· Basic idea

rely on two subroutines that (efficiently) compute a lower and an upper bound of the optimal value over a given region

upper bound can be found by choosing any point in the region, or by local minimization

lower bound can be found by convex relaxation, duality, Lipschitz or other bounds

partition feasible set into convex sets, and find lower/upper bound for each

form global lower and upper bound $\longrightarrow$ if close, quit

$\searrow$ else, refine partition and repeat

e.g. $\min c^T x$

s.t. $Ax \le b \qquad x_i \in \{0,1\} \quad i=1,2,\cdots 10$

this gives $2^{10} = 1024$ LPs

when solving one of these LPs with interior point method, each iteration gives a dual value

if take a pattern (e.g. 1,1,...), the instant dual value goes over $f(x_{best})$, quit solving for this pattern

· Unconstrained nonconvex minimization

find global minimum of function $f: \mathbb{R}^m \to \mathbb{R}$ over a m-dimensional rectangle Q_{init} to within some prescribed accuracy ϵ

for any rectangle $Q \subseteq Q_{init}$, define $\Phi_{min}(Q) = \inf_{x \in Q} f(x)$

global optimal value is $f^* = \Phi_{min}(Q_{init})$

for any rectangle $Q \subseteq Q_{init}$, use lower and upper bound functions Φ_{lb} and Φ_{ub} s.t.

$$\Phi_{lb}(Q) \le \Phi_{min}(Q) \le \Phi_{ub}(Q)$$

bounds must be tight as rectangle shrinks

$$\forall \epsilon > 0 \ \exists \delta > 0 \ \forall Q \subseteq Q_{init} \quad size(Q) \le \delta \Rightarrow \Phi_{ub}(Q) - \Phi_{lb}(Q) \le \epsilon$$

where size(Q) is diameter (length of longest edge of Q)

simplest upper bound: value of f at the middle of the rectangle

simplest lower bound: upperbound $- L/2 \times$ diameter $\quad$ L is the Lipschitz constant

to be practical, $\Phi_{ub}(Q)$ and $\Phi_{lb}(Q)$ should be cheap to compute (but good)

- Branch and bound algorithm

1. compute lower and upper bounds on f^*

   $L_1 = \Phi_{lb}(Q_{init})$ $\quad U_1 = \Phi_{ub}(Q_{init})$

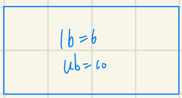
   terminate if $U_1 - L_1 \le \epsilon$

2. partition Q_{init} into 2 rectangles $Q_{init} = Q_1 \cup Q_2$
3. compute $\Phi_{lb}(Q_i)$ and $\Phi_{ub}(Q_i)$ $\quad i = 1,2$
4. update lower and upper bounds on f^*

   update lower bound : $L_2 = \min\{\Phi_{lb}(Q_1), \Phi_{lb}(Q_2)\}$

   update upper bound : $U_2 = \min\{\Phi_{ub}(Q_1), \Phi_{ub}(Q_2)\}$

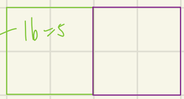
5. refine partition by spliting Q_1 or Q_2 and repeat step 3 and 4

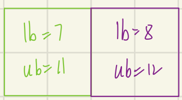


Q_{init}

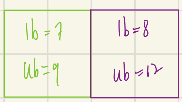
any point in Q_{init} has objective value bigger than 6,
but not necessarily smaller than 10
ub and lb are bounds on optimal value

valid but no use
can replace it with lb = 6

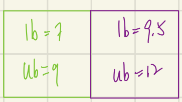




when call the upper bound on the parent, can ask for x where the upperbound is achieved
the child that contains x can't have an upperbound that's worse
one of ub = 11 and ub = 12 can be modified to ub = 10



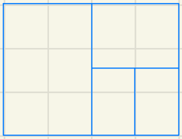
new global lower bound 7 $\quad (\min\{7, 8\})$
new global upper bound 9

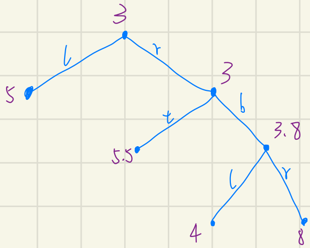


the optimal point is on the left side
lb > ub $\quad$ new bounds are 7 and 9

there is a x s.t. $f(x) = 9$ in the left child
every x in the right child have $f(x) \ge 9.5$

Can assume, without loss of generality, that U_i is nonincreasing, L_i is nondecreasing at each step we have a partially developed binary tree



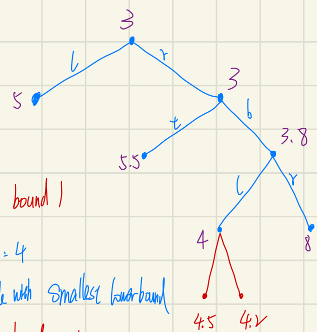


for every node here, we have a lower and upper bound
the leaves constitute Q_{init}

if the lower bounds of each node
the lower bound of the optimal value over Q_{init} is 4

the actual certificate proving the lower bound is a partition of the original space
with each partition has it's own lower bound

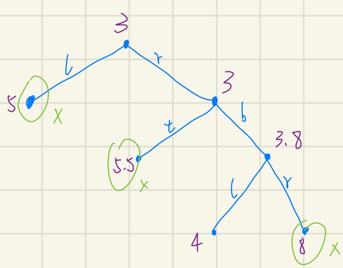
- at each step, need rules for choosing
  which rectangle to split
  which variable (edge) to split
  where to split (what value of variable)



Some good rules: split rectangle with smallest lower bound (pull up the lower bound)
along longest edge
in half

global lower bound = 4
split the rectangle with smallest lower bound
new global lower bound 4.2
4 → 4.2 progress in lower bound

can prune any rectangle Q in tree with $\Phi_{lb}(Q) > U_k$



suppose we have found a point with $f(x) = 4.2$
4.2 is a upper bound, can prune all rectangles with lower bound larger than 4.2

- eg. mixed boolean - convex problem

$$\min_{x,z} \; f_0(x,z)$$
$$\text{s.t.} \; f_i(x,z) \le 0$$
$$z_i \in \{0,1\}$$

f_i convex in both x and z

lower bound via convex relaxation

$$\min_{x,z} \; f_0(x,z)$$
$$\text{s.t.} \; f_i(x,z) \le 0$$
$$0 \le z_i \le 1$$

optimal value is a lower bound on p^*

L_1 can be $+\infty$, which implies original problem infeasible

upper bound:

1. can find a upper bound by rounding z_i^* to 0 or 1
2. can round z_i then solve for x
3. random generate $z_i \in \{0,1\}$ $p(z_i=1) = z_i^*$

convex relaxation followed by local optimization works shockingly well

branching

pick any index k, form 2 subproblems

$$\min_{x,z} \; f_0(x,z) \quad \text{s.t.} \; f_i(x,z) \le 0, \; z_j \in \{0,1\}, \; z_k = 0$$

$$\min_{x,z} \; f_0(x,z) \quad \text{s.t.} \; f_i(x,z) \le 0, \; z_j \in \{0,1\}, \; z_k = 0$$

branch and bound algorithm

continue to form binary tree by splitting, relaxing, calculating bounds on subproblems

cannot go more than 2^n steps before $U = L$

can prune nodes with L exceeding current U_k

pick variable to split:

- least ambivalent choose k for which $z^* = 0$ or 1 with largest lagrange multiplier
- most ambivalent choose k for which $|z_k^* - 1/2|$ is minimum

eg. minimum cardinality example

$$\min. \ \mathrm{card}(x)$$
$$\text{s.t.} \ Ax \le b$$

$$\Updownarrow$$

$$\min. \ \mathbf{1}^T z$$
$$\text{s.t.} \ L_i z_i \le x_i \le U_i z_i$$
$$Ax \le b$$
$$z_i \in \{0, 1\}$$

$L_i = \min \ x_i$ s.t. $Ax \le b$

$U_i = \max \ x_i$ s.t. $Ax \le b$

$\Updownarrow$ relaxation

$$\min. \ \mathbf{1}^T z$$
$$\text{s.t.} \ L_i z_i \le x_i \le U_i z_i$$
$$Ax \le b$$
$$0 \le z_i \le 1$$

$\Longleftrightarrow$ assuming $L_i < 0 < U_i$

$$\min. \ \sum_{i=1}^{n} 1/U_i \cdot (x_i)_+ + (-1/L_i)(x_i)_-$$
$$\text{s.t.} \ Ax \le b$$

the relaxed problem is the L_1 norm heuristic for find a sparse solution

take $\mathrm{card}(x^*)$ as upper bound