15-780: Grad AI
Lecture 12: Optimization, Duality

Geoff Gordon (this lecture)
Tuomas Sandholm
TAs Sam Ganzfried, Byron Boots

Review

LPs, ILPs, and their ilk

- *LPs, ILPs, MILPs, 0-1 ILPs*
- *Relaxations, integrality gap*
- *Complexity (LP=poly, ILP=NP)*

Pseudo-boolean inequalities

- *0-1 ILPs w/o objective*
- *Useful generalization of SAT*
- *Parallels*
  - *LP relaxation vs. unit resolution*
  - *LP relaxation + Gomory vs. resolution*
  - *DPLL+CL vs. branch & cut*

Resolution / Gomory example

- $(x \vee \neg y \vee \neg z) \wedge (z \vee \neg y \vee a)$

$$(x \vee \bar{y} \vee \bar{z}) \wedge (z \vee a \vee \bar{y}) \longrightarrow x \vee \bar{y} \vee a$$

$$x + \bar{y} + \bar{z} \geq 1 \quad z + a + \bar{y} \geq 1 \longrightarrow x + 2\bar{y} + a + 1 \geq 2$$

---

vars $x, \bar{x}, y, \bar{y}, z, \bar{z}, a, \bar{a}, s \geq 0$

$$x + \bar{x} = 1 \quad y + \bar{y} = 1 \quad \ldots \quad a + \bar{a} = 1$$

$$x + 2\bar{y} + a = 1 + s$$

---

basis: $\bar{x}, \bar{a}, y, \bar{y}, z$

$$\bar{x} = 1 - x, \quad \bar{a} = 1 - a, \quad z = 1 - \bar{z}$$

$$\bar{y} = [1 + s - a - x]/2 = 1/2 + s/2 - a/2 - x/2$$

$$y = 1 - [1/2 + s/2 - a/2 - x/2]$$

$$\bar{y} = 1/2 + 5/2 - a/2 - x/2$$

$\Downarrow$ positivity

$$\bar{y} \geq 1/2 - a - x$$

$\Downarrow$ rearrange

$$\bar{y} + a + x \geq 1/2$$

$\Downarrow$ integrality

$$\bar{y} + a + x \geq 1$$

Branch & bound (& cut)

[schema, value] = bb(F, sch, bnd)

  [v_{rx}, sch_{rx}] = relax(F, sch)

  if integer(sch_{rx}): return [sch_{rx}, v_{rx}]

  if $v_{rx} \geq$ bnd: return [sch, v_{rx}]

  Pick variable x_i

  [sch^0, v^0] = bb(F, sch/(x_i: 0), bnd)

  [sch^1, v^1] = bb(F, sch/(x_i: 1), min(bnd, v^0))

  if ($v^0 \leq v^1$): return [sch^0, v^0]

  else: return [sch^1, v^1]

Branch & bound (& cut)

[schema, value] = bb(F, sch, bnd)

$[v_{rx}, sch_{rx}]$ = relax(F, sch) ← *for branch & cut: add cuts as desired here, re-solve relaxation*

if integer(sch_{rx}): return $[sch_{rx}, v_{rx}]$

if $v_{rx} \geq$ bnd: return $[sch, v_{rx}]$

Pick variable x_i

$[sch^0, v^0]$ = bb(F, sch/(x_i: 0), bnd)

$[sch^1, v^1]$ = bb(F, sch/(x_i: 1), min(bnd, v^0))

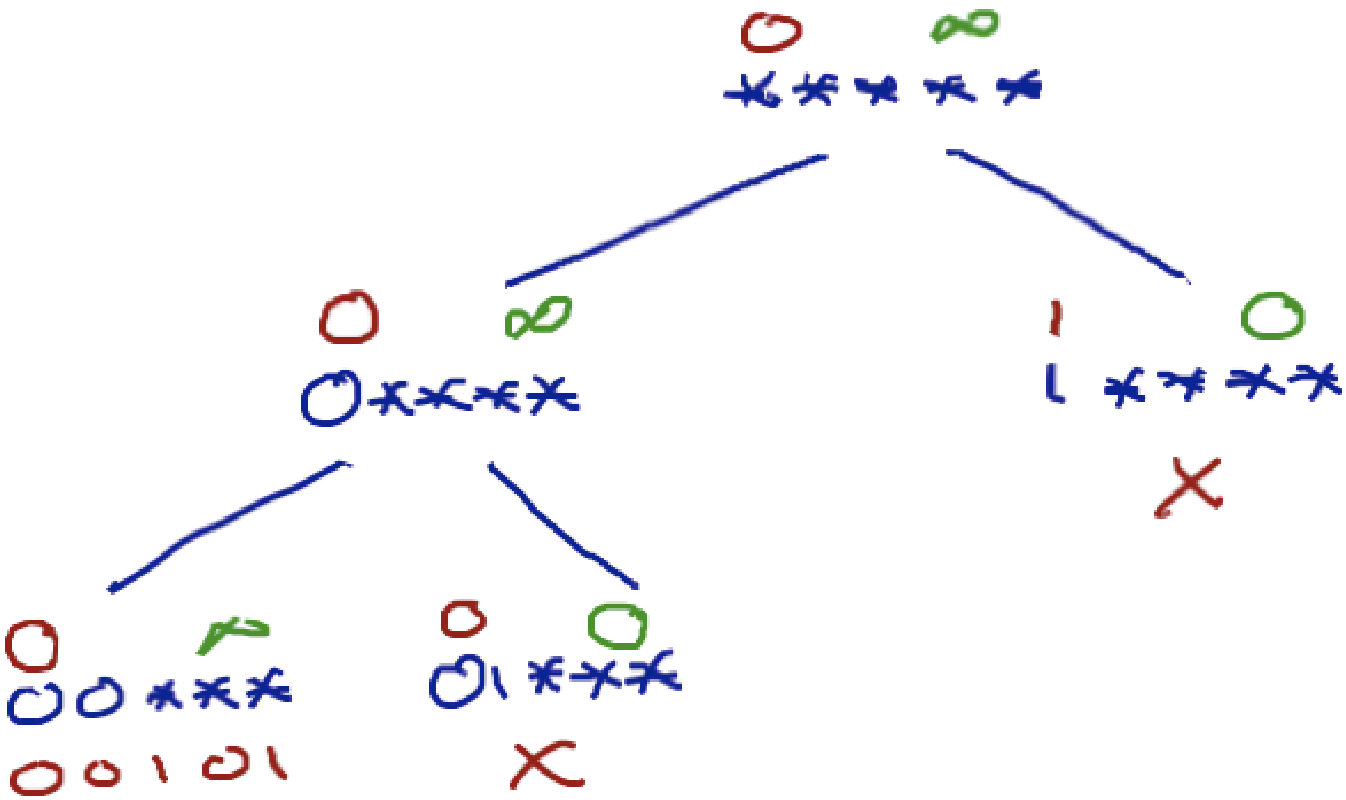
if ($v^0 \leq v^1$): return $[sch^0, v^0]$

else: return $[sch^1, v^1]$

A random 3CNF

$$(x_2 \vee x_5 \vee x_4) \wedge (\overline{x}_2 \vee \overline{x}_3 \vee \overline{x}_5) \wedge (\overline{x}_2 \vee \overline{x}_2 \vee x_2) \wedge (\overline{x}_3 \vee x_5 \vee \overline{x}_3)$$
$$\wedge (\overline{x}_2 \vee \overline{x}_3 \vee x_4) \wedge (\overline{x}_2 \vee \overline{x}_2 \vee x_3) \wedge (\overline{x}_1 \vee \overline{x}_5 \vee x_5) \wedge (x_3 \vee \overline{x}_2 \vee x_5)$$
$$\wedge (\overline{x}_3 \vee x_3 \vee \overline{x}_4) \wedge (x_2 \vee x_1 \vee x_5) \wedge (\overline{x}_1 \vee x_2 \vee x_1) \wedge (x_1 \vee x_3 \vee x_4)$$
$$\wedge (\overline{x}_5 \vee \overline{x}_4 \vee x_1) \wedge (\overline{x}_3 \vee x_5 \vee x_4) \wedge (x_5 \vee \overline{x}_1 \vee \overline{x}_5) \wedge (\overline{x}_3 \vee x_5 \vee \overline{x}_3)$$
$$\wedge (x_1 \vee \overline{x}_1 \vee \overline{x}_3) \wedge (x_5 \vee \overline{x}_4 \vee x_4) \wedge (x_5 \vee \overline{x}_5 \vee x_3) \wedge (\overline{x}_1 \vee \overline{x}_1 \vee x_5)$$
$$\wedge (\overline{x}_1 \vee \overline{x}_3 \vee x_4)$$

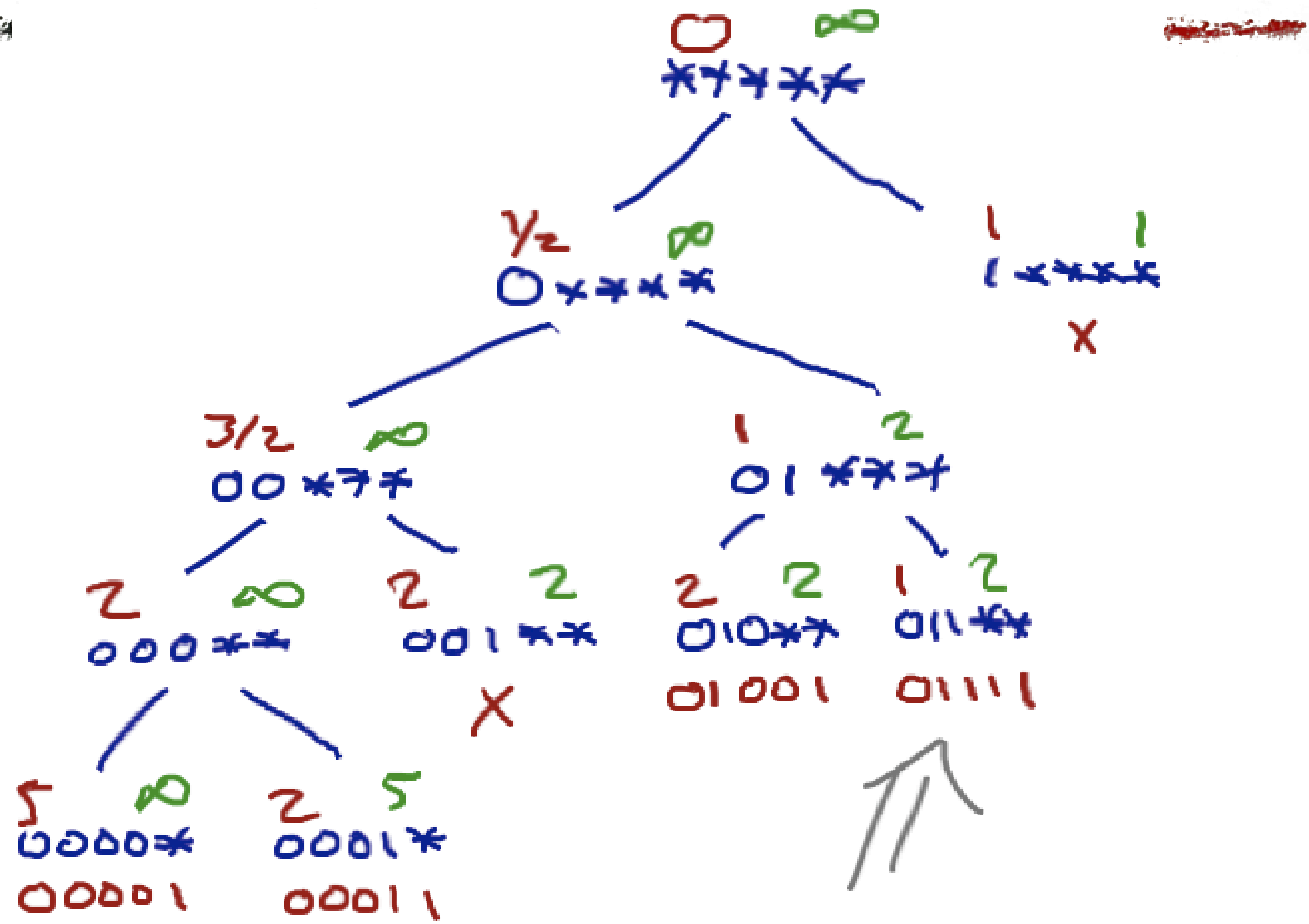
Branch & bound tree



A random 3CNF

$$(x_2 \vee \overline{x}_3 \vee x_1) \wedge (\overline{x}_3 \vee x_2 \vee x_1) \wedge (x_2 \vee x_2 \vee \overline{x}_1)$$
$$\wedge (x_1 \vee x_4 \vee x_3) \wedge (x_4 \vee x_4 \vee x_2) \wedge (\overline{x}_2 \vee \overline{x}_4 \vee x_3)$$
$$\wedge (\overline{x}_4 \vee \overline{x}_4 \vee x_5) \wedge (x_2 \vee x_3 \vee \overline{x}_5) \wedge (\overline{x}_2 \vee x_5 \vee x_3)$$
$$\wedge (x_3 \vee x_3 \vee x_3) \wedge (\overline{x}_2 \vee \overline{x}_1 \vee \overline{x}_3) \wedge (x_2 \vee x_4 \vee x_5)$$
$$\wedge (\overline{x}_1 \vee \overline{x}_4 \vee x_3) \wedge (\overline{x}_5 \vee x_2 \vee x_4) \wedge (\overline{x}_2 \vee \overline{x}_3 \vee x_1)$$
$$\wedge (\overline{x}_2 \vee \overline{x}_4 \vee \overline{x}_4) \wedge (x_4 \vee \overline{x}_3 \vee \overline{x}_2) \wedge (\overline{x}_2 \vee \overline{x}_5 \vee \overline{x}_5)$$
$$\wedge (\overline{x}_4 \vee x_5 \vee \overline{x}_2) \wedge (x_4 \vee x_2 \vee x_3) \wedge (\overline{x}_4 \vee x_5 \vee x_3)$$

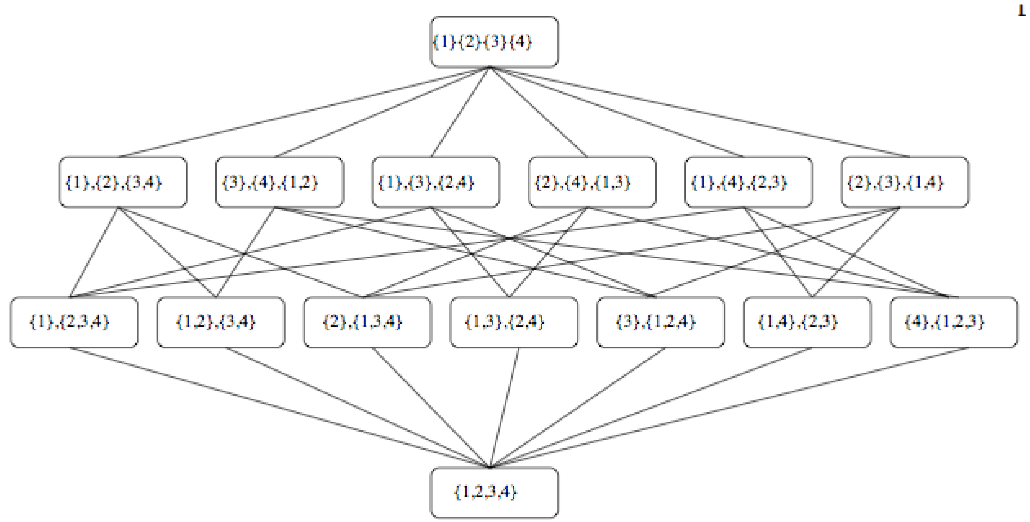
Branch & bound tree



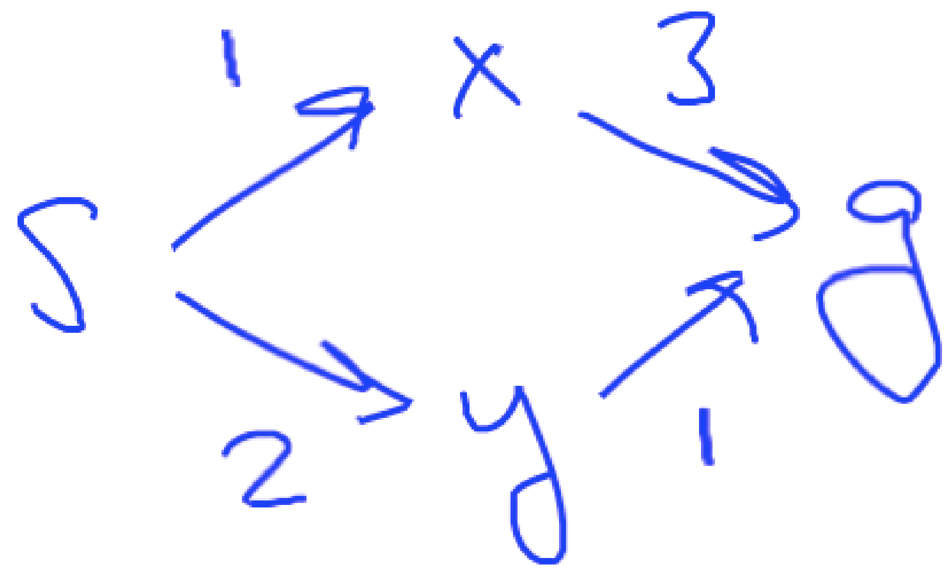
Examples

Examples

- *Any problem in NP, since "does MILP have solution of value ≥ z?" NP-complete*
- *E.g., allocation problems like clearing combinatorial auctions*



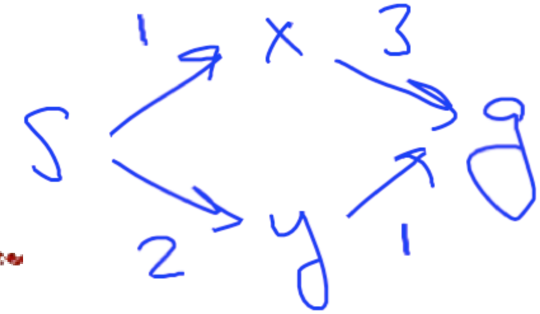
Path planning



- *Find the min-cost path: 0-1 variables*

$P_{sx}, P_{sy}, P_{xg}, P_{yg} \geq 0$

Path planning



min

$$p_{sx} + 3p_{xg} + 2p_{sy} + p_{yg}$$

st

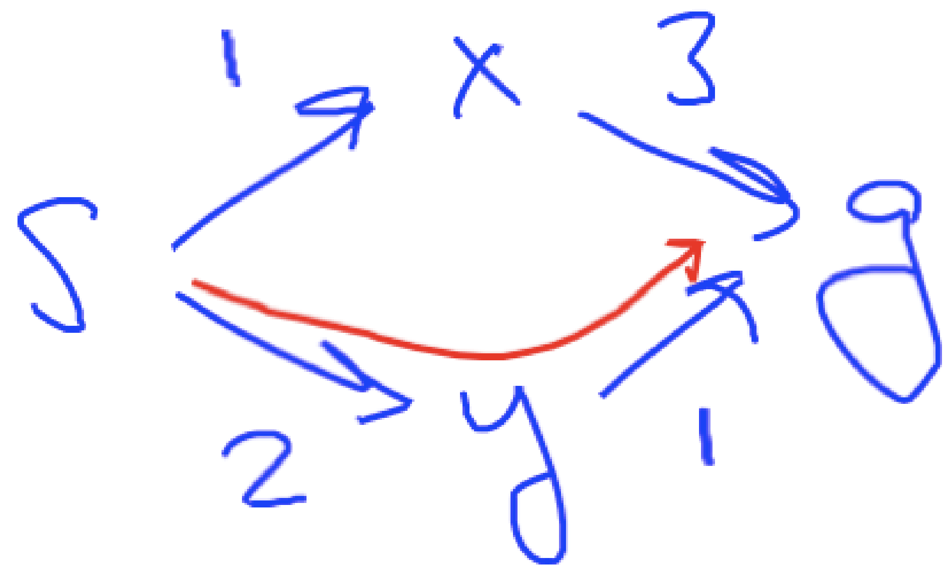
$$p_{sx} + p_{sy} = 1$$

$$-p_{sx} + p_{xg} = 0$$

$$-p_{sy} + p_{yg} = 0$$

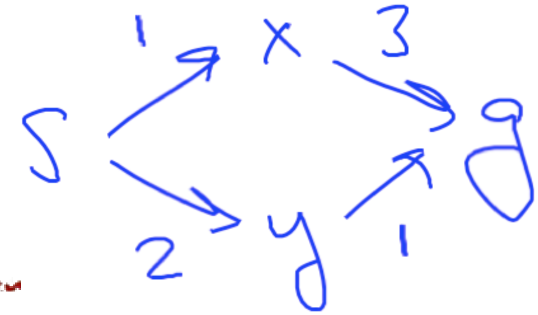
$$-p_{xg} - p_{yg} = -1$$

Optimal solution



$p_{sy} = p_{yg} = 1, \quad p_{sx} = p_{xg} = 0, \quad cost\ 3$

Matrix form



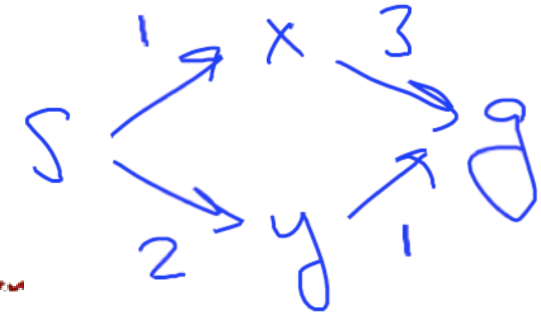
$$\min \ (1\ 3\ 2\ 1)\, p$$

st

$$\begin{pmatrix} 1 & 0 & 1 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & -1 & 0 & -1 \end{pmatrix} p = \begin{pmatrix} 1 \\ 0 \\ 0 \\ -1 \end{pmatrix}$$

$$p \geq 0$$

Matrix form



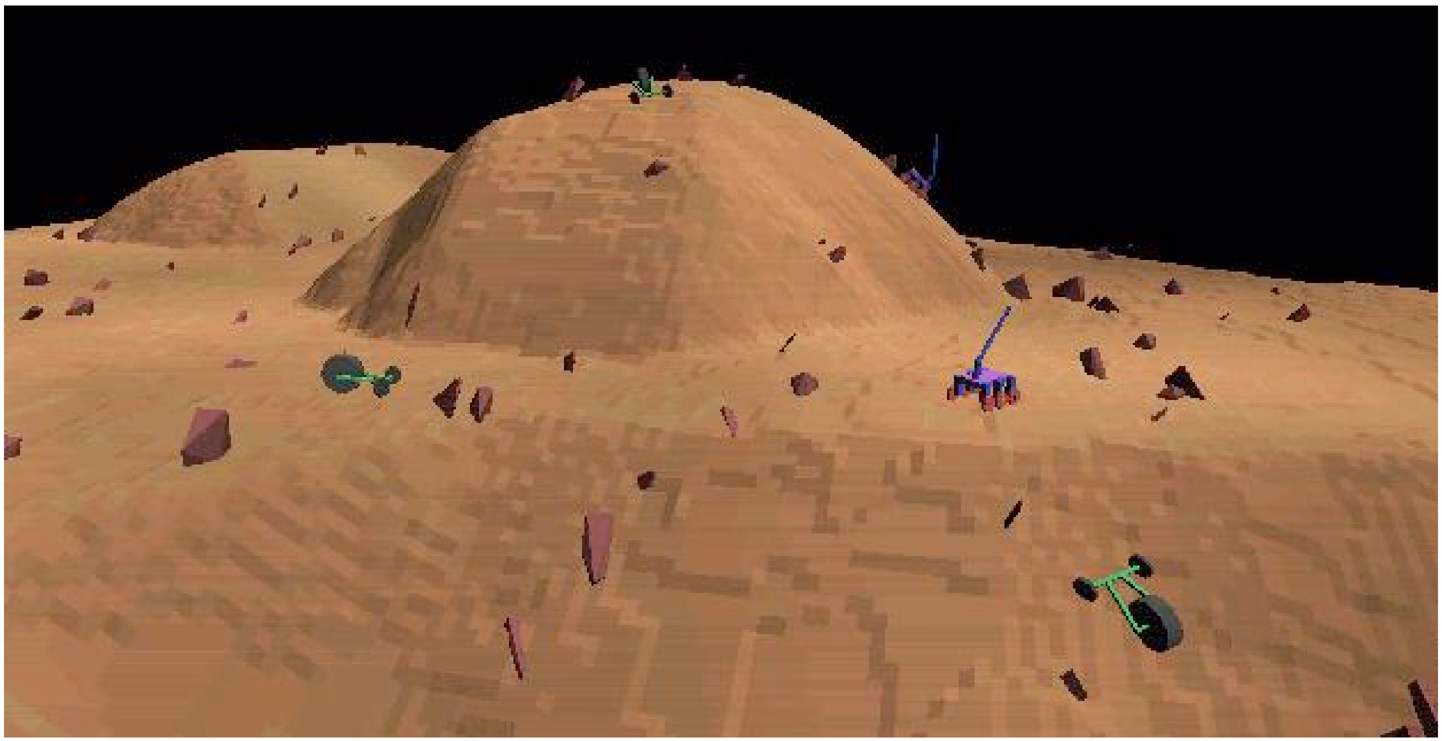
$$\min \; (1\ 3\ 2\ 1)\, p$$

st

$$\begin{pmatrix} 1 & 0 & 1 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & -1 & 0 & -1 \end{pmatrix} p = \begin{pmatrix} 1 \\ 0 \\ 0 \\ -1 \end{pmatrix}$$

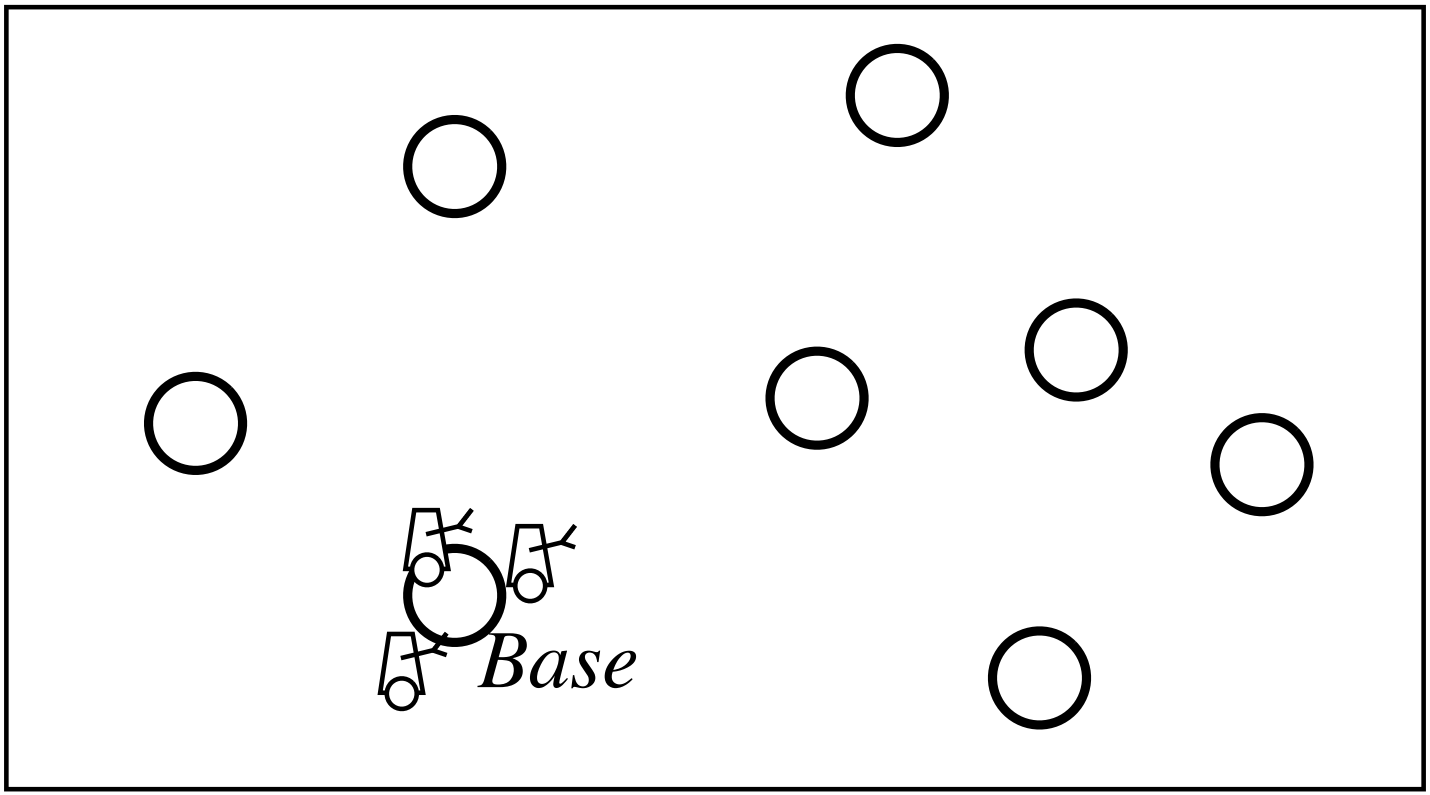
?? $p \in \{0,1\}^4$ $\quad p \geq 0$

Example: robot exploration task assignment

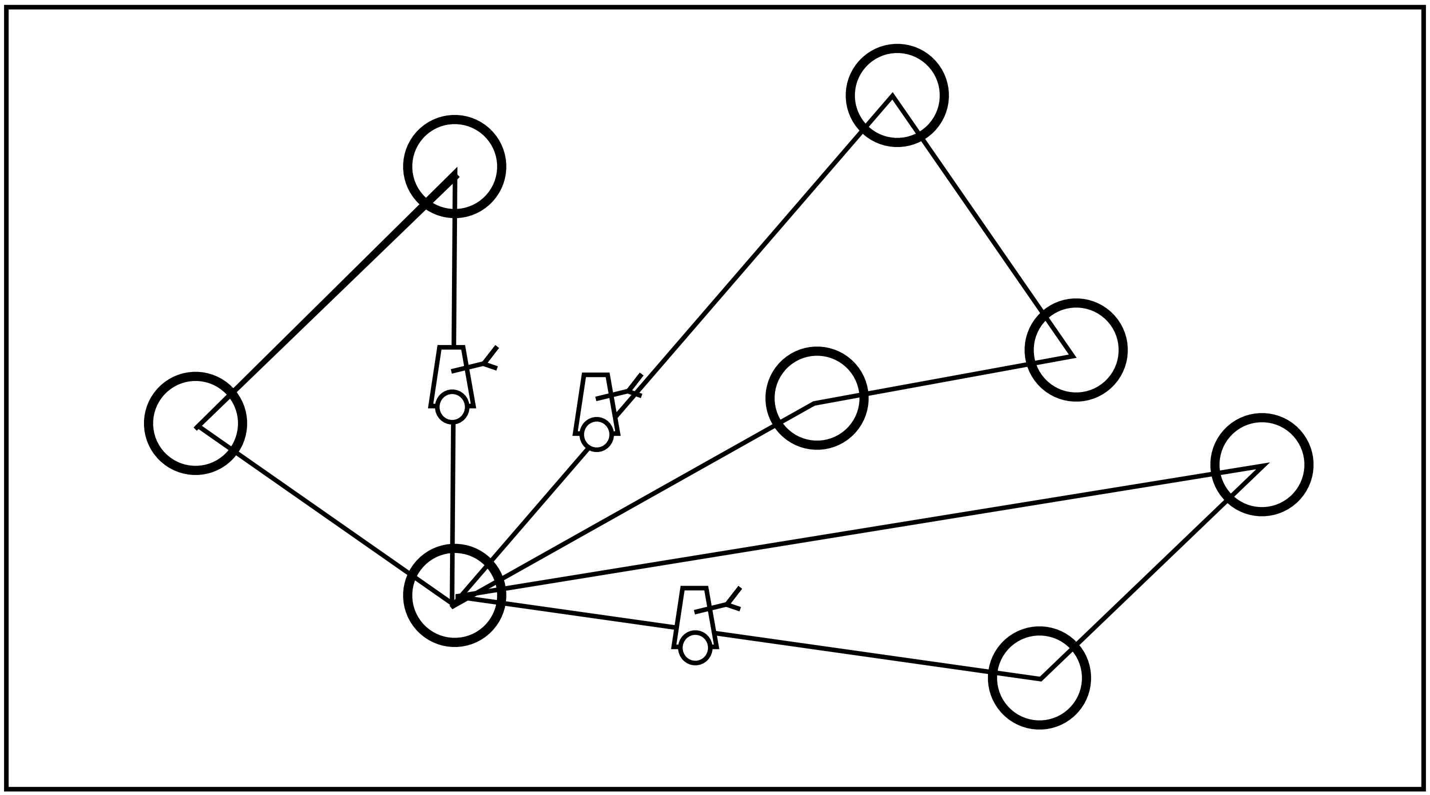


- *Team of robots must explore unknown area*

Points of interest



Exploration plan



ILP

- *Variables (all 0/1):*

  z_{ri} = *robot r does task i*

  x_{rijt} = *robot r uses edge ij at step t*

- *Minimize cost = [path cost – task bonus]*

$$\sum_{rijt} x_{rijt} c_{rijt} - \sum_{ri} z_{ri} b_{ri}$$

r indexes robots, i&j index tasks, t indexes steps

Constraints

- *Assigned tasks:* $\forall r, j, \sum_{it} x_{rijt} \geq z_{rj}$
- *One edge per step:* $\forall r, t, \sum_{ij} x_{rijt} = 1$
  - *self-loops @ base to allow idling*
- *For each i, path forms a tour from base:*
  - $\forall r, i, t, \sum_j x_{rjit} = \sum_j x_{rij(t+1)}$
  - *edges used into node = edges used out*
  - *except at times 0 and T*

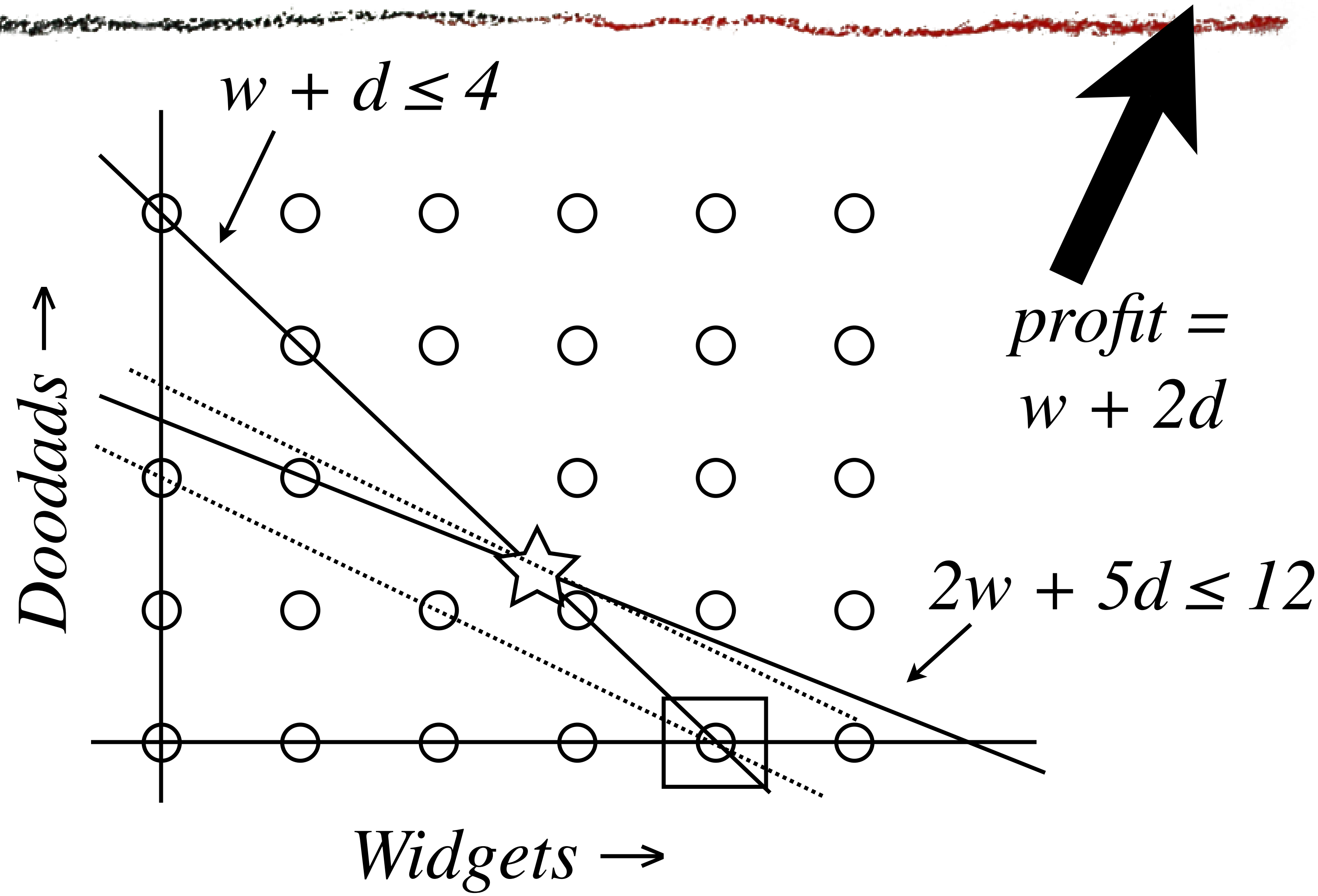
r indexes robots, i&j index tasks, t indexes steps

Duality

Branch & bound summary

- *B&B idea 1: if we have a solution with profit 3, add global constraint "profit ≥ 3"*
  - *If we then find a solution with profit 4, replace constraint with "profit ≥ 4"*
- *B&B idea 2: LP relaxations to get constraints like "profit ≤ 5 1/3" (valid at node and children)*
  - *LP may become infeasible ⇒ prune!*

Factory example



Early stopping

- *So, we have a solution of profit $4*
- *And we know the best solution has profit no more than $5 1/3*
- *If we're lazy, we can stop now*
- *Can we get smarter? Or lazier?*

What if we're *really* lazy?

- *To get our bound: had to solve the LP and find its exact optimum*
- *Can we do less work?*
- *Idea: find a suboptimal solution to LP?*
  - *Sadly, a non-optimal feasible point in the LP relaxation gives us no useful bound*

A simple bound

- *Recall:*
  - *constraint* $w + d \leq 4$ *(limit on wood)*
  - *profit* $w + 2d$
- *Since* $w, d \geq 0$,
  - $\mathit{profit} = w + 2d \leq 2w + 2d$
- *And, doubling both sides of constraint,*
  - $2w + 2d \leq 8 \Rightarrow \mathit{profit} \leq 8$

The same trick works twice

- *Try other constraint (steel use)*
  - $2w + 5d \leq 12$
- $2 * profit = 2w + 4d \leq 2w + 5d \leq 12$
- *So* $profit \leq 6$

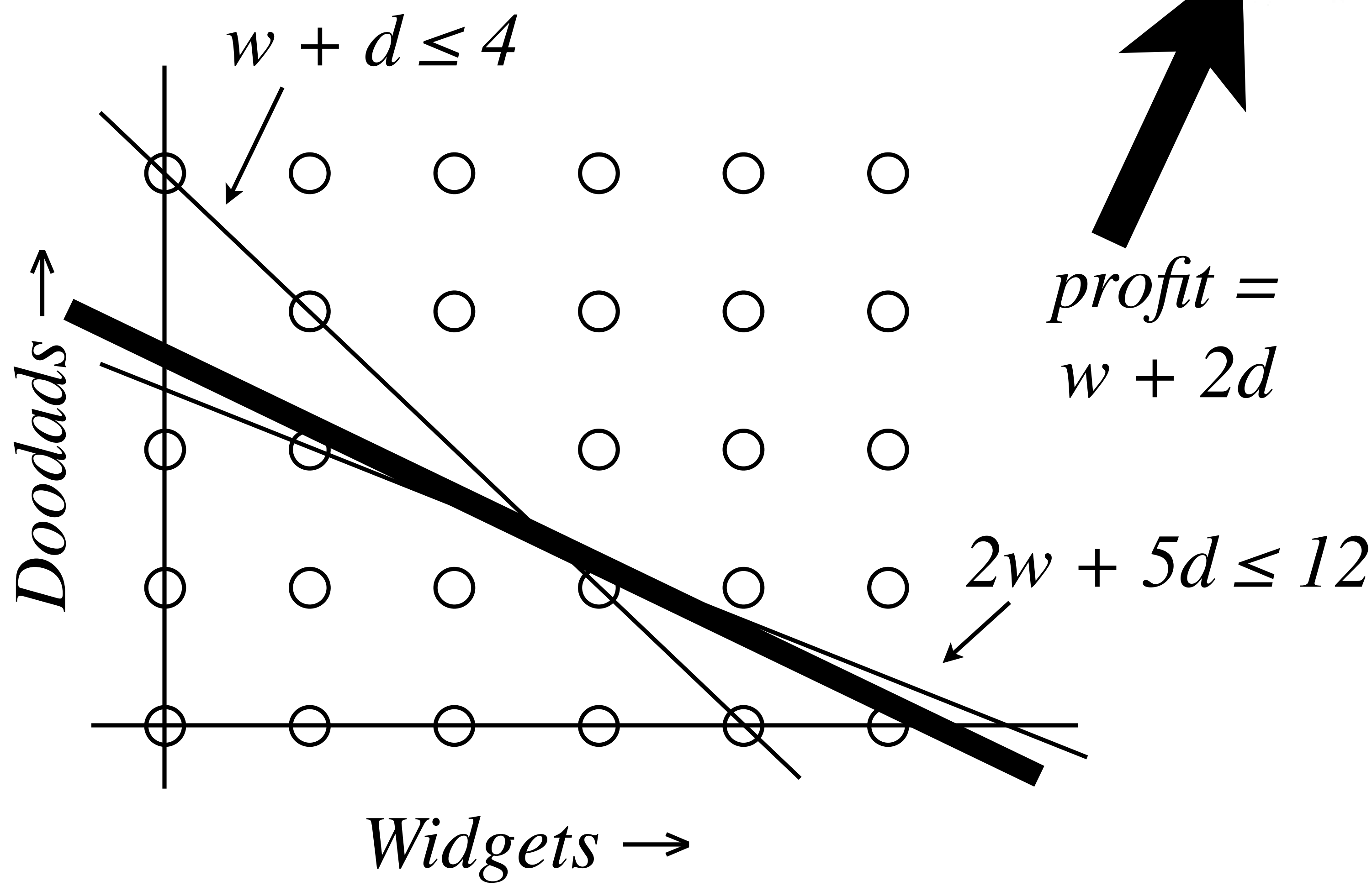
In fact it works infinitely often

- *Could take any positive-weight linear combination of our constraints*
  - *negative weights would flip sign*

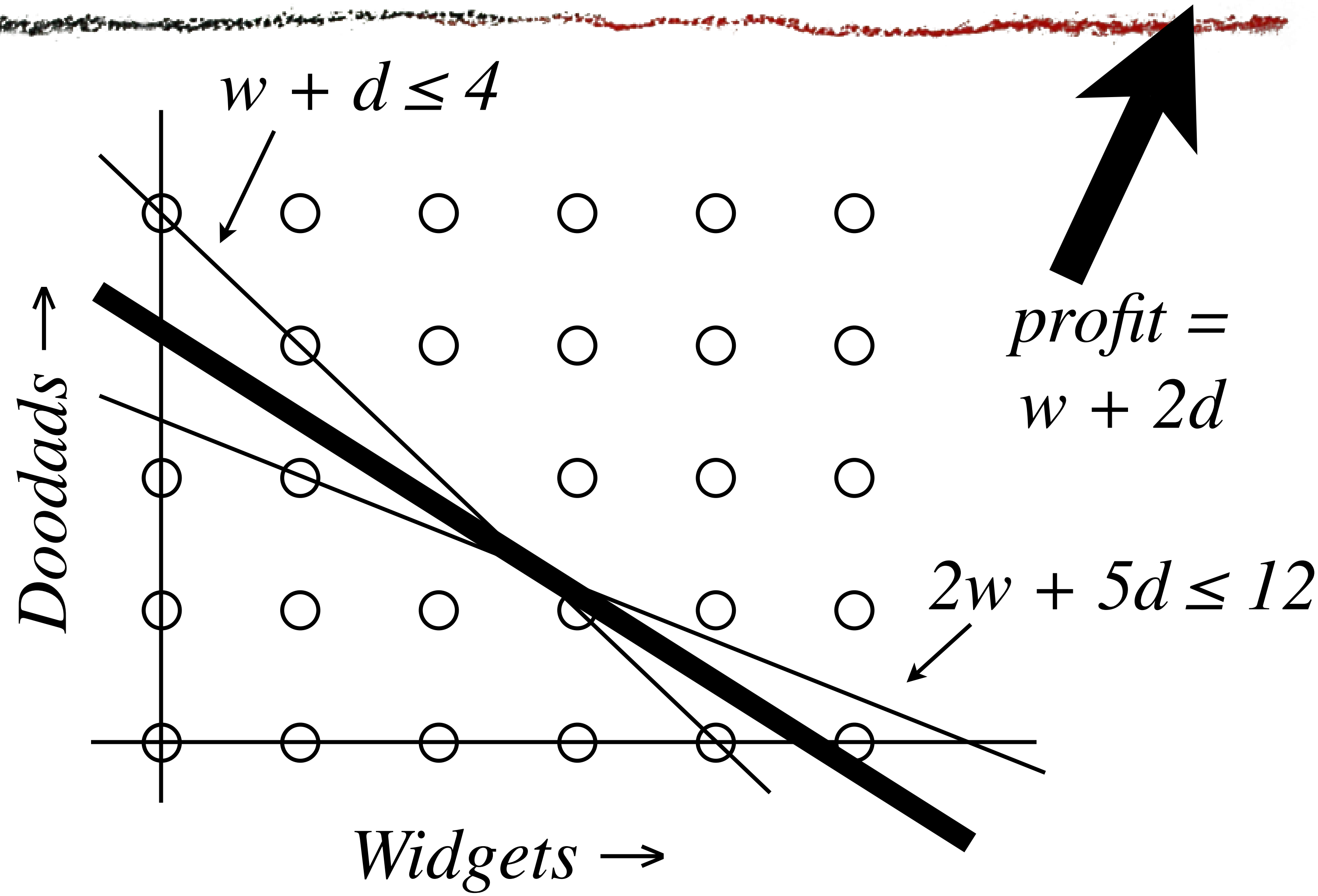
$$a\,(w + d - 4) + b\,(2w + 5d - 12) \leq 0$$

$$(a + 2b)\,w + (a + 5b)\,d \leq 4a + 12b$$

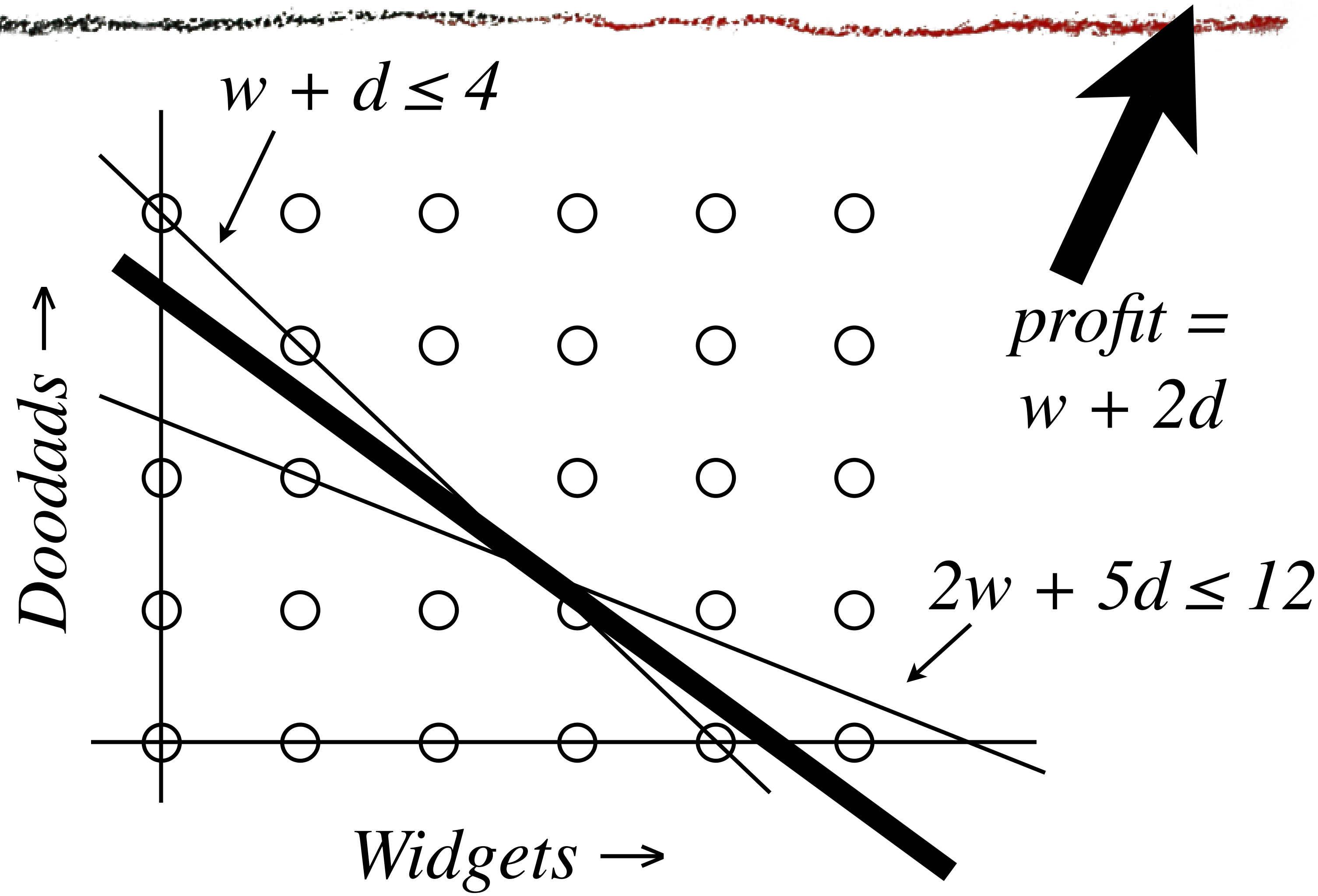
Geometrically



Geometrically



Geometrically



Bound

- $(a + 2b)\, w + (a + 5b)\, d \leq 4a + 12b$
- $profit = 1w + 2d$
- So, if $1 \leq (a + 2b)$ and $2 \leq (a + 5b)$, we know that $profit \leq 4a + 12b$

Bound

- $(a + 2b)\, w + (a + 5b)\, d \leq 4a + 12b$
- $profit = 1w + 2d$
- *So, if* $1 \leq (a + 2b)$ *and* $2 \leq (a + 5b)$, *we know that* $profit \leq 4a + 12b$

Bound

- $(a + 2b)\, w + (a + 5b)\, d \leq 4a + 12b$
- $profit = 1w + 2d$
- *So, if* $1 \leq (a + 2b)$ *and* $2 \leq (a + 5b)$*, we know that* $profit \leq 4a + 12b$

The best bound

- *If we search for the tightest bound, we have an LP:*

  minimize $4a + 12b$ *such that*

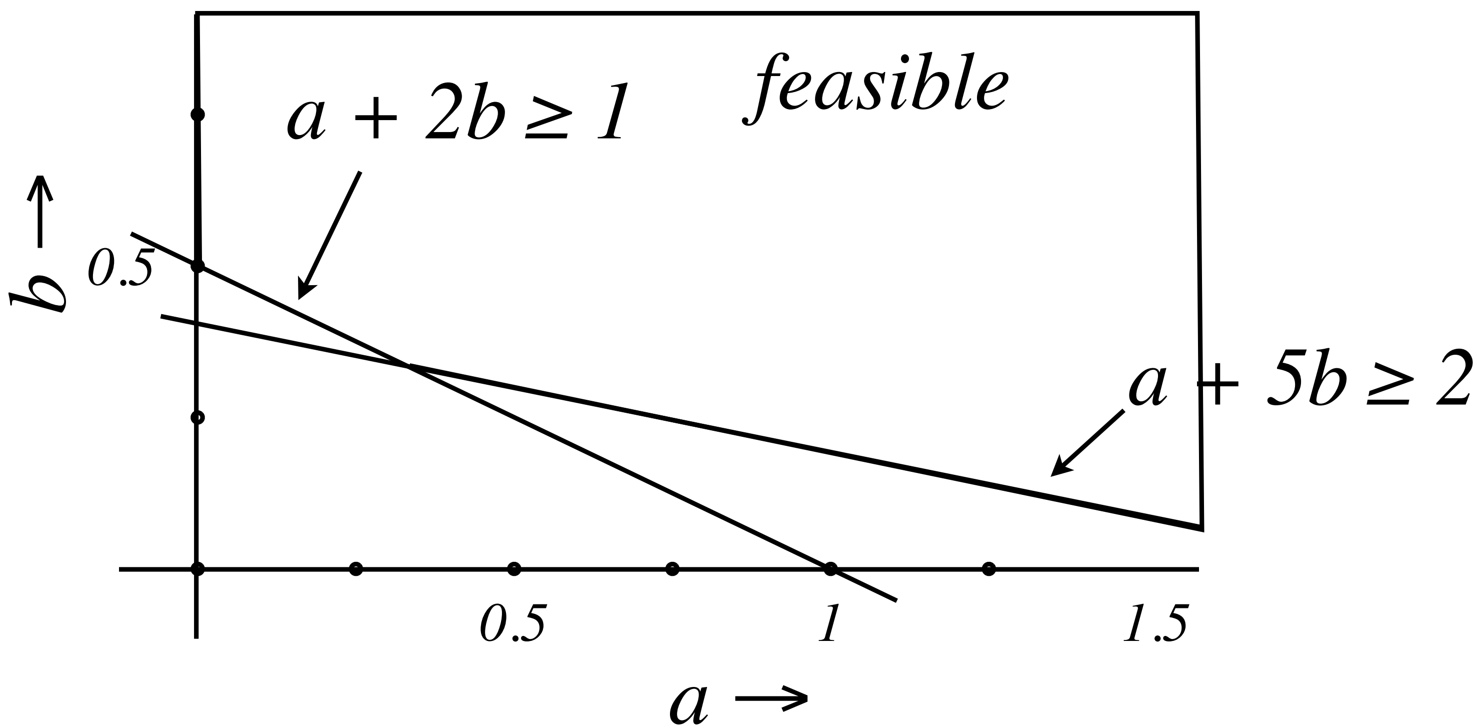
  $a + 2b \geq 1$

  $a + 5b \geq 2$

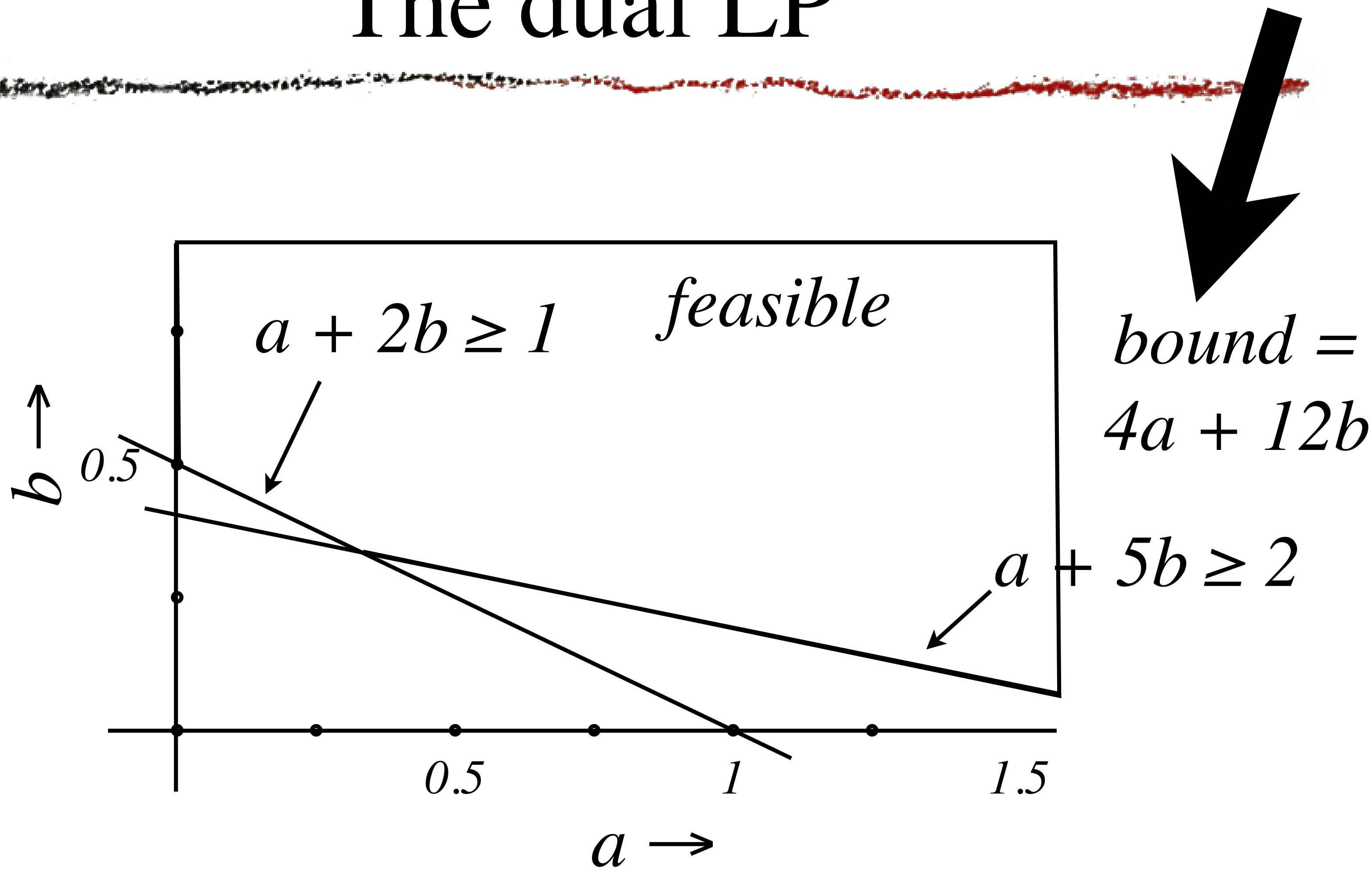
  $a, b \geq 0$

- *Called the* ***dual***

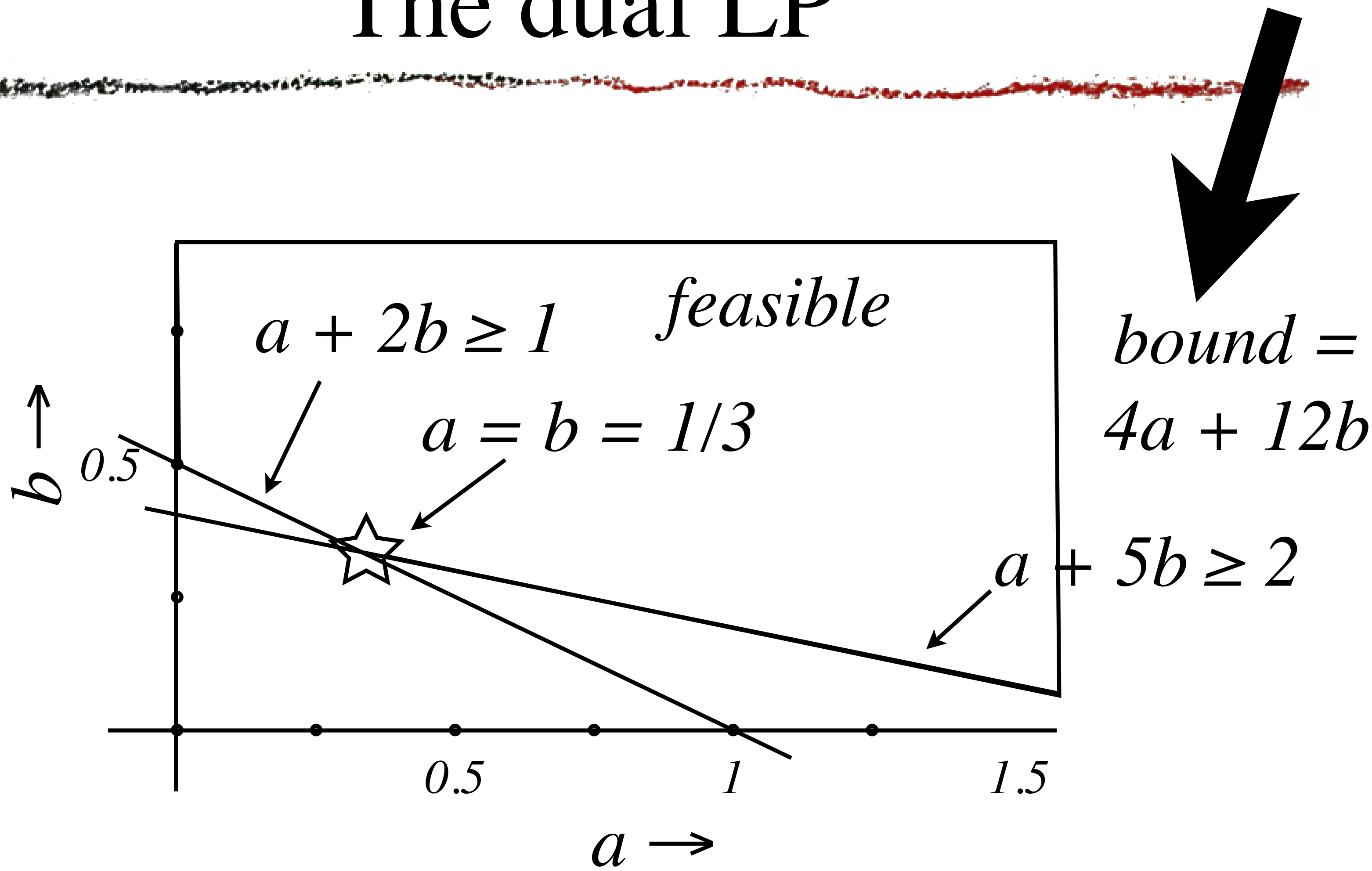
The dual LP



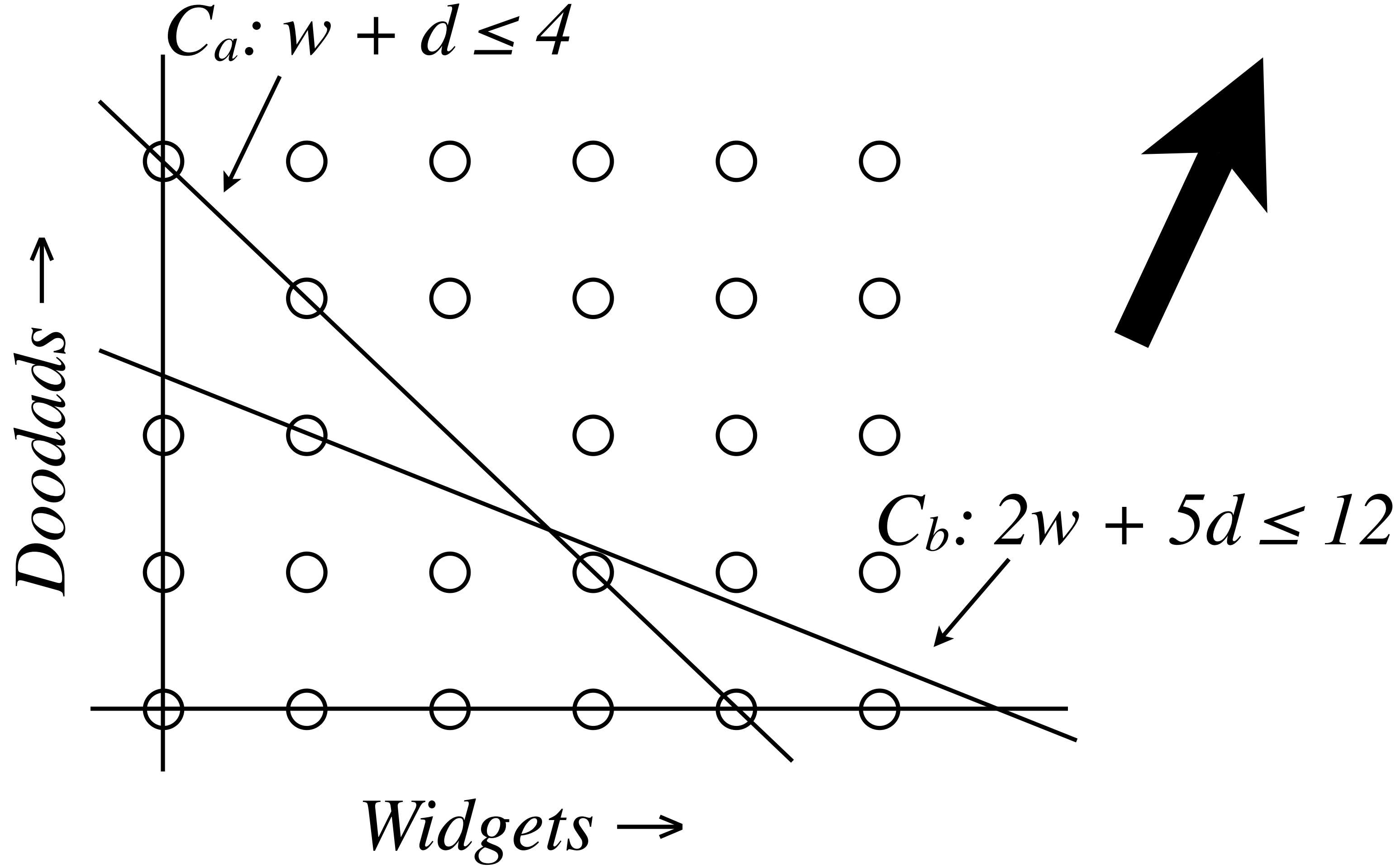
The dual LP



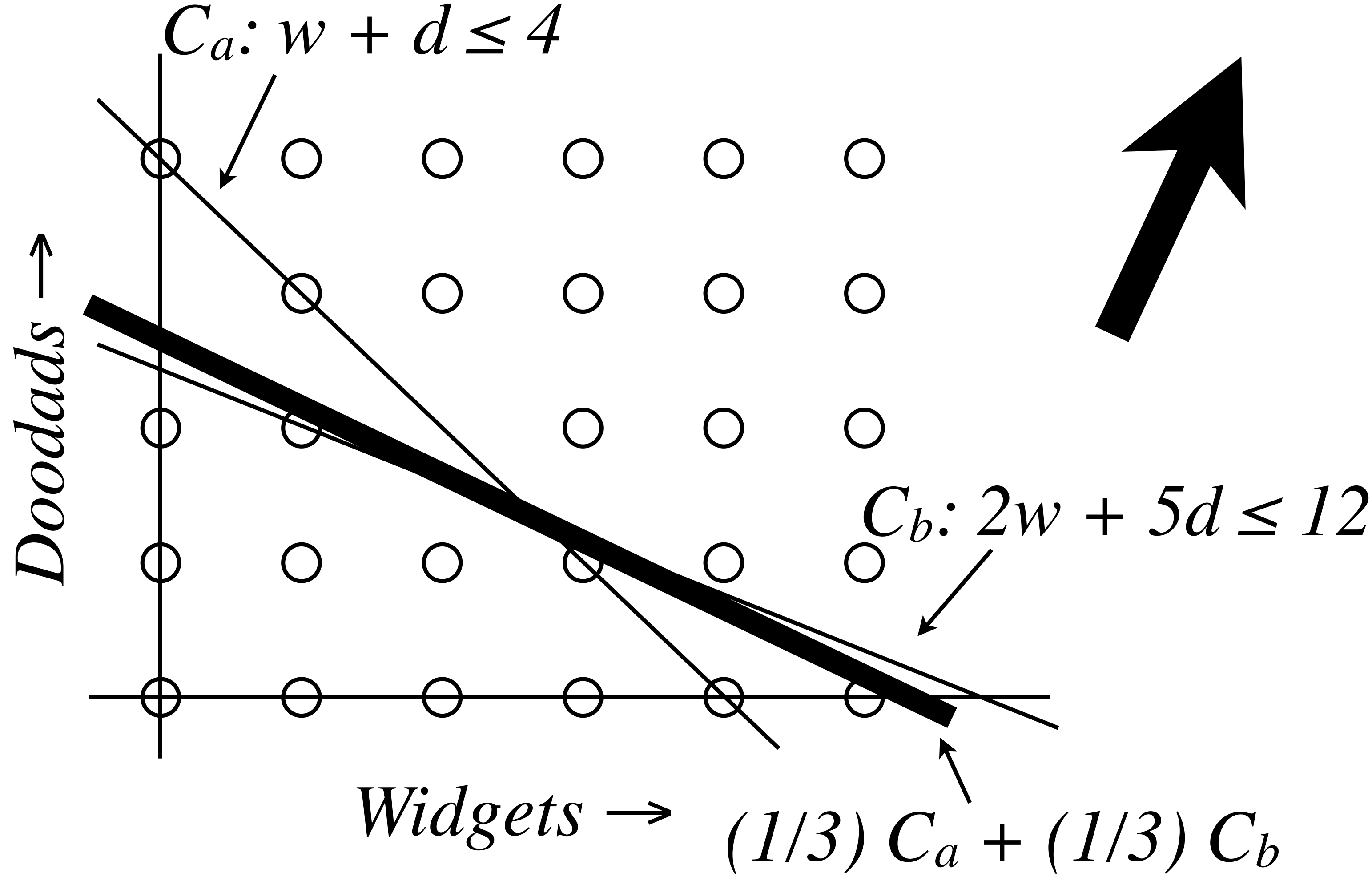
The dual LP



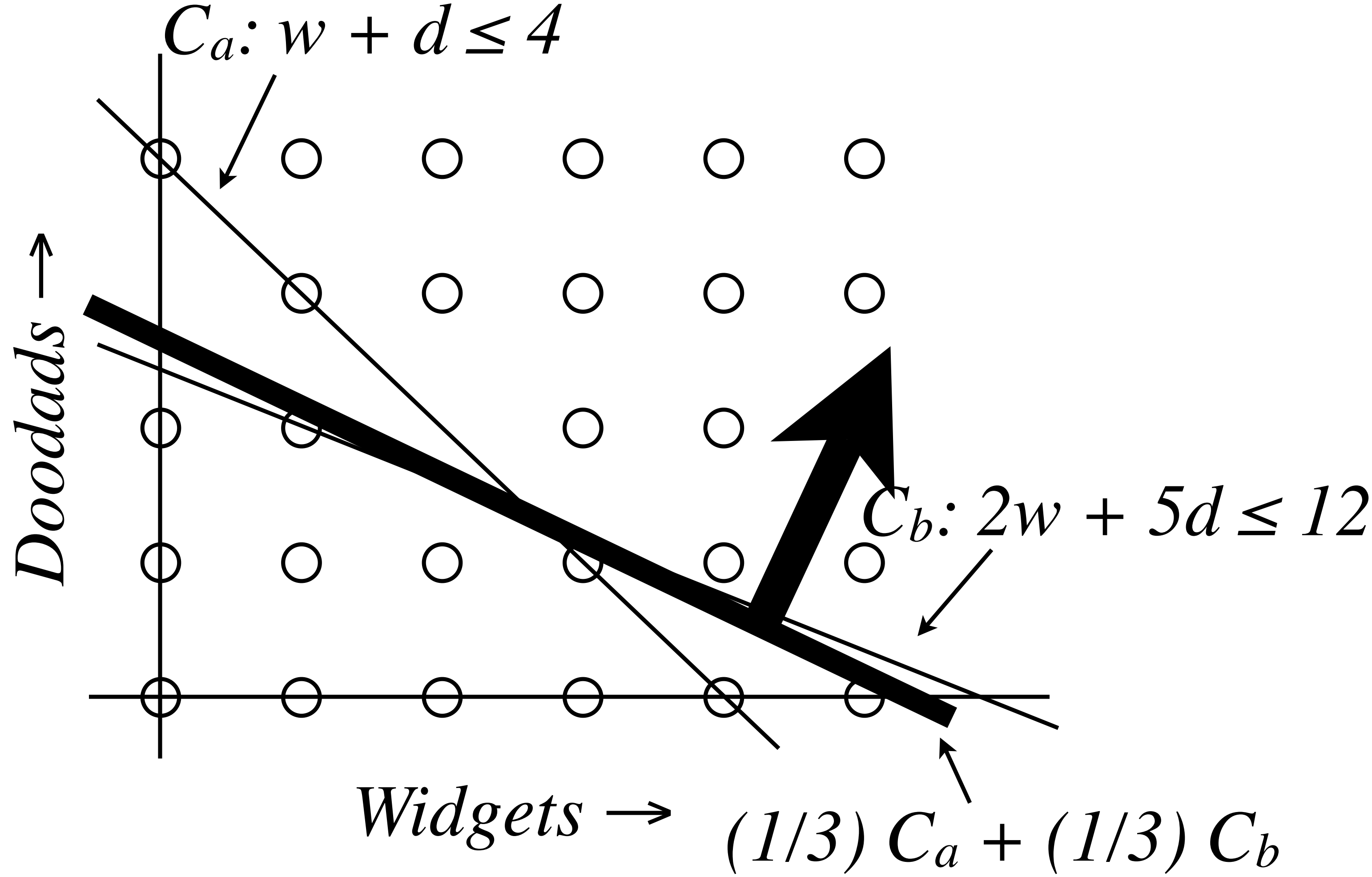
Best bound, as primal constraint



Best bound, as primal constraint



Best bound, as primal constraint



Bound from dual

- *a = b = 1/3 yields bound of*

  $4a + 12b = 16/3 = 5\ 1/3$

- *Same as bound from original relaxation!*
- *No accident: dual of an LP always* has same objective value*

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*

Recap

- *Each feasible point of dual is an upper bound on objective*
- *Each feasible point of primal is a lower bound on objective*
    - *for ILP, each integral feasible point*

Recap

- *If search in primal finds a feasible point w/ objective 4*
- *And approximate solution to dual has value 6*
    - *approximate = feasible but not optimal*
- *Then we know we're ≥ 66% of best*

Duality w/ equality

Recall duality w/ inequality

- *Take a linear combination of constraints to bound objective*
- $(a + 2b)\,w + (a + 5b)\,d \le 4a + 12b$
- $profit = 1w + 2d$
- *So, if* $1 \le (a + 2b)$ *and* $2 \le (a + 5b)$, *we know that* $profit \le 4a + 12b$

Equality example

- *minimize y subject to*
- $x + y = 1$
- $2y - z = 1$
- $x, y, z \geq 0$

Equality example

- *Want to prove bound* $y \geq \ldots$
- *Look at 2nd constraint:*

  $2y - z = 1 \quad \Rightarrow$

  $y - z/2 = 1/2$
- *Since* $z \geq 0$*, dropping* $-z/2$ *can only increase LHS* $\Rightarrow$
  - $y \geq 1/2$

Duality w/ equalities

- *In general, could start from any linear combination of equality constraints*
  - *no need to restrict to +ve combination*
- $a\,(x + y - 1) + b\,(2y - z - 1) = 0$
- $a\,x + (a + 2b)\,y - b\,z = a + b$

Duality w/ equalities

- $a\,x + (a + 2b)\,y - b\,z = a + b$
- *As long as coefficients on LHS* $\leq (0, 1, 0)$,
  - *objective* $= 0\,x + 1\,y + 0\,z \geq a + b$
- *So, maximize* $a + b$ *subject to*
  - $a \leq 0$
  - $a + 2b \leq 1$
  - $-b \leq 0$

Duality recipes

Recipe for inequalities

- *If we have an LP in matrix form,*

  maximize $c'x$ subject to

  $Ax \le b$

  $x \ge 0$

- *Its dual is a similar-looking LP:*

  minimize $b'y$ subject to

  $A'y \ge c$

  $y \ge 0$

$Ax \le b$ means every component of Ax is $\le$ corresponding component of b

Recipe with ≤ and =

- *If we have an LP with equalities,*

  maximize $c'x$ *s.t.*

  $Ax \le b$

  $Ex = f$

  $x \ge 0$

- *Its dual has some unrestricted variables:*

  minimize $b'y + f'z$ *s.t.*

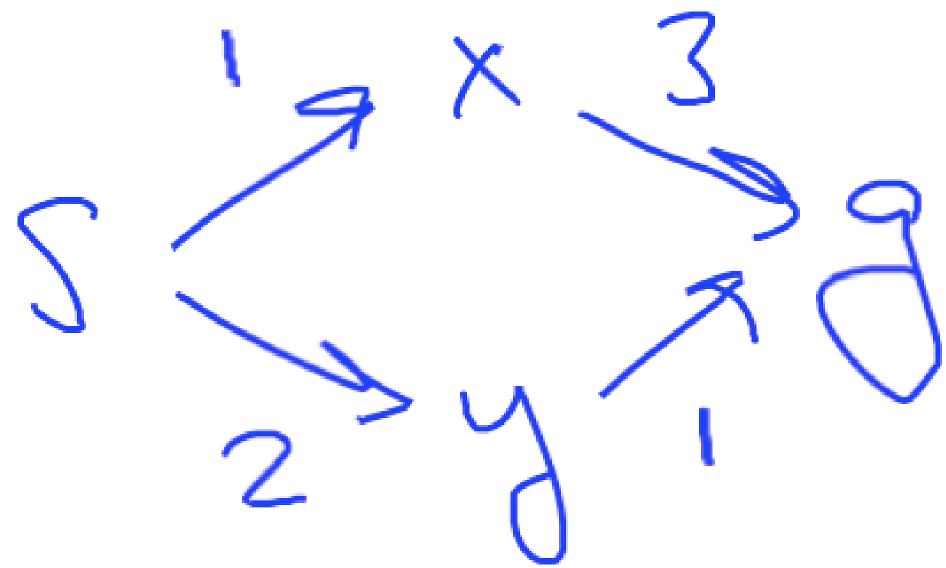
  $A'y + E'z \ge c$

  $y \ge 0$

  z *unrestricted*

Duality example

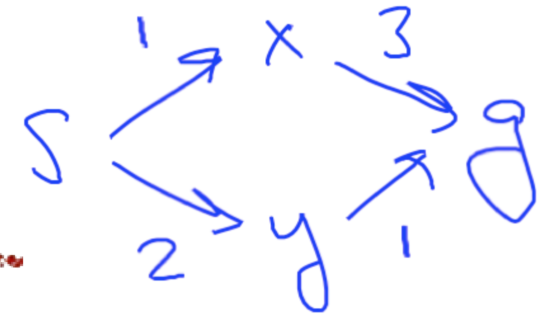
Path planning LP



- *Find the min-cost path: variables*

$p_{sx}, p_{sy}, p_{xg}, p_{yg} \geq 0$

Path planning LP



min

$$p_{sx} + 3p_{xg} + 2p_{sy} + p_{yg}$$

st

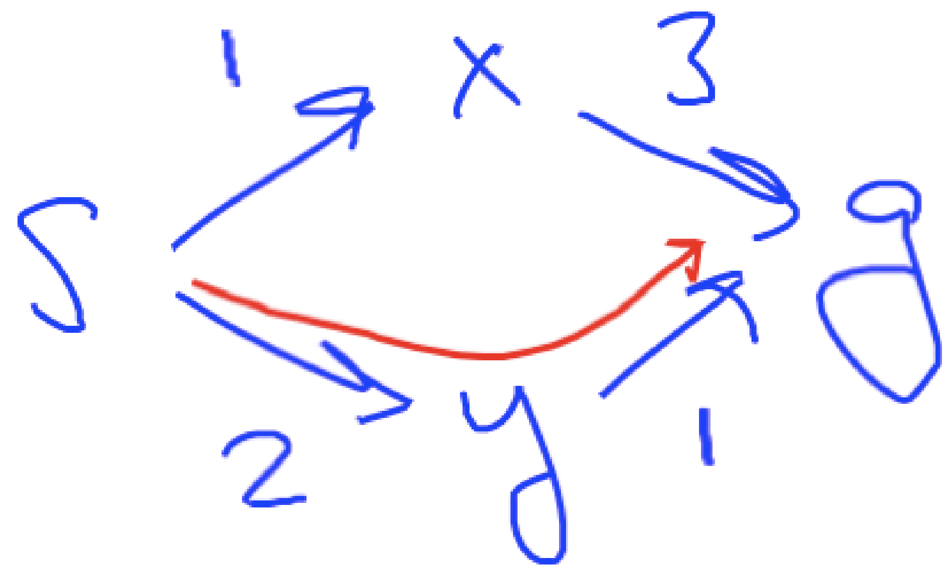
$$p_{sx} + p_{sy} = 1$$

$$-p_{sx} + p_{xg} = 0$$

$$-p_{sy} + p_{yg} = 0$$

$$-p_{xg} - p_{yg} = -1$$

Optimal solution



$p_{sy} = p_{yg} = 1, \quad p_{sx} = p_{xg} = 0, \quad cost\ 3$

Matrix form

$$\min \ (1\ 3\ 2\ 1)\, p$$

st

$$\begin{pmatrix} 1 & 0 & 1 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & -1 & 0 & -1 \end{pmatrix} p = \begin{pmatrix} 1 \\ 0 \\ 0 \\ -1 \end{pmatrix}$$

$$p \geq 0$$

Matrix form

$$\min \; (1\;3\;2\;1)\, p$$

st

$$\begin{matrix} \lambda_s \\ \lambda_x \\ \lambda_y \\ \lambda_g \end{matrix} \begin{pmatrix} 1 & 0 & 1 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & -1 & 0 & -1 \end{pmatrix} p = \begin{pmatrix} 1 \\ 0 \\ 0 \\ -1 \end{pmatrix}$$

$$p \geq 0$$

Dual

$$\max \ \lambda_s - \lambda_g$$

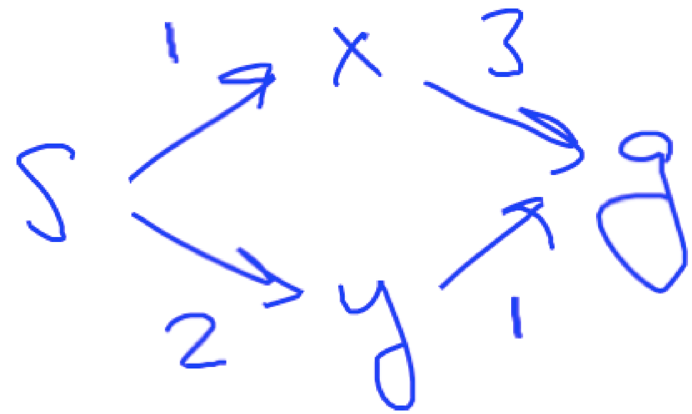
st

$$\lambda_s - \lambda_x \le 1$$

$$\lambda_x - \lambda_g \le 3$$

$$\lambda_s - \lambda_y \le 2$$

$$\lambda_y - \lambda_g \le 1$$



Optimal dual solution

$$\max \ \lambda_s - \lambda_g$$

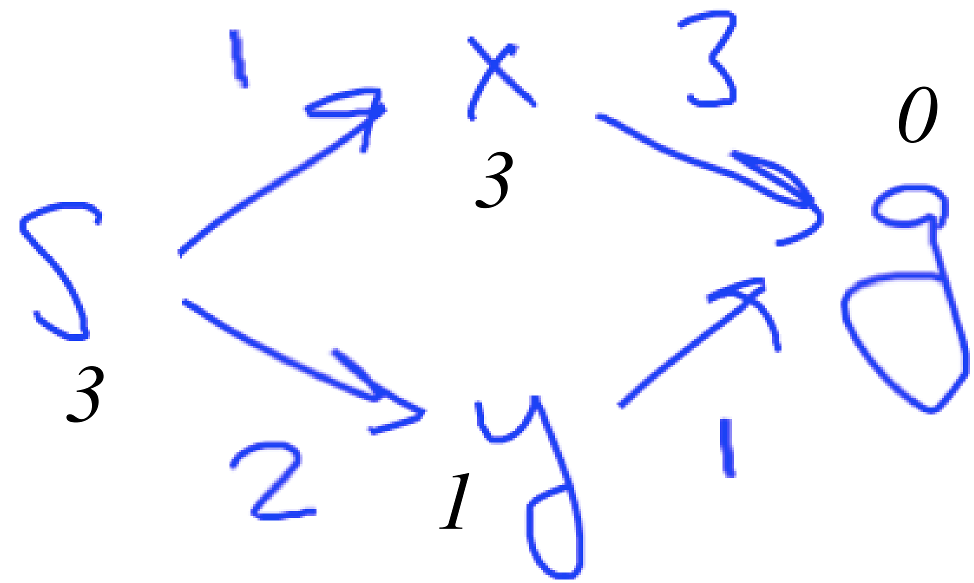
st

$$\lambda_s - \lambda_x \le 1$$

$$\lambda_x - \lambda_g \le 3$$

$$\lambda_s - \lambda_y \le 2$$

$$\lambda_y - \lambda_g \le 1$$



Any solution which adds a constant to all λs also works; $\lambda_x = 2$ also works

More about the dual

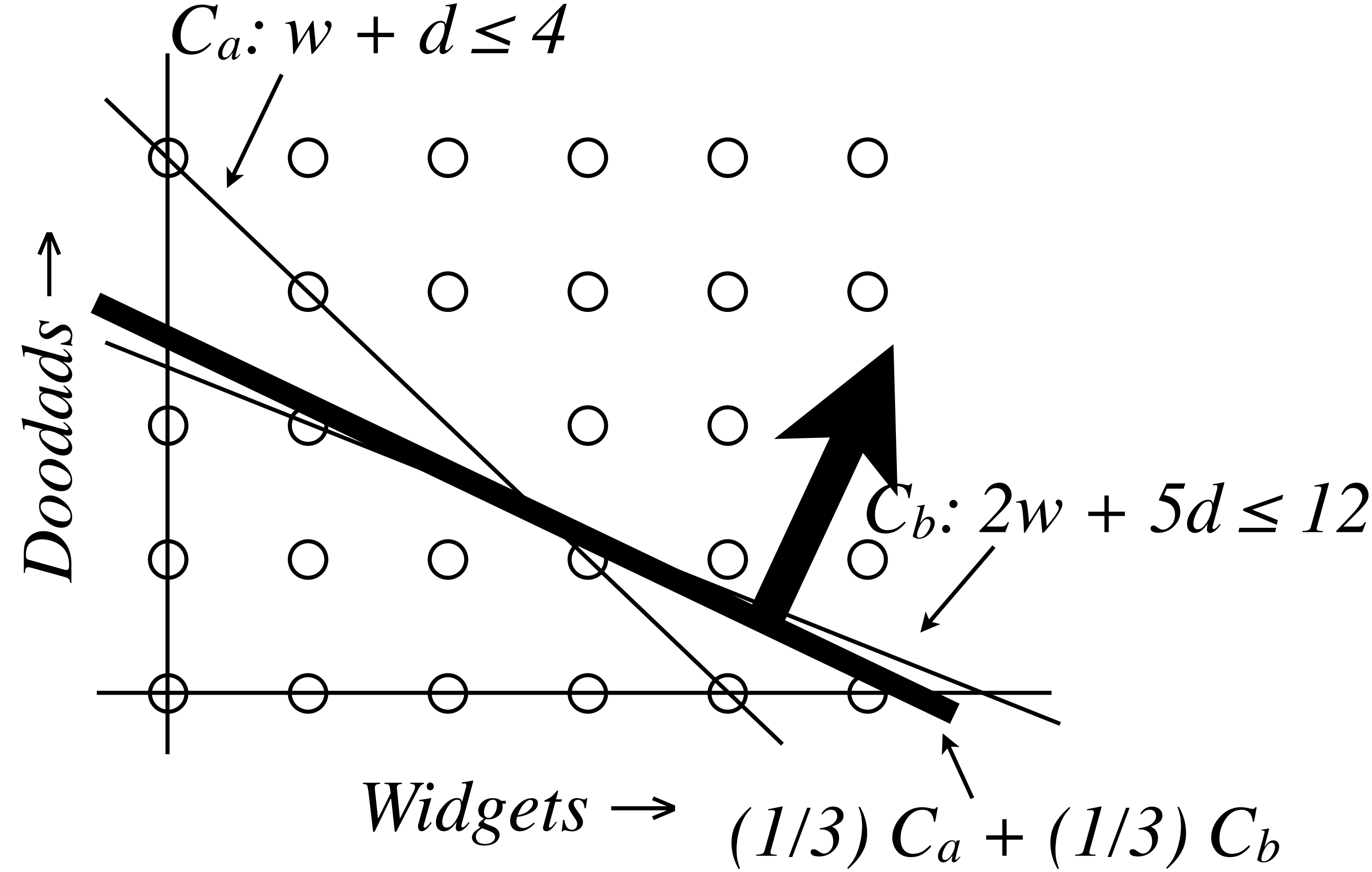
Dual dual

- *Take the dual of an LP twice, get the original LP back (called* ***primal****)*
- *Many LP solvers will give you both primal and dual solutions at the same time for no extra cost*

Interpreting the dual variables

- *The primal variable variables in the factory LP were how many widgets and doodads to produce*
- *We interpreted dual variables as multipliers for primal constraints*

Dual variables as multipliers



Dual variables as prices

- *"Multiplier" interpretation doesn't give much intuition*
- *It is 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*

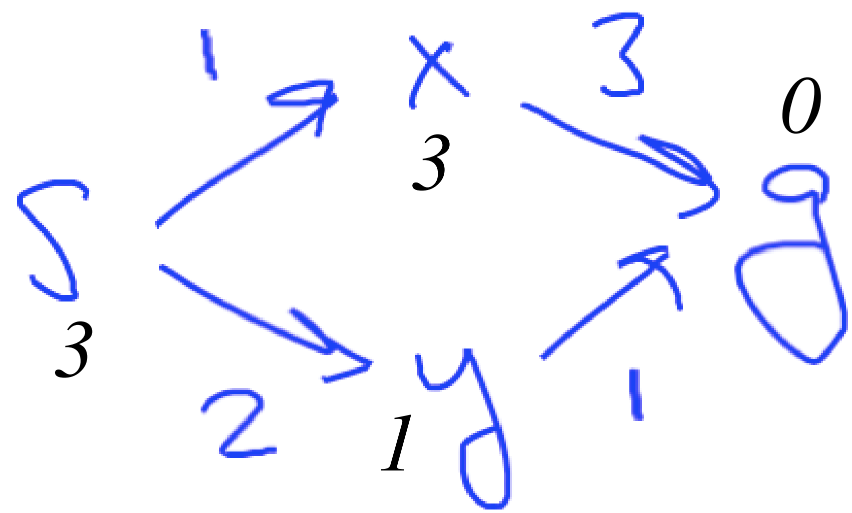
  $w + d \leq 4 + \varepsilon$

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

Dual variables as prices

- *RHS in primal is objective in dual*
- *So, dual constraints stay same, previous solution a = b = 1/3 still dual feasible*
  - *still optimal if* ε *small enough*
- *Bound changes to (4 +* ε*) a + 12 b, difference of* ε ** 1/3*
- *So we should pay up to $1/3 per unit of wood (in small quantities)*

Price example: path planning



- *Dual variables are prices on nodes: how much does it cost to start there?*
- *Dual constraints are local price constraints: edge* xg *(cost 3) means that node x can't cost more than 3 + price of node g*