

I 5-780: Grad AI

Lec. 9: Linear programs, Duality

Geoff Gordon (this lecture)

Tuomas Sandholm

TAs Erik Zawadzki, Abe Othman

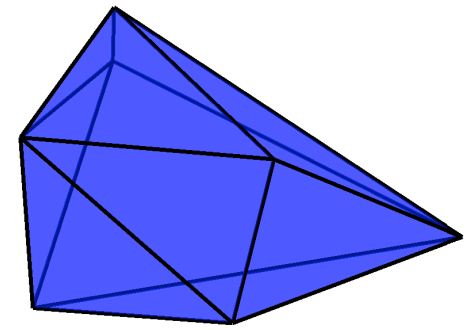
Admin

- Have you tested your handin directories?
 - ▶ `/afs/cs/user/aothman/dropbox/USERID/`
 - ▶ where USERID is your Andrew ID
- Poster session:
 - ▶ ???

Review

- LPs, ILPs, MILPs
 - ▶ \mathbb{R} or \mathbb{Z} variables
 - ▶ linear $\leq \geq =$
 - ▶ linear objective
 - ▶ LP relaxations, integrality gap
 - ▶ relation to SAT, MAXSAT, PBI
 - ▶ complexity (LP: P; ILP: NP & no approx)
 - ▶ (in)feasible, (sub)optimal, (in)active

Review



- Standard form: all vars ≥ 0 , all = constraints
- Nonsingular: n vars $\geq m$ constraints, rank m
- Basis
 - ▶ spans $\text{Rng}(A)$ ($m \times m$ invertible submatrix)
 - ▶ corresponds to “corner”
 - ▶ using row ops to make basic variables into “slacks” \rightarrow **tableau** notation
- Degeneracy: distinct bases yield same corner
- Naïve algorithm: check all bases

Finding corners

x	y	u	v	w	RHS
1	1	1	0	0	4
2	5	0	1	0	12
1	2	0	0	1	5

set $x, y = 0$

$u = 4 \quad v = 12 \quad w = 5$

1	1	1	0	0	4
2	5	0	1	0	12
1	2	0	0	1	5

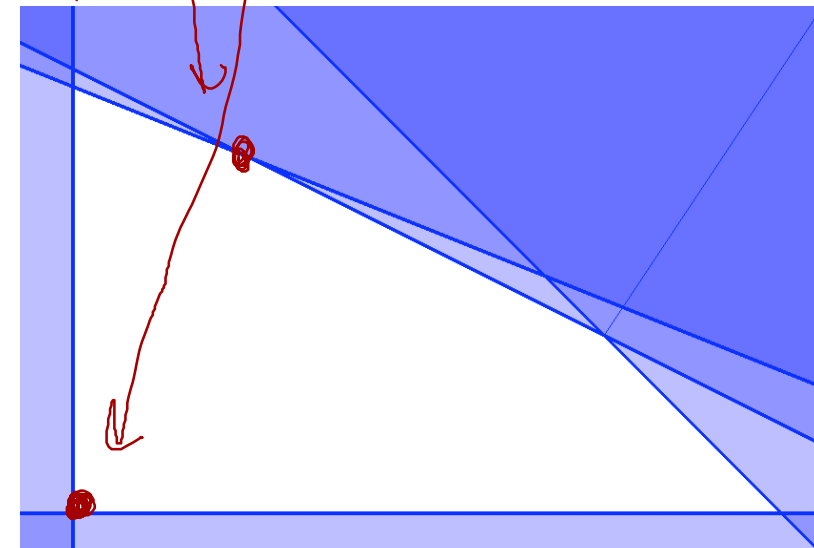
set $v, w = 0$

$x = 1 \quad y = 2 \quad u = 1$

1	1	1	0	0	4
2	5	0	1	0	12
1	2	0	0	1	5

set $x, u = 0$

$y = 4$
 $v = -8$
 $w = -3$



Simplex in one slide

(ignoring degeneracy, which is actually important)

- Given a nonsingular standard-form LP
 - ▶ make it nonsingular if needed
- Start from a feasible basis and its tableau
 - ▶ big-M if needed
- Pick non-basic variable w/ objective > 0 (max)
- Pivot it into basis, getting neighboring basis
 - ▶ select exiting variable to keep feasibility
- Repeat until all non-basic variables have objective < 0 (max)

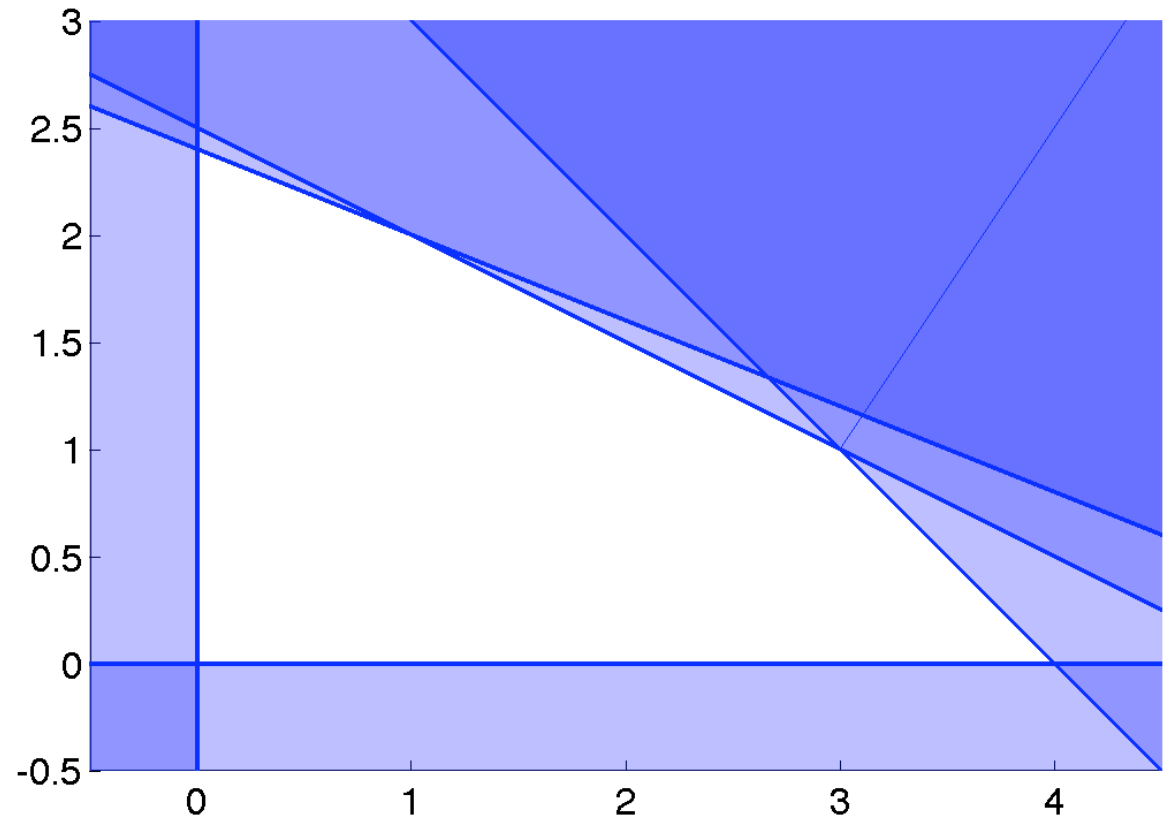
Example

max $2x + 3y$ s.t.

$$x + y \leq 4$$

$$2x + 5y \leq 12$$

$$x + 2y \leq 5$$



<u>x</u>	<u>y</u>	<u>s</u>	<u>t</u>	<u>u</u>	<u>RHS</u>
1	1	1	0	0	4
2	5	0	1	0	12
1	2	0	0	1	5
2	3	0	0	0	↑

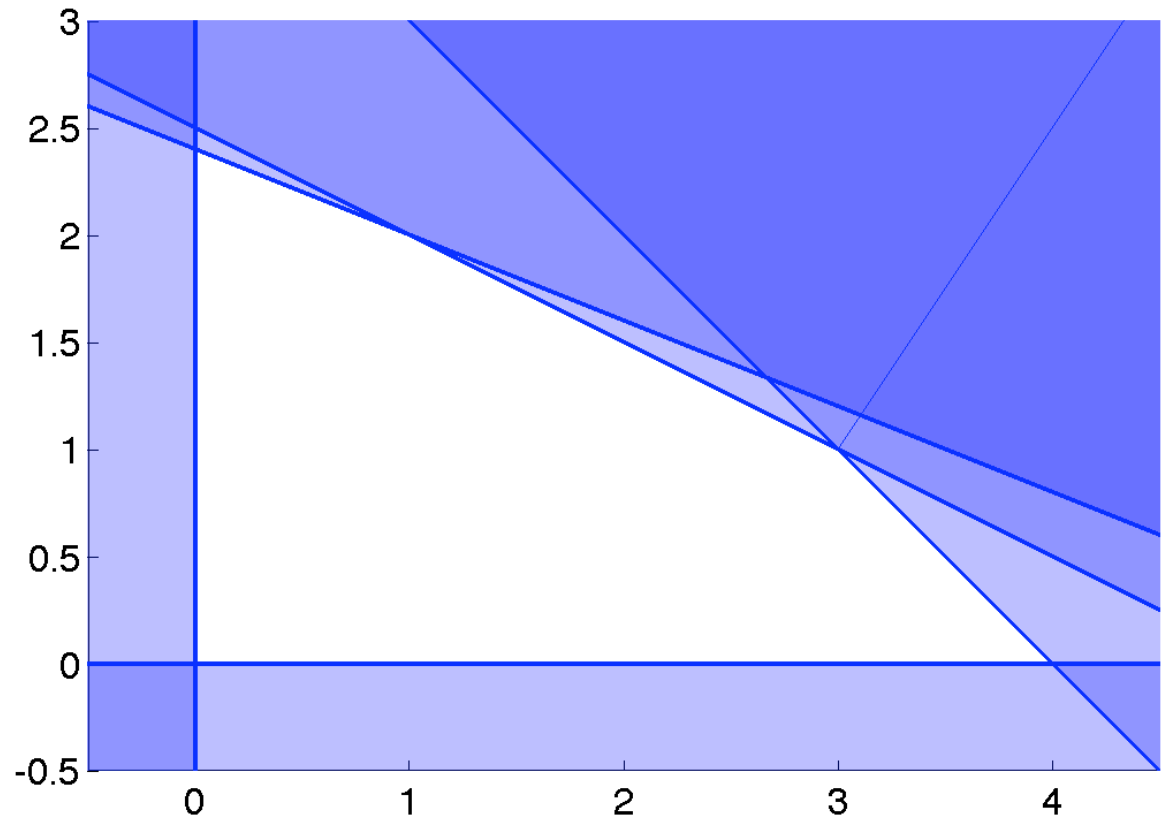
Example

$$\max 2x + 3y \text{ s.t.}$$

$$x + y \leq 4$$

$$2x + 5y \leq 12$$

$$x + 2y \leq 5$$



<u>x</u>	<u>y</u>	<u>s</u>	<u>t</u>	<u>u</u>	<u>RHS</u>
0.4	1	0	0.2	0	2.4
0.6	0	1	-0.2	0	1.6
0.2	0	0	-0.4	1	0.2
0.8	0	0	-0.6	0	↑

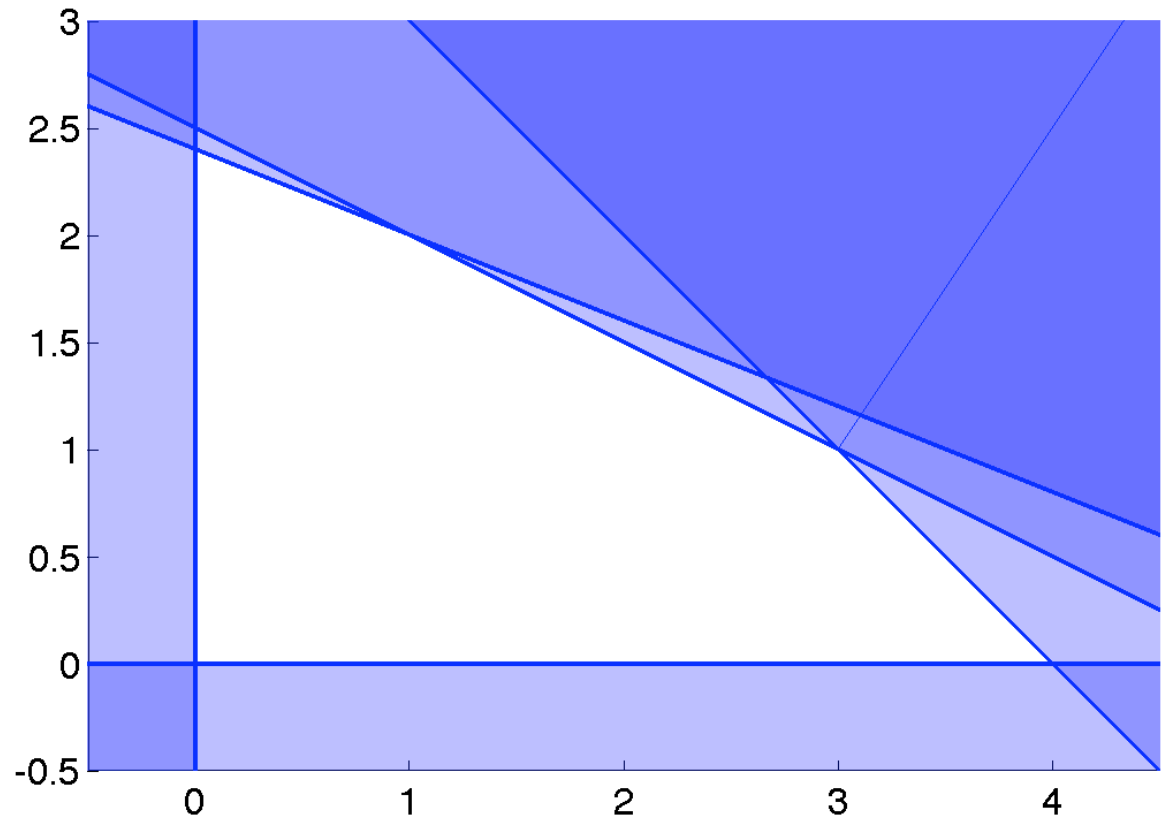
Example

$$\max 2x + 3y \text{ s.t.}$$

$$x + y \leq 4$$

$$2x + 5y \leq 12$$

$$x + 2y \leq 5$$



<u>x</u>	<u>y</u>	<u>s</u>	<u>t</u>	<u>u</u>	<u>RHS</u>
1	0	0	-2	5	1
0	1	0	1	-2	2
0	0	1	1	-3	1
0	0	0	1	-4	↑

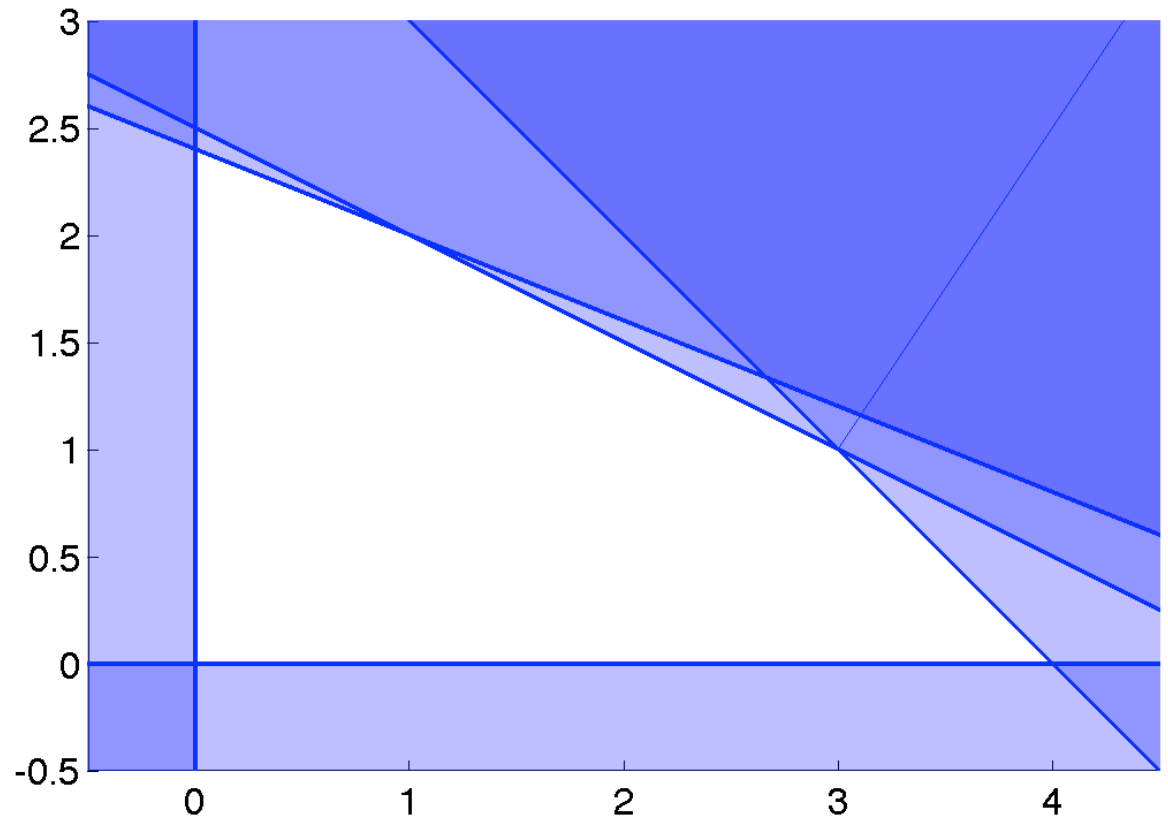
Example

$$\max 2x + 3y \text{ s.t.}$$

$$x + y \leq 4$$

$$2x + 5y \leq 12$$

$$x + 2y \leq 5$$

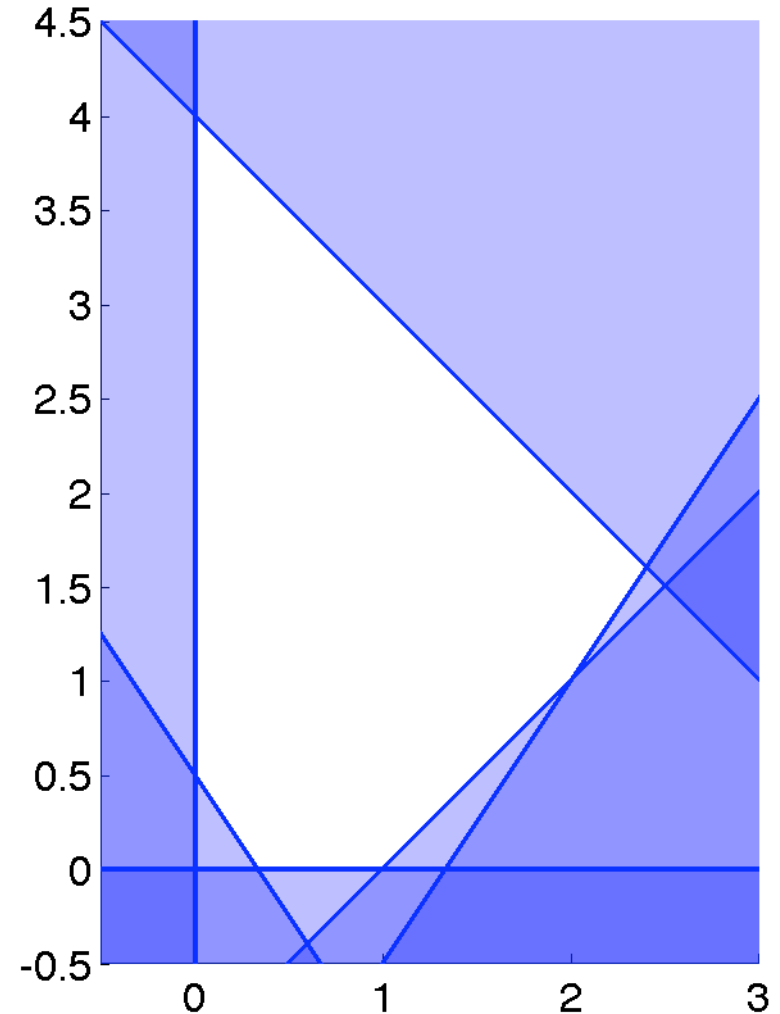


<u>x</u>	<u>y</u>	<u>s</u>	<u>t</u>	<u>u</u>	<u>RHS</u>
1	0	2	0	-1	3
0	1	-1	0	1	1
0	0	1	1	-3	1
0	0	-1	0	-1	↑

Big M

<u>x</u>	<u>y</u>	<u>t</u>	<u>u</u>	<u>v</u>	<u>w</u>	<u>RHS</u>
1	1	1	0	0	0	4
3	-2	0	1	0	0	4
1	-1	0	0	1	0	1
-3	-2	0	0	0	1	-1
1	-2	0	0	0	0	↑

- So far, assumed we started w/ initial feasible basis
- How do we get one?
 - ▶ for each violated constraint, add var w/ coeff -1
 - ▶ penalize in objective, include in initial basis



Ex: combinatorial auctions

- Goods: Newspaper, Magazine, L shoe, R shoe
- Bids (note use of bidding language: 7 rt 16 numbers for B_1 and 1 rt 16 for B_2):

▶ N: +5; M: +4

▶ N, M: -3

▶ L, R: +10

▶ N, L, R: -5;

M, L, R: -4;

N, M, L, R: +3

▶ M: +10

Bidder 1

Bidder 2

Winner determination

- Goods: Newspaper, Magazine

- Bids:

- ▶ N: +5; M: +4

- ▶ N, M: -3

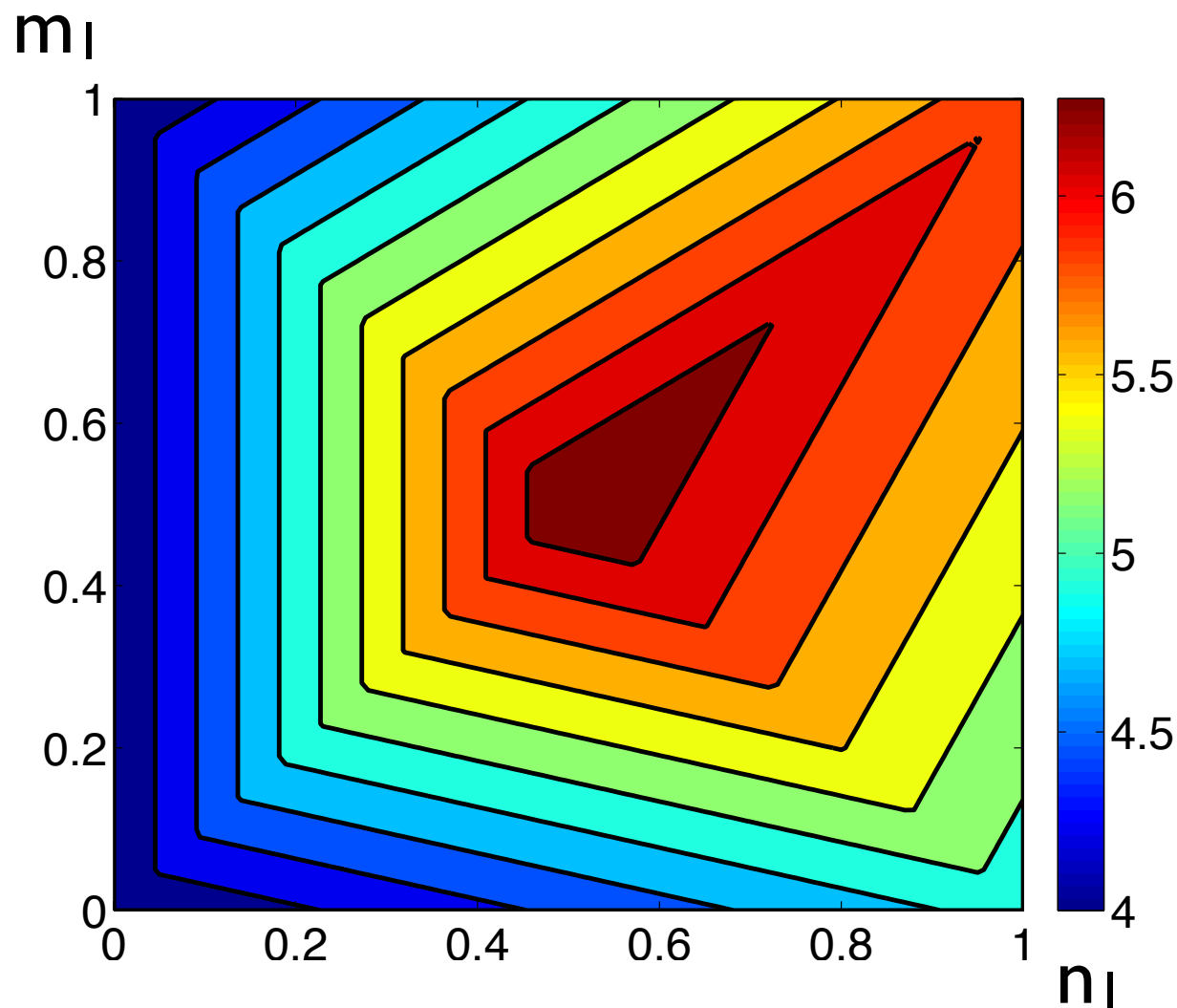
Bidder 1

- ▶ N, M: +4

Bidder 2

Bounds

- Any feasible point yields lower bd: (N to B_I , keep M) $\rightarrow 5$
- Upper bound: solve LP relaxation
 - ▶ a bit expensive
 - ▶ can we be lazier?



Being lazy

- A “hard” LP:

$$\max x + y \text{ s.t.}$$

$$x + y \leq 3$$

$$x \leq 1$$

$$y \leq 1$$

OK, we got lucky

- What if it were:

$$\max x + 3y \text{ s.t.}$$

$$x + y \leq 3$$

$$x \leq 1$$

$$y \leq 1$$

How general is this?

- What if it were:

$$\max px + qy \text{ s.t.}$$

$$x + y \leq 3$$

$$x \leq 1$$

$$y \leq 1$$

Let's do it again

- Note \geq , \leq , $=$ constraints, min obj

$$\min x - 2y \text{ s.t.}$$

$$x + y \geq 2$$

$$y \leq 3$$

$$2x - y = 0$$

Summary of LP duality

- Use multipliers to write combined constraints

\geq

\leq

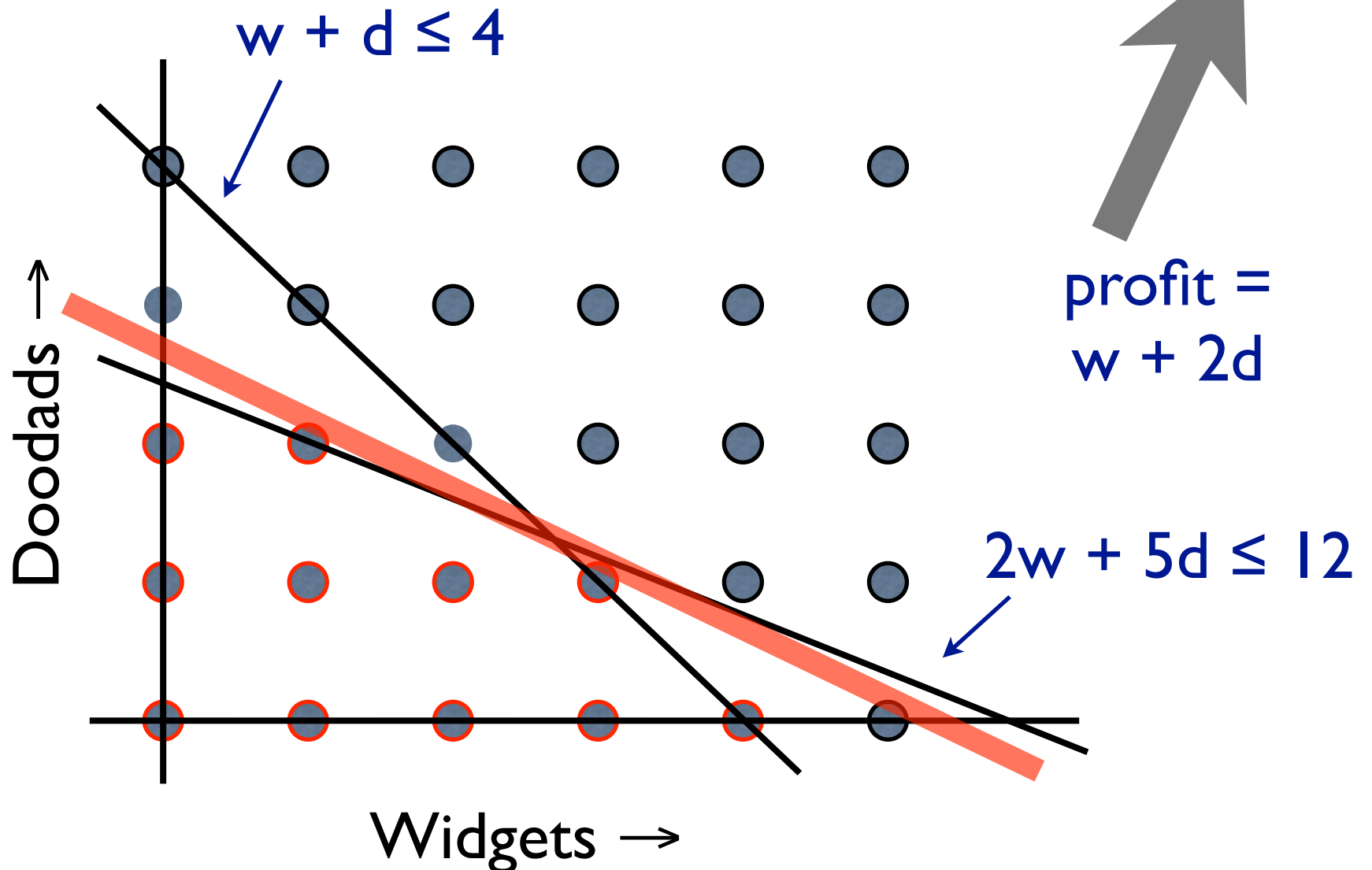
$=$

- Constrain multipliers to give us a bound on objective (by matching coefficients)
- Optimize to get tightest bound
- Q: what happens if we take dual of dual?

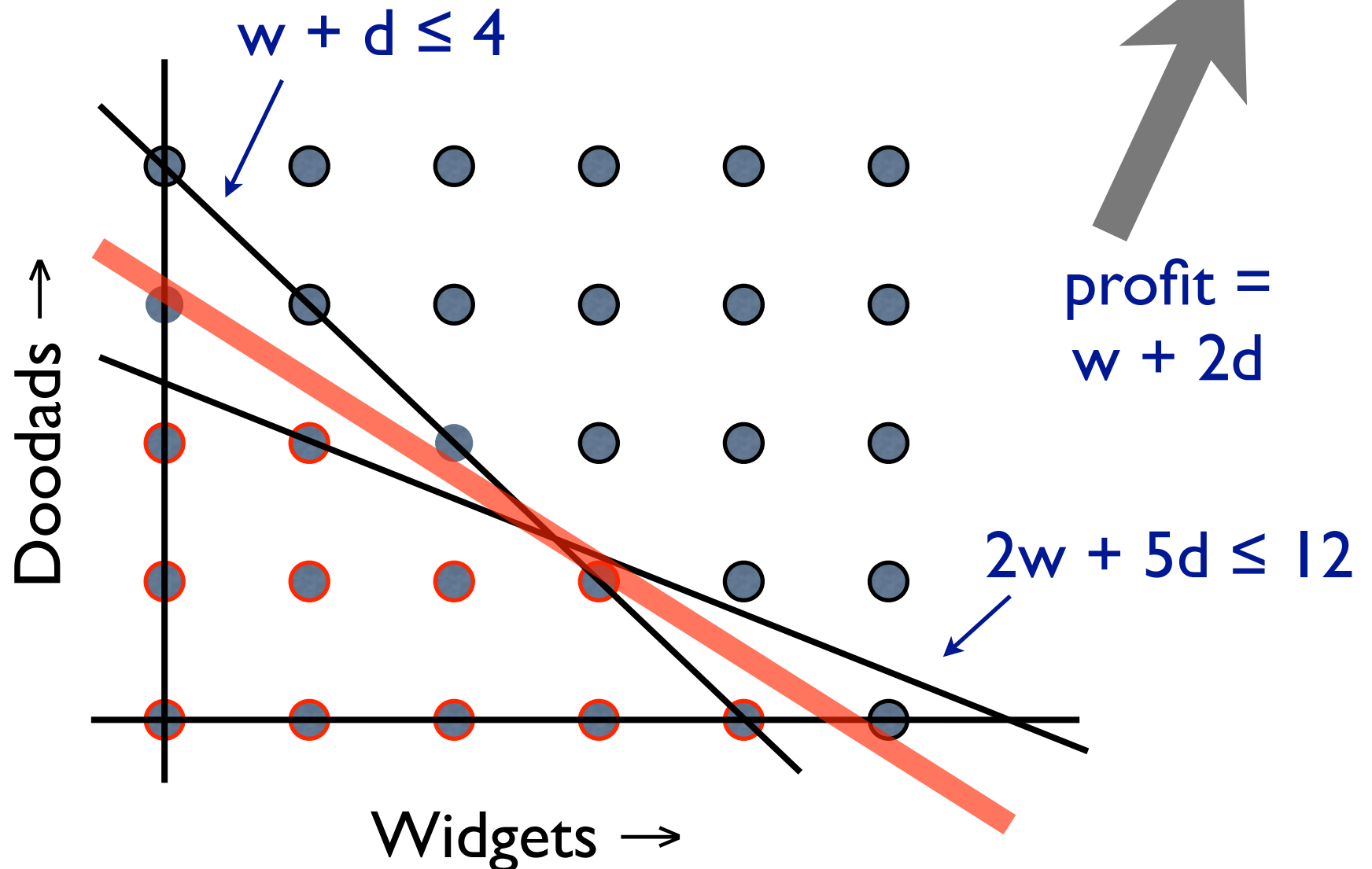
Ordering

- For primal max problem (dual min):
 - ▶ primal feas primal opt dual opt dual feas
- For primal min problem (dual max):
 - ▶ primal feas primal opt dual opt dual feas

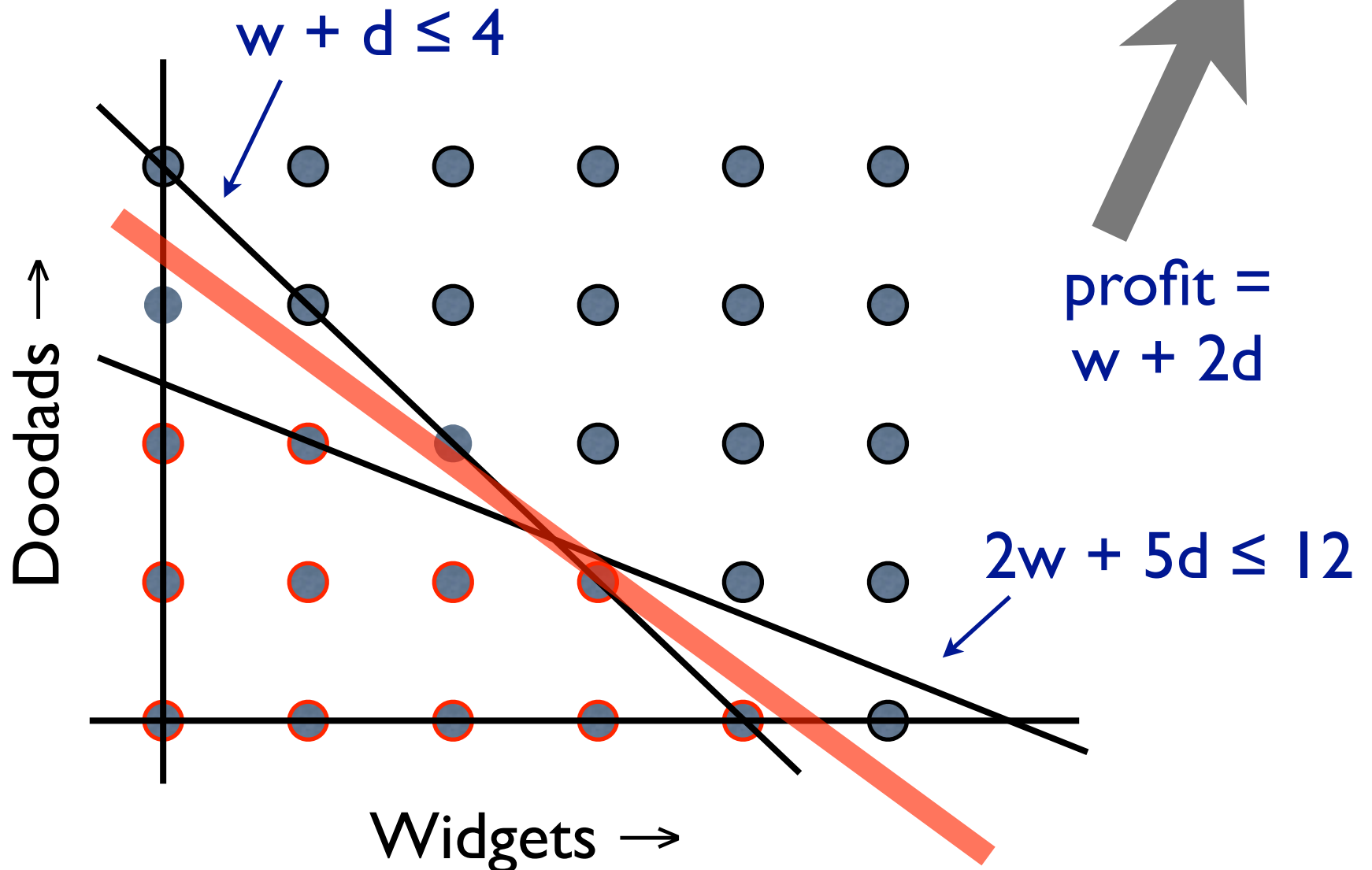
Geometrically



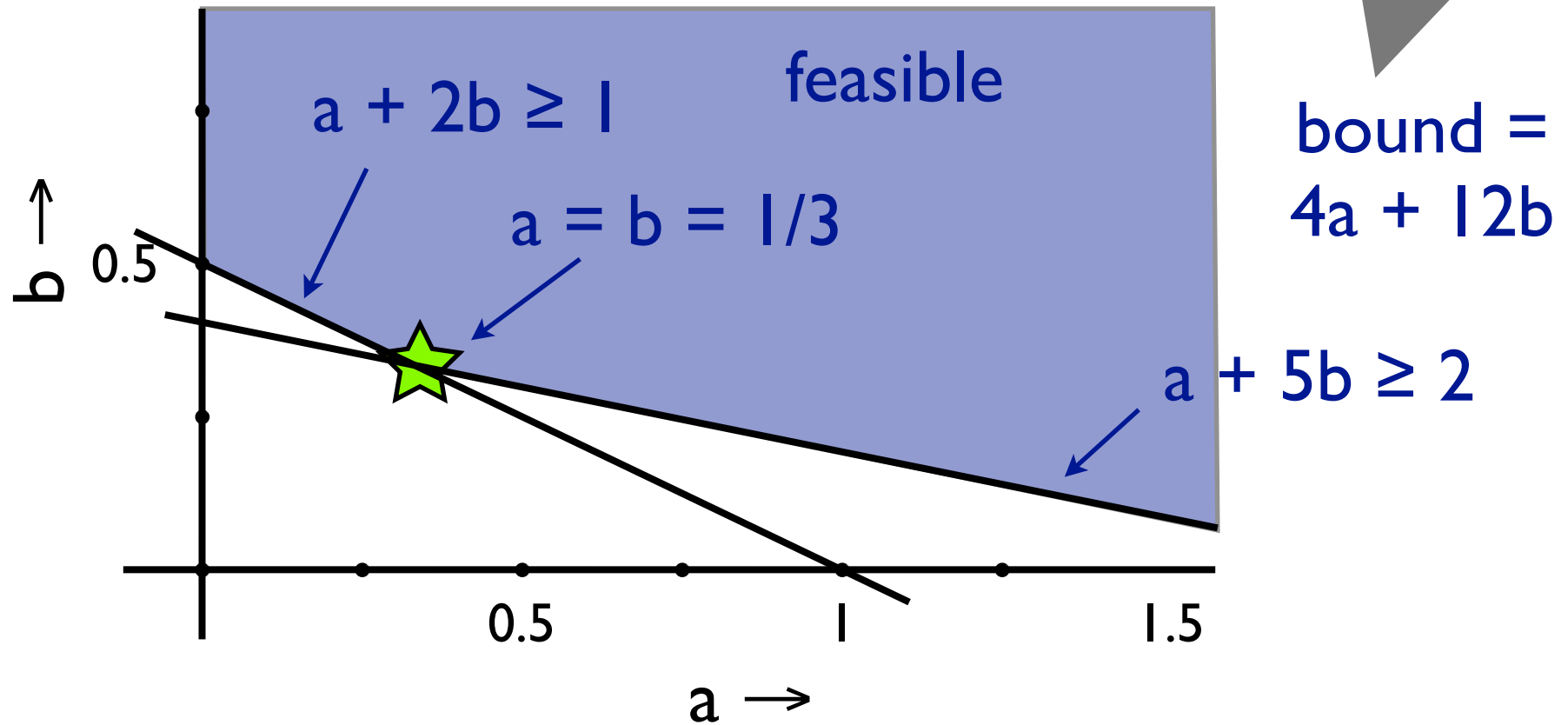
Geometrically



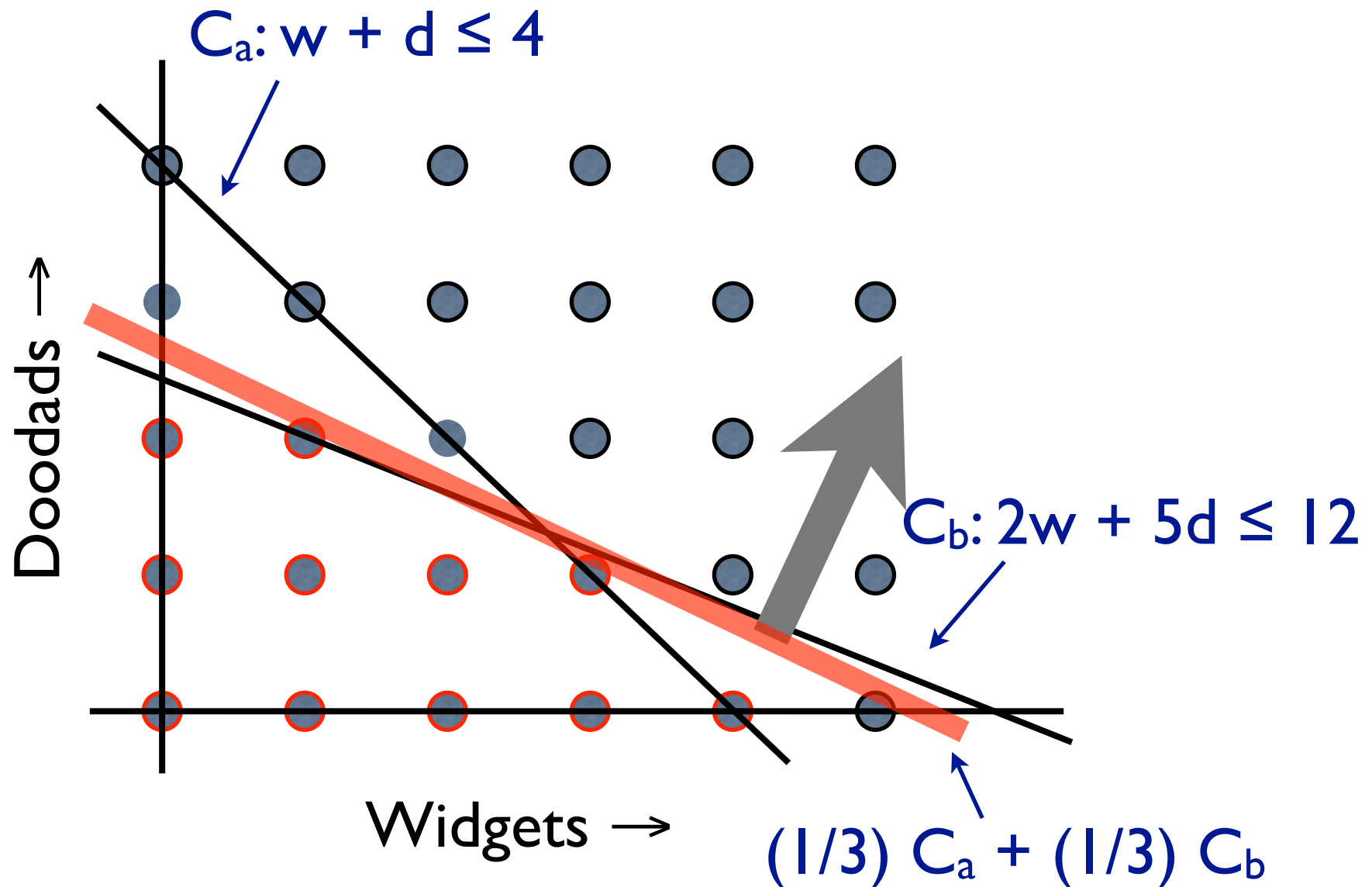
Geometrically



Dual widgets



Dual variables as multipliers



So why bother?

- Reason 1: any feasible solution to dual yields upper bound (compared with only optimal solution to primal)
- Reason 2: dual might be easier to work with
- Reason 3: solvers can often work w/ primal and dual at the same time for no extra cost

Interpreting the dual variables

- Primal variables in the factory LP were how many widgets and doodads to produce
- Interpreted dual variables as multipliers for primal constraints—not much intuition
- Often possible to interpret dual variables as **prices** for primal constraints

Dual variables as prices

- Suppose someone offered us a quantity ε of wood, loosening constraint to

$$w + d \leq 4 + \varepsilon$$

- How much should we be willing to pay for this wood?

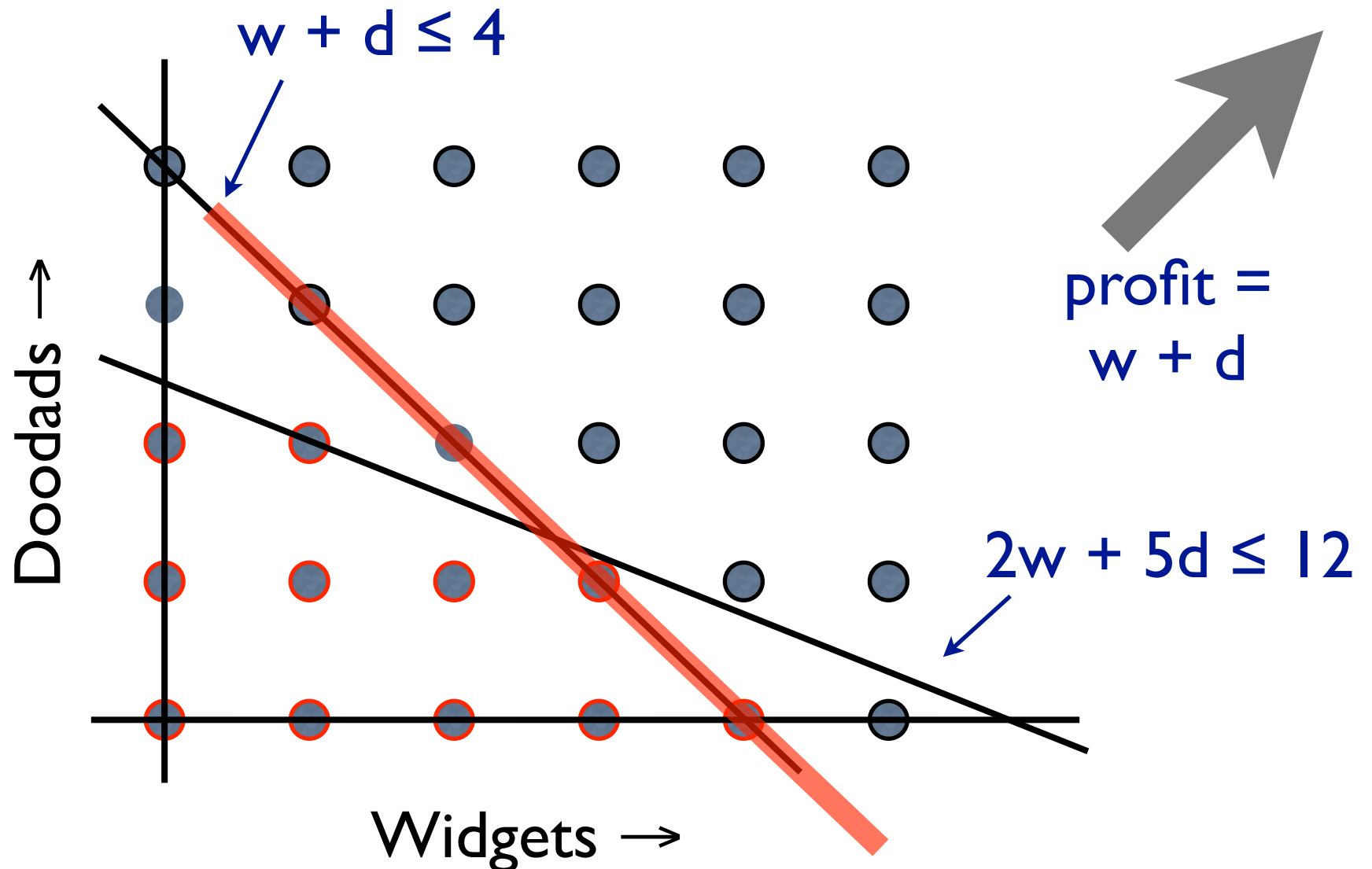
Dual variables as prices

- Dual constricts stay same: $a + 2b \geq 1$, $a + 5b \geq 2$
- Dual objective becomes: $\min (4+\varepsilon)a + 12b$
- Previous solution $a = b = 1/3$ still feasible
 - ▶ still optimal if ε small enough
- Bound changes to $(4+\varepsilon)a + 12b$, increase by $\varepsilon/3$
- So we should pay up to \$1/3 per unit of wood (in small quantities)

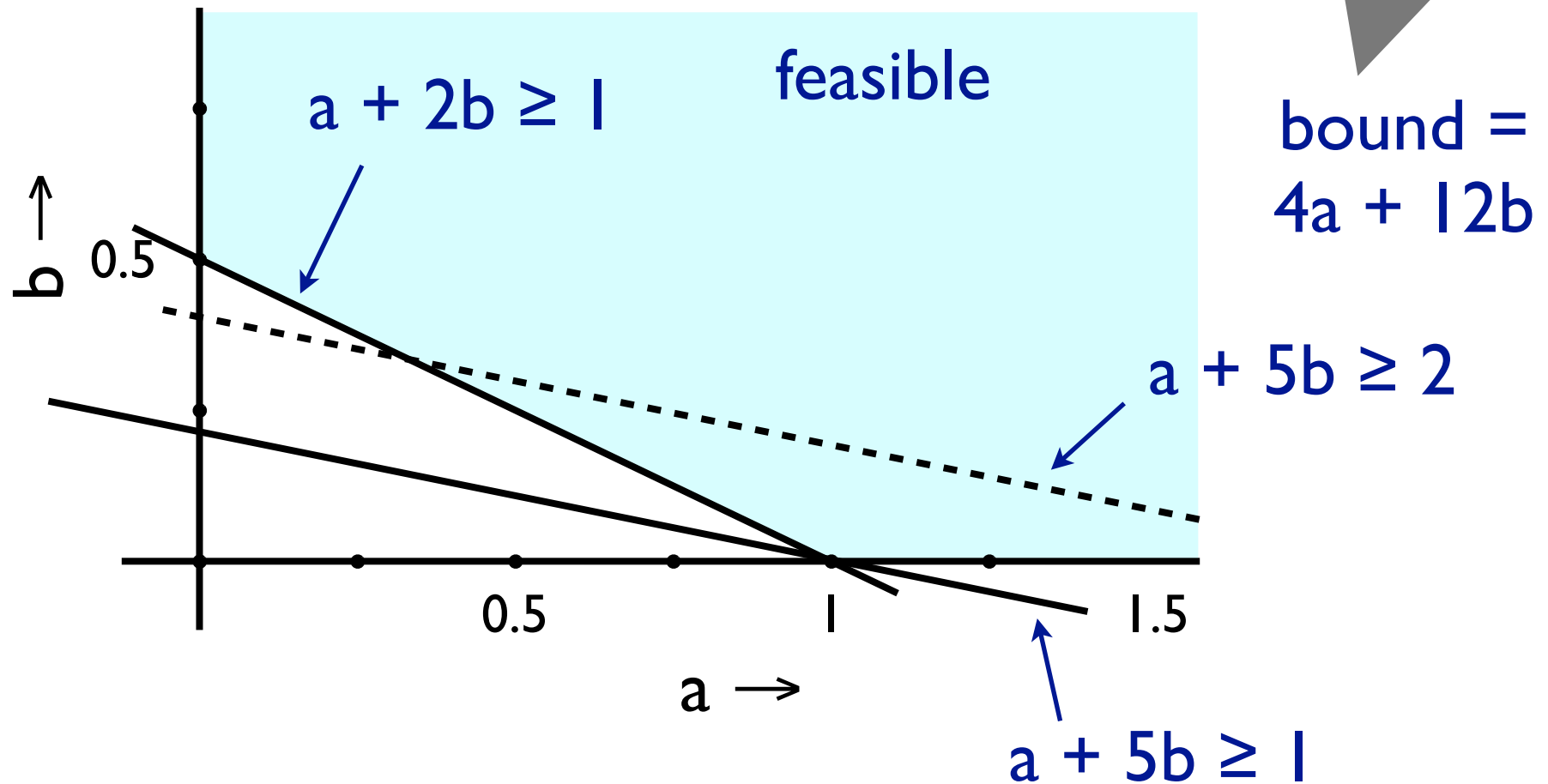
Dual degeneracy

- Primal degenerate = two bases, same corner
- Dual can be degenerate too
 - ▶ so, 4 possibilities for degeneracy
- E.g., what if objective were $w+d$ (not $w+2d$)?

Dual degeneracy



Dual degeneracy



Complementary slackness

- Suppose a constraint is inactive. Would we pay anything to have it relaxed?
- Write $s_j \geq 0$ for slack in primal constraint j
- Write $d_j \geq 0$ for dual variable (multiplier, price) for constraint j
- CS: at optimal primal and dual solutions,
- Uses: certificate of optimality, proving that optimal solution satisfies some property