$
- flow conservation:  $Ax + b = 0$

## Network flow optimization problem

$$\begin{aligned} & \text{minimize} && \sum_{j=1}^n \phi_j(x_j) \\ & \text{subject to} && Ax + b = 0 \end{aligned}$$

$\phi(x) = \sum_{j=1}^n \phi_j(x_j)$  is separable convex flow cost function

- convex, readily solved with standard methods
- dual decomposition yields decentralized solution method

## Network flow dual

### Lagrangian

$$\begin{aligned} L(x, \nu) &= \phi(x) + \nu^T (Ax + b) \\ &= b^T \nu + \sum_{j=1}^n (\phi_j(x_j) + (a_j^T \nu) x_j) \end{aligned}$$

- $a_j$  is  $j$ th column of  $A$
- we'll interpret  $\nu_i$  as potential at node  $i$
- $y_j = -a_j^T \nu$  is the potential difference across edge  $j$  (potential at start node minus potential at end node)

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

$$g(\nu) = \inf_x L(x, \nu) = b^T \nu - \sum_{j=1}^n \phi_j^*(-a_j^T \nu)$$

## Recovering primal from dual

- strictly convex  $\phi_j$  means unique minimizer

$$x_j^*(y) = \operatorname{argmin}_{x_j} (\phi_j(x_j) - yx_j)$$

- if  $\phi_j$  is differentiable,  $x_j^*(y) = (\phi_j')^{-1}(y)$  (inverse of derivative function)
- optimal flows, from optimal potentials, are  $x_j^*(y_j^*)$  where  $y^* = -A^T v^*$
- subgradient of negative dual function  $-g$  at  $v$ :

$$-(Ax^*(y) + b) \quad \text{where} \quad y = -A^T v$$

subgradient is negative of flow conservation residual

## Dual decomposition network flow algorithm

**given** initial potential vector  $v$

**repeat**

1. determine link flows from potential differences  $y = -A^T v$

$$x_j := x_j^*(y_j), \quad j = 1, \dots, n$$

2. compute flow surplus at each node

$$s_i := a_i^T x + b_i, \quad i = 1, \dots, p$$

3. update node potentials

$$v_i := v_i + t s_i, \quad i = 1, \dots, p$$

$t$  is an appropriate step size

# Dual decomposition network flow algorithm

- decentralized:
  - flow calculated from potential difference across edge
  - node potential updated from its own flow surplus
- $g(\nu)$  gives lower bound on  $p^*$
- flow conservation  $Ax + b = 0$  only holds in limit

## Electrical network interpretation

network flow optimality conditions (with differentiable  $\phi_j$ )

$$Ax + b = 0, \quad y + A^T \nu = 0, \quad y_j = \phi'_j(x_j), \quad j = 1, \dots, n$$

network with node incidence matrix  $A$ , nonlinear resistors in branches

**Kirchhoff current law (KCL):**  $Ax + b = 0$

$x_j$  is the current flow in branch  $j$ ;  $b_i$  is external current injected at node  $i$

**Kirchhoff voltage law (KVL):**  $y + A^T \nu = 0$

$\nu_j$  is node potential;  $y_j = -a_j^T \nu$  is  $j$ th branch voltage

**current-voltage characteristics:**  $y_j = \phi'_j(x_j)$

for example,  $\phi_j(x_j) = R_j x_j^2 / 2$  for linear resistor  $R_j$

current and potentials in circuit are optimal flows and dual variables

## Example: minimum queueing delay

flow cost function

$$\phi_j(x_j) = \frac{x_j}{c_j - x_j}, \quad \text{dom } \phi_j = [0, c_j)$$

where  $c_j > 0$  are given *link capacities*

conjugate

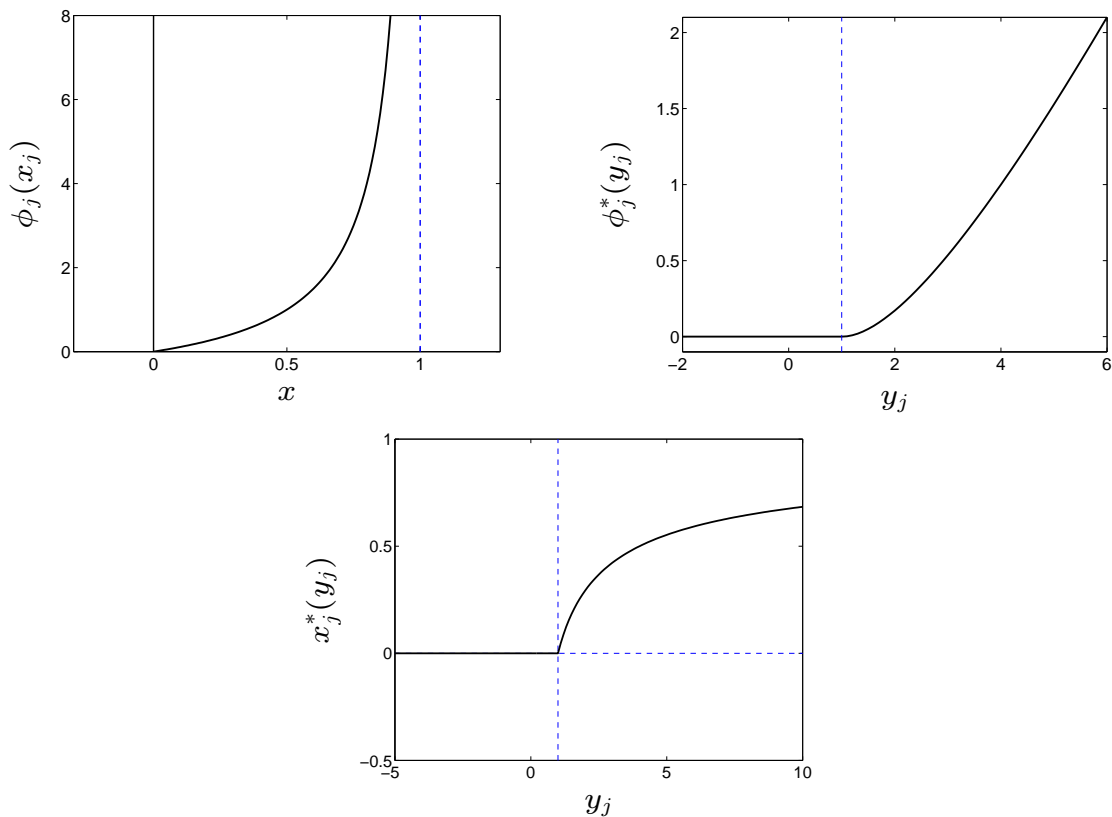
$$\phi_j^*(y_j) = \begin{cases} (\sqrt{c_j y_j} - 1)^2 & y_j > 1/c_j \\ 0 & y_j \leq 1/c_j \end{cases}$$

inverse derivative map

$$\phi_j'(x_j) = y_j \iff x_j = x_j^*(y_j) = c_j - \sqrt{c_j/y_j}$$

Dual methods II

14-9

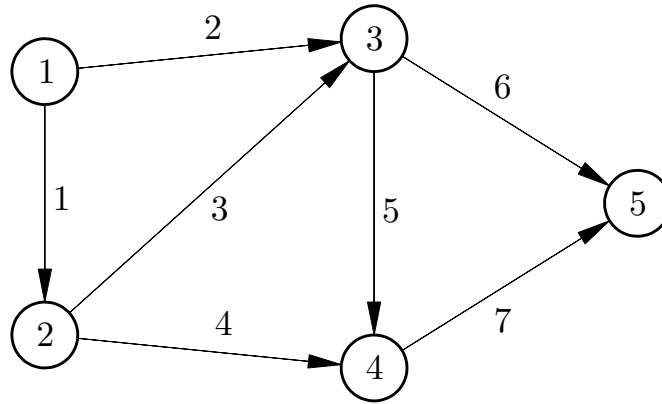


Dual methods II

14-10

## A specific example

network with 5 nodes, 7 links, capacities  $c_j = 1$

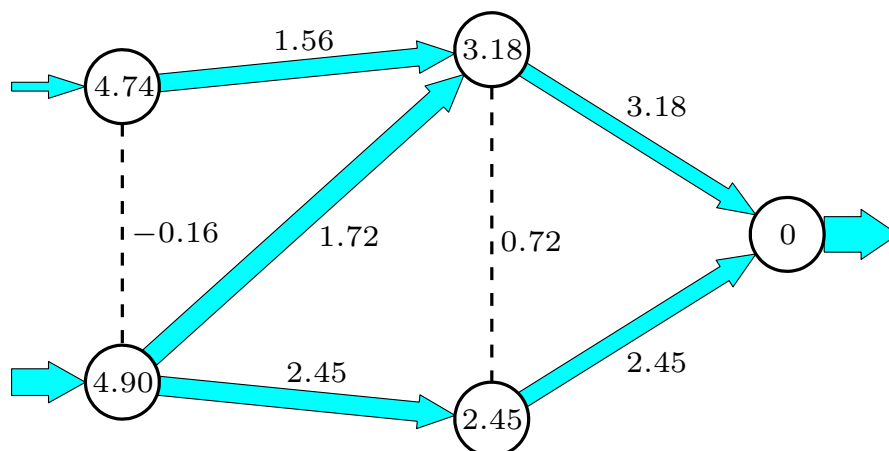


Dual methods II

14-11

## Optimal flow

optimal flows shown as width of arrows; optimal dual variables shown in nodes; potential differences shown on links

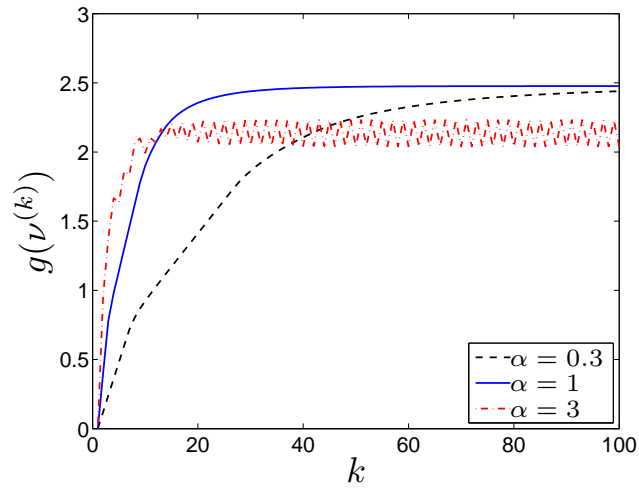


Dual methods II

14-12

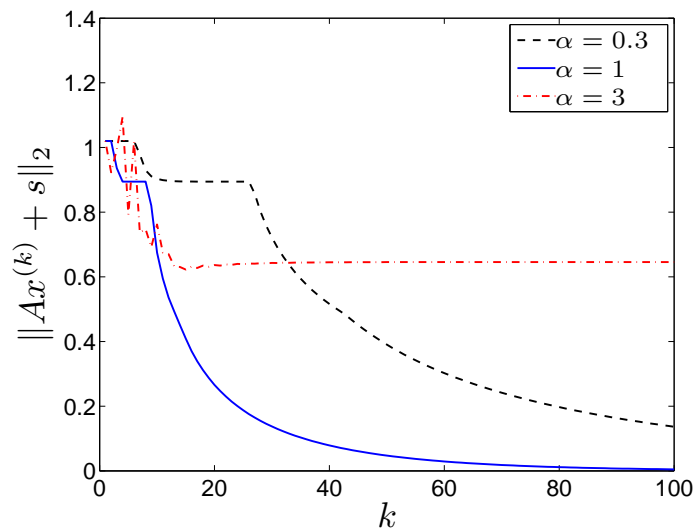
## Convergence of dual function

fixed step size rules,  $\alpha = 0.3, 1, 3$



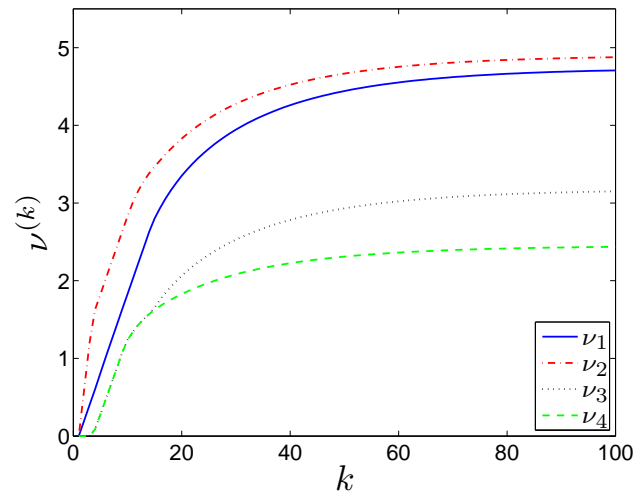
for  $\alpha = 1$ , converges to  $p^* = 2.48$  in around 40 iterations

## Convergence of primal residual



## Convergence of dual variables

$\nu^{(k)}$  versus iteration number  $k$ , fixed step size rule  $\alpha = 1$



( $\nu_5$  is fixed as zero)

## Proximal approximation

$$\begin{aligned} &\text{minimize} && f(x) \\ &\text{subject to} && x \in C \end{aligned}$$

$f$  convex, continuous, not necessarily differentiable  
 $C \subseteq \text{dom } f$  is closed, convex, nonempty

**equivalent formulation:** with  $\mu > 0$ ,

$$\text{minimize } f_\mu(x) \quad \text{where } f_\mu(x) \triangleq \inf_{y \in C} \left( f(y) + \frac{1}{2\mu} \|x - y\|_2^2 \right)$$

- equivalence follows from switching order of minimization over  $x$  and  $y$
- $x^*$  minimizes  $f_\mu(x)$  if and only if  $x^*$  minimizes  $f(x)$  over  $x \in C$
- $f_\mu$  is called *proximal approximation* of  $f$

## Convexity of proximal approximation

$$f_\mu(x) = \inf_{y \in C} \left( f(y) + \frac{1}{2\mu} \|x - y\|_2^2 \right)$$

- $f_\mu$  is convex (infimum over  $y$  of a jointly convex function of  $x, y$ )
- domain of  $f_\mu$  is  $\mathbf{R}^n$
- equivalent expression from duality

$$f_\mu(x) = \sup_{\lambda} \left( x^T \lambda - f^*(\lambda) - \frac{\mu}{2} \|\lambda\|_2^2 \right)$$

where  $f^*$  is the conjugate of  $f$ :

$$f^*(z) = \sup_{y \in C} (z^T y - f(y))$$

in other words,  $f_\mu$  is conjugate of  $f^*(\lambda) + (\mu/2)\|\lambda\|_2^2$

## Gradient of proximal approximation

- $f_\mu$  is differentiable with gradient

$$\nabla f_\mu(x) = \frac{1}{\mu} (x - \hat{y}), \quad \hat{y} = \operatorname{argmin}_{y \in C} \left( f(y) + \frac{1}{2\mu} \|x - y\|_2^2 \right)$$

the mapping  $\operatorname{prox}(x) : x \mapsto \hat{y}$  is called the *proximal mapping*

- alternative expression from dual

$$\nabla f_\mu(x) = \hat{\lambda}, \quad \hat{\lambda} = \operatorname{argmax}_{\lambda} \left( x^T \lambda - f^*(\lambda) - \frac{\mu}{2} \|\lambda\|_2^2 \right)$$

in other words,  $\operatorname{prox}(x) = x - \mu \hat{\lambda}$

- $\nabla f_\mu$  is Lipschitz continuous with constant  $1/\mu$

## Proximal algorithm

minimize  $f_\mu$  by gradient method with step  $\mu$ : for  $k = 1, 2, \dots$

$$\begin{aligned}x^{(k)} &= x^{(k-1)} - \mu \nabla f_\mu(x^{(k-1)}) \\ &= \text{prox}(x^{(k-1)})\end{aligned}$$

**primal form** (from the primal definition of prox)

$$x^{(k)} = \underset{y \in C}{\text{argmin}} \left( f(y) + \frac{1}{2\mu} \|y - x^{(k-1)}\|_2^2 \right), \quad k = 1, 2, \dots$$

**dual form** (from the dual definition of prox)

$$x^{(k)} = x^{(k-1)} - \mu \underset{\lambda}{\text{argmax}} \left( x^T \lambda - f^*(\lambda) - \frac{\mu}{2} \|\lambda\|_2^2 \right), \quad k = 1, 2, \dots$$

## Convex problem with equality constraints

$$\begin{aligned}\text{minimize} & \quad f(x) \\ \text{subject to} & \quad Ax = b\end{aligned}$$

- we don't assume  $f$  is strictly convex, so  $g$  may not be differentiable
- dual function  $g(\nu) = -b^T \nu - f^*(-A^T \nu)$

**proximal approximation of dual function** (with  $\mu > 0$ )

$$g_\mu(\nu) = \sup_z \left( g(z) - \frac{1}{2\mu} \|z - \nu\|_2^2 \right)$$

- maximizing  $g_\mu(\nu)$  is equivalent to maximizing  $g(\nu)$
- $g_\mu$  is concave and differentiable
- $\nabla g_\mu$  is Lipschitz continuous with constant  $1/\mu$

## Proximal mapping

$$\text{prox}(\nu) = \underset{z}{\text{argmax}} \left( g(z) - \frac{1}{2\mu} \|z - \nu\|_2^2 \right)$$

**primal expression:**  $\text{prox}(\nu) = \nu + \mu(A\hat{x} - b)$  where  $\hat{x}$  solves

$$\text{minimize } f(x) + \nu^T(Ax - b) + \frac{\mu}{2} \|Ax - b\|_2^2$$

this follows from the duality of the problems

$$\begin{array}{ll} \text{minimize}_{x,y} & f(x) + \nu^T y + \frac{\mu}{2} \|y\|_2^2 \\ \text{subject to} & Ax - b = y \end{array} \quad \text{maximize}_z \quad g(z) - \frac{1}{2\mu} \|z - \nu\|_2^2$$

at optimality  $z = \mu y + \nu = \mu(Ax - b) + \nu$

## Gradient of proximal approximation

$$\begin{aligned} g_\mu(\nu) &= \sup_z \left( g(z) - \frac{1}{2\mu} \|\nu - z\|_2^2 \right) \\ &= \inf_x \left( f(x) + \nu^T(Ax - b) + \frac{\mu}{2} \|Ax - b\|_2^2 \right) \end{aligned}$$

**from dual expression**

$$\nabla g_\mu(\nu) = \frac{1}{\mu}(\hat{z} - \nu) = \frac{1}{\mu}(\text{prox}(\nu) - \nu)$$

where  $\hat{z} = \text{prox}(\nu)$  is the maximizer in the definition for  $g_\mu$

**from primal expression**

$$\nabla g_\mu(\nu) = A\hat{x} - b$$

where  $\hat{x}$  is the minimizer in the definition for  $g_\mu$

# Augmented Lagrangian method

**given** initial  $\nu^{(0)}$ ,  $\mu > 0$

**for**  $k = 1, 2, \dots$

1. let  $x^{(k)}$  be the solution of

$$\text{minimize } f(x) + (\nu^{(k-1)})^T (Ax - b) + (\mu/2) \|Ax - b\|_2^2$$

2. take  $\nu^{(k)} = \nu^{(k-1)} + \mu(Ax^{(k)} - b)$

- recall this is the gradient method with step size  $\mu$  applied to

$$\text{maximize } g_\mu(\nu) = \sup_z \left( g(z) - \frac{1}{2\mu} \|z - \nu\|_2^2 \right)$$

- known as the *method of multipliers* or *augmented Lagrangian method* (in step 1 we minimize the Lagrangian augmented with penalty term)

## Applications

augmented Lagrangian method is useful when subproblem

$$\text{minimize } f(x) + \frac{\mu}{2} \|Ax - b - \frac{1}{\mu} \nu\|_2^2$$

is much easier than minimizing  $f(x)$  subject to  $Ax = b$

### examples

- subproblem is parallelizable
- $f(x) = \|x\|_1$ : augmented Lagrangian method is equivalent to the *Bregman iteration* for basis pursuit (see references)

## Example: linear program with complicating constraint

$$\begin{aligned} & \text{minimize} && c_1^T x_1 + c_2^T x_2 \\ & \text{subject to} && A_1 x_1 \preceq b_1, \quad A_2 x_2 \preceq b_2 \\ & && B_1 x_1 + B_2 x_2 = d \end{aligned}$$

$$A_i \in \mathbf{R}^{m_i \times n_i}, B_i \in \mathbf{R}^{p \times n_i} \text{ with } p \ll m_i$$

### dual problem

$$\text{maximize} \quad g_1(\nu) + g_2(\nu) - d^T \nu$$

where

$$g_i(\nu) = \inf_{A_i x \preceq b_i} (B_i^T \nu + c_i)^T x = \sup_{\substack{A_i^T z + B_i^T \nu + c_i = 0 \\ z \succeq 0}} -b_i^T z$$

### dual decomposition via subgradient method

**given** initial  $\nu^{(0)}$

**for**  $k = 1, 2, \dots$

1. let  $x_i^{(k)}$ ,  $i = 1, 2$ , be the solution of

$$\begin{aligned} & \text{minimize} && (B_i^T \nu^{(k-1)} + c_i)^T x \\ & \text{subject to} && A_i x \preceq b_i \end{aligned}$$

2.  $\nu^{(k)} = \nu^{(k-1)} + t_k \left( B_1 x_1^{(k)} + B_2 x_2^{(k)} - d \right)$  for some step size  $t_k$

- subproblems in step 1 can be solved in parallel
- main difficulty is slow convergence of the subgradient method

## dual decomposition via augmented Lagrangian method

given initial  $\nu^{(0)}$ ,  $\mu > 0$

for  $k = 1, 2, \dots$

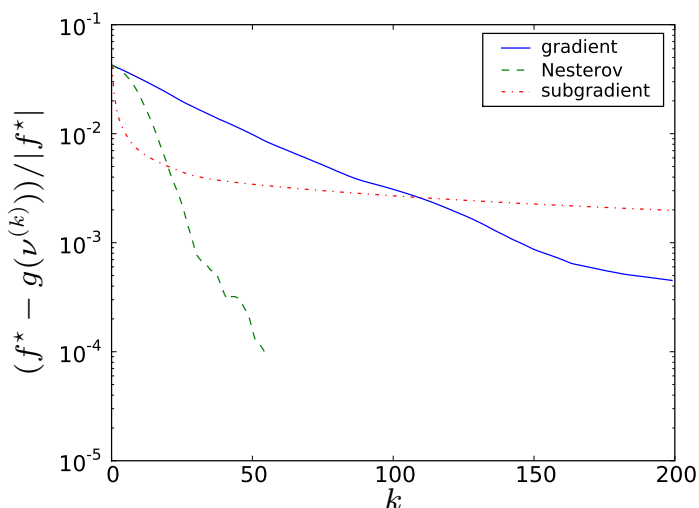
1. let  $x_1^{(k)}, x_2^{(k)}$  be the solution of

$$\begin{aligned} & \text{minimize} && \sum_{i=1,2} (B_i^T \nu^{(k-1)} + c_i)^T x_i + \frac{\mu}{2} \|B_1 x_1 + B_2 x_2 - d\|_2^2 \\ & \text{subject to} && A_1 x_1 \preceq b_1, \quad A_2 x_2 \preceq b_2 \end{aligned}$$

$$2. \nu^{(k)} = \nu^{(k-1)} + \mu \left( B_1 x_1^{(k)} + B_2 x_2^{(k)} - d \right)$$

- step 1 can be solved by alternating optimization over  $x_1$  and  $x_2$
- step 2 is gradient step for proximal approximation of dual function
- small  $\mu$  means fewer iterations in step 1, but smaller steps in step 2
- step 2 can be replaced by update in Nesterov's gradient method

**example** ( $m_1 = m_2 = 500$ ,  $n_1 = n_2 = 100$ ,  $p = 10$ )



- 'subgradient': subgradient method with step size  $1/k$
- 'gradient': algorithm of p. 14-27
- 'Nesterov': same method with Nesterov's gradient update (lecture 1)
- $\mu$  chosen so that on average two alternating minimizations suffice in 1

## Convex problem with inequality constraints

$$\begin{aligned} & \text{minimize} && f(x) \\ & \text{subject to} && Ax \preceq b \end{aligned}$$

- ideas extends to problems with nonlinear inequalities
- dual function is  $g(\lambda) = -b^T \lambda - f^*(-A^T \lambda)$

**proximal approximation of dual** (with  $\mu > 0$ )

$$g_\mu(\lambda) = \sup_{z \succeq 0} \left( g(z) - \frac{1}{2\mu} \|z - \lambda\|_2^2 \right)$$

- maximizing  $g_\mu$  is equivalent to maximizing  $g$
- $g_\mu$  is concave and differentiable
- $\nabla g_\mu$  is Lipschitz continuous with constant  $1/\mu$

## Proximal mapping

$$\text{prox}(\lambda) = \underset{z \succeq 0}{\text{argmax}} \left( g(z) - \frac{1}{2\mu} \|z - \lambda\|_2^2 \right)$$

**primal expression:**  $\text{prox}(\lambda) = \lambda + \mu(A\hat{x} + \hat{s} - b)$  where  $\hat{x}, \hat{s}$  solve

$$\begin{aligned} & \text{minimize}_{x,s} && f(x) + \lambda^T (Ax + s - b) + \frac{\mu}{2} \|Ax + s - b\|_2^2 \\ & \text{subject to} && s \succeq 0 \end{aligned}$$

this follows from duality between the problems

$$\begin{aligned} & \text{minimize}_{x,s,y} && f(x) + \lambda^T y + \frac{\mu}{2} \|y\|_2^2 && \text{maximize}_z && g(z) - \frac{1}{2\mu} \|z - \nu\|_2^2 \\ & \text{subject to} && Ax + s - b = y && && z \succeq 0 \\ & && s \succeq 0 && && \end{aligned}$$

at optimality  $z = \mu y + \lambda = \mu(Ax + s - b) + \lambda$

## Gradient of proximal approximation

$$\begin{aligned}g_{\mu}(\lambda) &= \sup_{z \succeq 0} \left( g(z) - \frac{1}{2\mu} \|\lambda - z\|_2^2 \right) \\ &= \inf_{x, s \succeq 0} \left( f(x) + \lambda^T (Ax + s - b) + \frac{\mu}{2} \|Ax + s - b\|_2^2 \right)\end{aligned}$$

from dual expression

$$\nabla g_{\mu}(\lambda) = \frac{1}{\mu}(\hat{z} - \lambda) = \frac{1}{\mu}(\text{prox}(\lambda) - \lambda)$$

where  $\hat{z} = \text{prox}(\lambda)$  is the maximizer in the definition of  $g_{\mu}$

from primal expression

$$\nabla g_{\mu}(\lambda) = A\hat{x} + \hat{s} - b$$

where  $\hat{x}, \hat{s}$  are the minimizer in the definition of  $g_{\mu}$

## Augmented Lagrangian method

**given** initial  $\lambda^{(0)} \succ 0, \mu > 0$

**for**  $k = 1, 2, \dots$

1. let  $x^{(k)}, s^{(k)}$  be the solution of

$$\begin{aligned}\text{minimize} & \quad f(x) + (\lambda^{(k-1)})^T (Ax + s - b) + (\mu/2) \|Ax + s - b\|_2^2 \\ \text{subject to} & \quad s \succeq 0\end{aligned}$$

2. take  $\lambda^{(k)} = \lambda^{(k-1)} + \mu(Ax^{(k)} + s^{(k)} - b)$

this is the gradient method with step size  $\mu$  applied to

$$\text{maximize} \quad g_{\mu}(\lambda) = \sup_{z \succeq 0} \left( g(z) - \frac{1}{2\mu} \|z - \lambda\|_2^2 \right)$$

## Elimination of slack variables in augmented Lagrangian

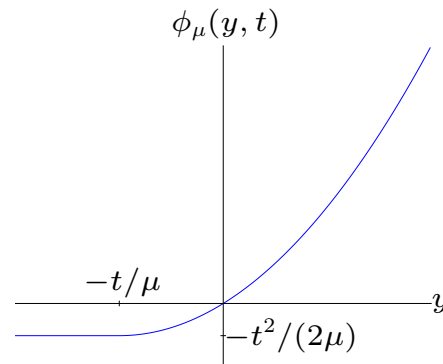
$$\begin{aligned} & \text{minimize} && f(x) + \lambda^T(Ax + s - b) + \frac{\mu}{2}\|Ax + s - b\|_2^2 \\ & \text{subject to} && s \succeq 0 \end{aligned}$$

optimal  $s$  is  $s_i = \max\{0, -a_i^T x - b_i - \lambda_i/\mu\}$ ; problem reduces to

$$\text{minimize} \quad f(x) + \sum_{i=1}^m \phi_\mu(a_i^T x + b_i, \lambda_i)$$

$$\phi_\mu(y, t)$$

$$= \begin{cases} -t^2/(2\mu) & y + t/\mu \leq 0 \\ ty + \mu y^2/2 & y + t/\mu \geq 0 \end{cases}$$



## Practical aspects

### choice of $\mu$

- large  $\mu$  improves convergence of gradient method
- large  $\mu$  can make primal subproblems more difficult
- practical implementations often use an increasing sequence of values

### inexact methods

- compute an approximate solution of the subproblems
- extensive convergence theory

## References

- S. Boyd, course notes for EE364b, Convex Optimization II  
the network flow example is taken from EE364b lecture 9 and the notes on decomposition
- D.P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods* (1982)
- D.P. Bertsekas, *Network Optimization. Continuous and Discrete Models* (1998)
- W. Yin, S. Osher, D. Goldfarb, J. Darbon, *Bregman iterative algorithms for  $\ell_1$ -minimization with applications to compressed sensing*, *SIAM J. Imaging Sciences* (2008)