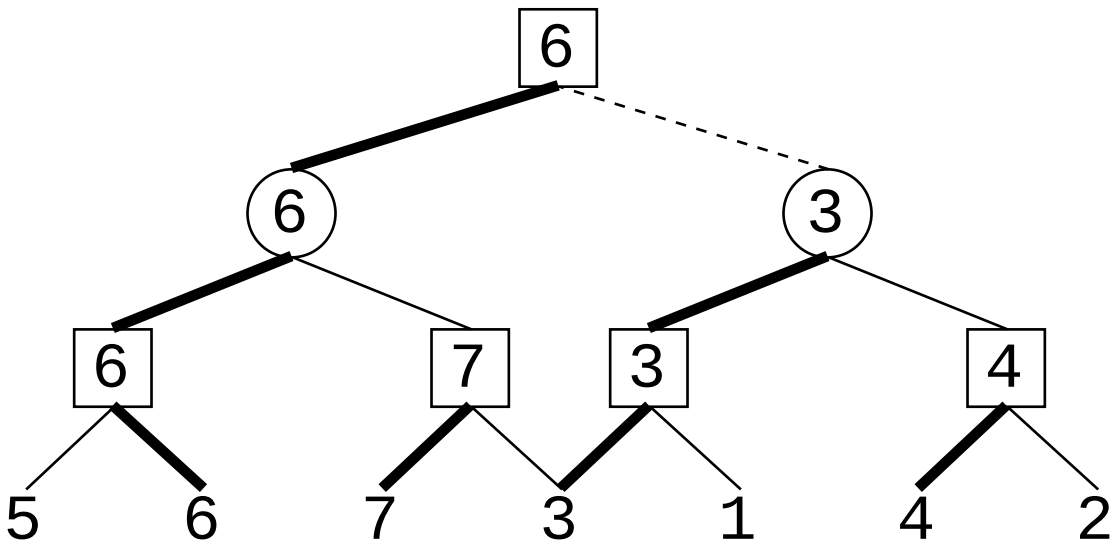
Tuning Evalution Functions for Search

Chris McConnell
Carnegie Mellon University

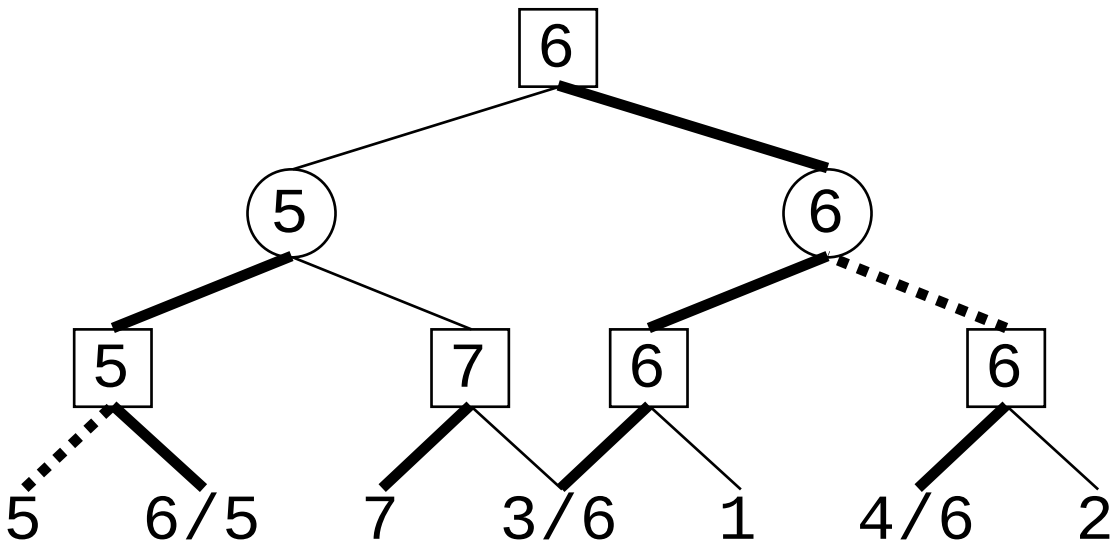
November 8, 1995

Work with:
Dr. Hans Berliner
Dr. Merrick Furst

Example DAG

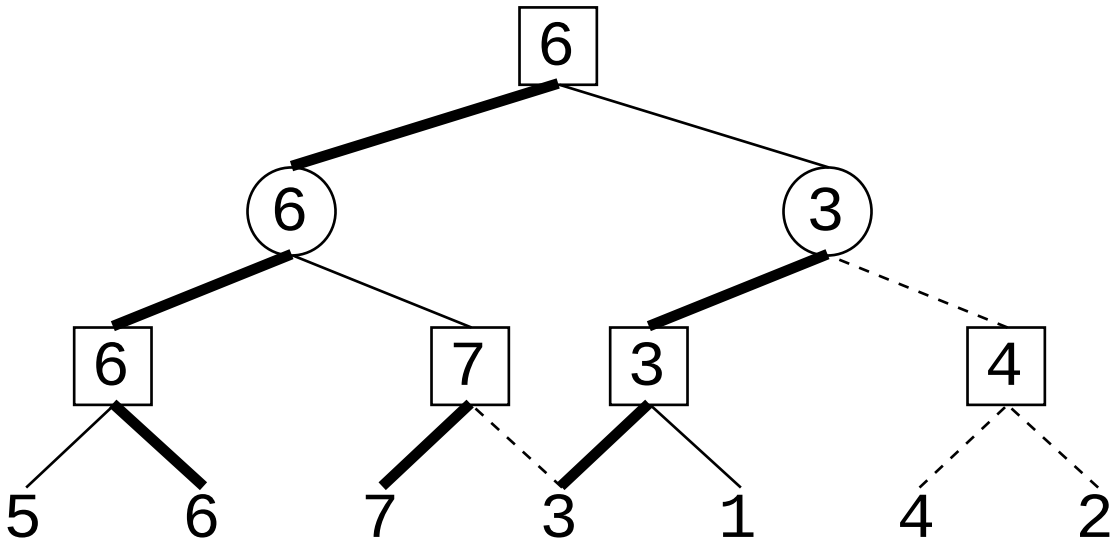


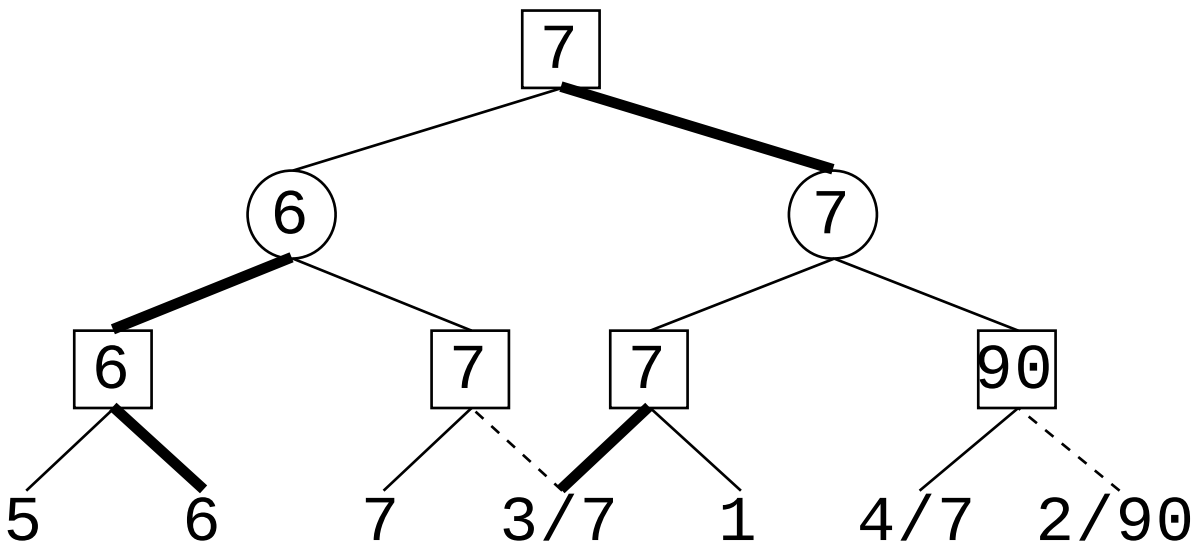
Possible Solution



History

- Long history (Samuel, 1959)
  - Global
  - Lots of data
  - Shallow
- This work
  - Local
  - Limited data
  - Deep
- Hope to generalize

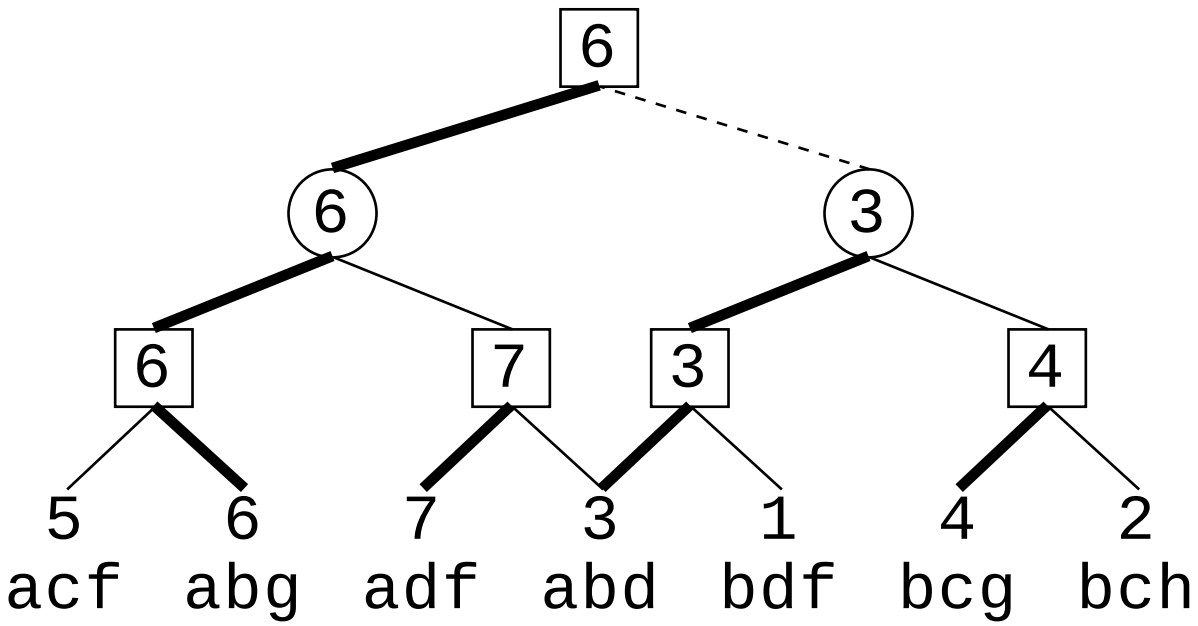

6
6
3
6
7
3
4
5
6
7
3
1
4
2


7
6
7
6
7
7
90
5
6
7
3/7
1
4/7
2/90

What is a good evaluation function?

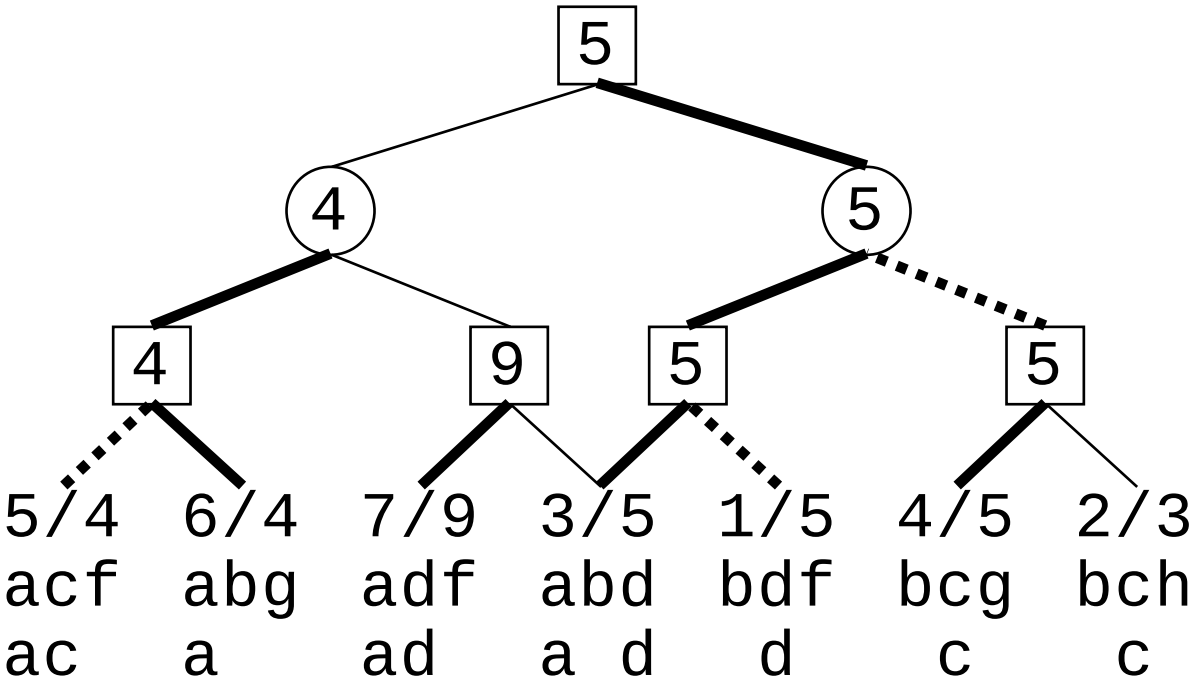
- One that works well in conjunction with search
- Reasonable assumption: $E(b) = \Sigma_i \, c_i F_i$
- Find c_i s.t. $E(b)$ changes path of search

Evaluation Interactions



- $E(n) = 3a - 3b + c + 3d + f + 6g + 4h$

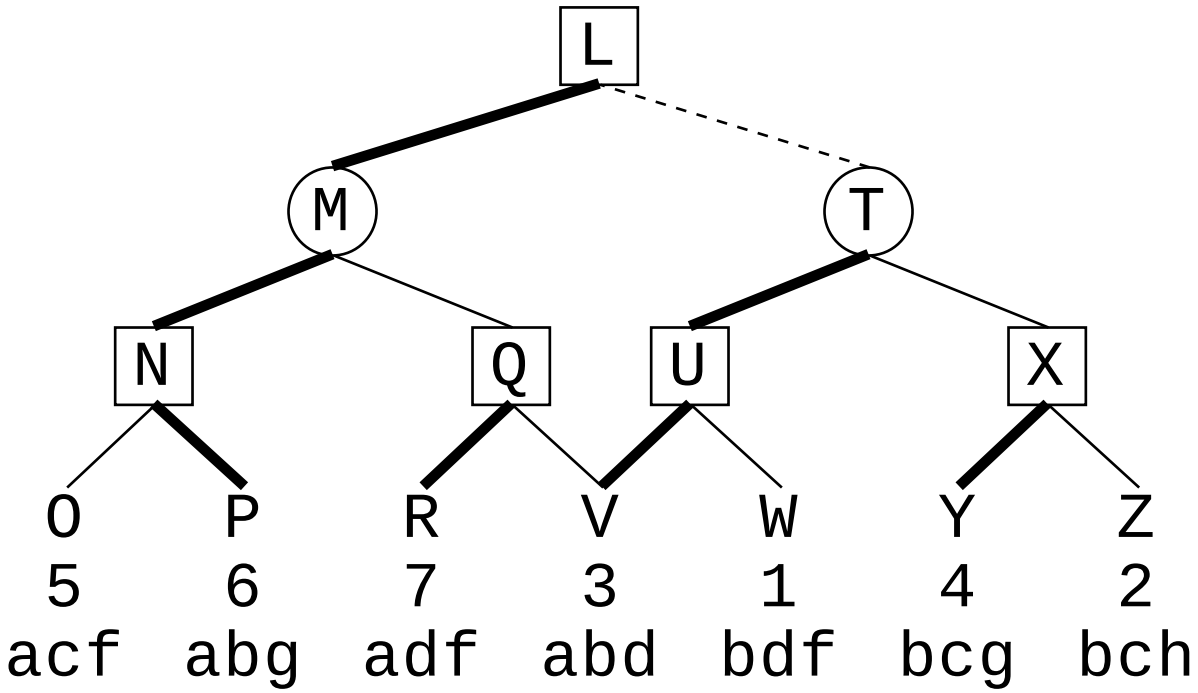
A Solution

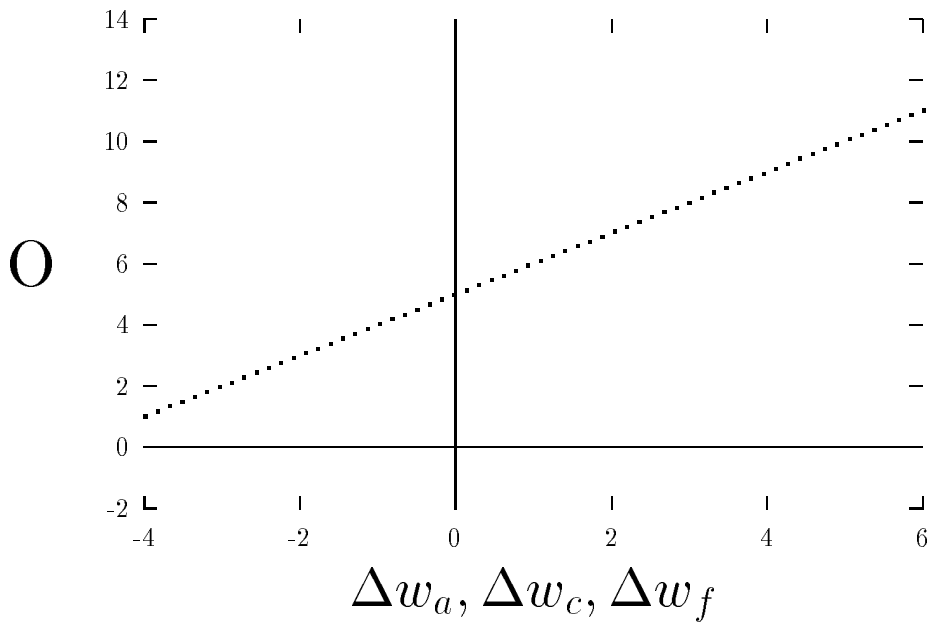


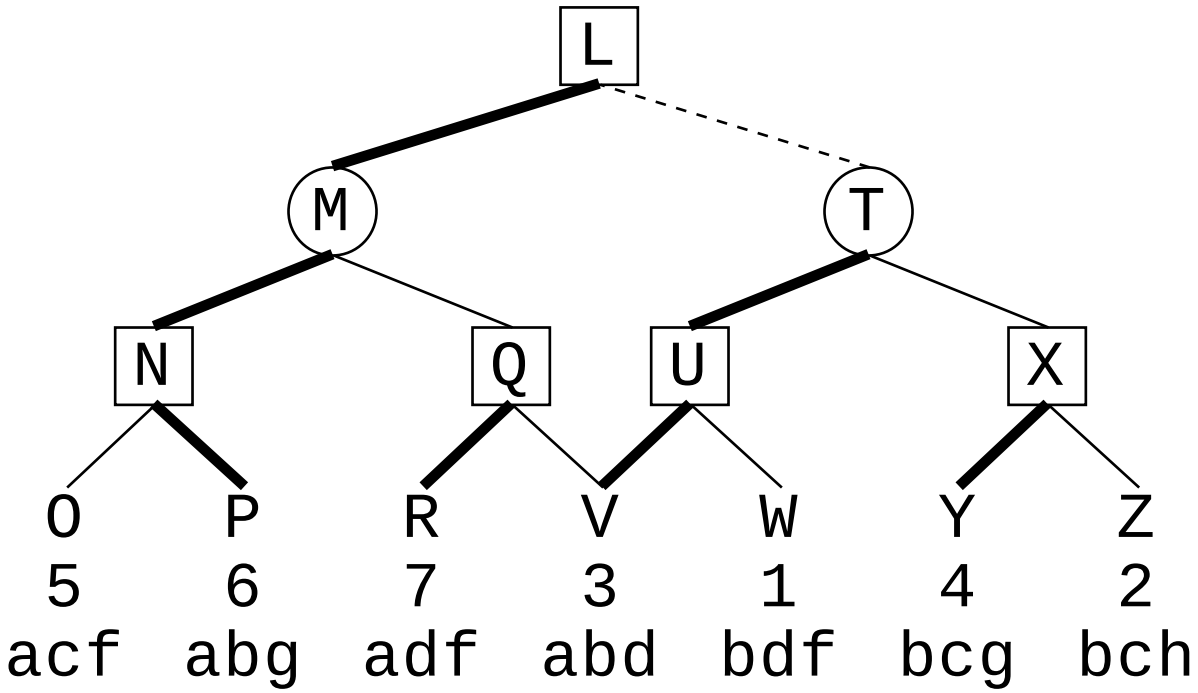
- $E(n) = 3a - 3b + c + 3d + f + 6g + 4h$
- $\Delta a = -2, \Delta c = 1, \Delta d = 4$
- $E'(n) = 1a - 3b + 2c + 7d + f + 6g + 4h$
- $Cost = \mid \Delta a \mid + \mid \Delta c \mid + \mid \Delta d \mid = 7$

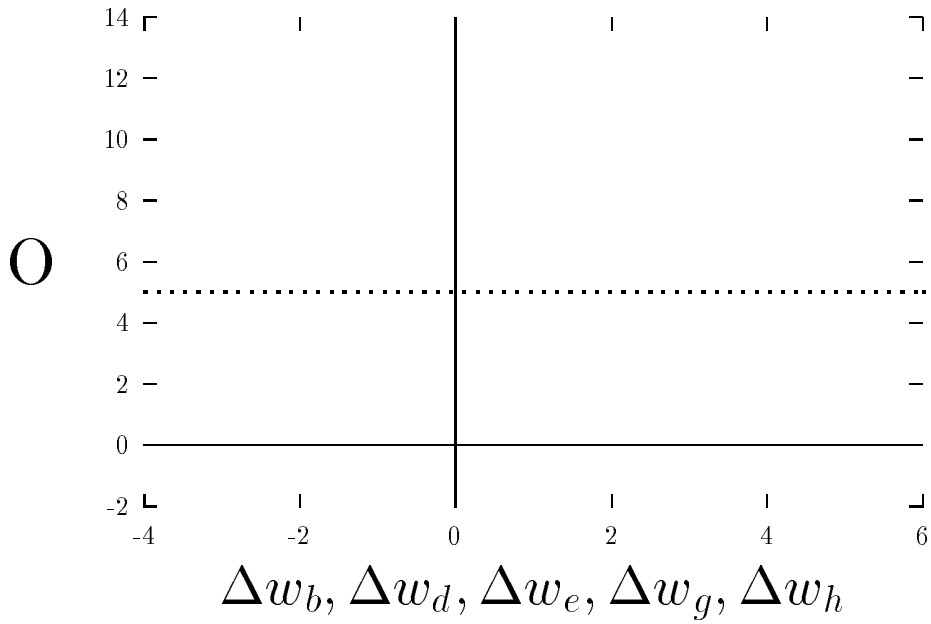
Optimization

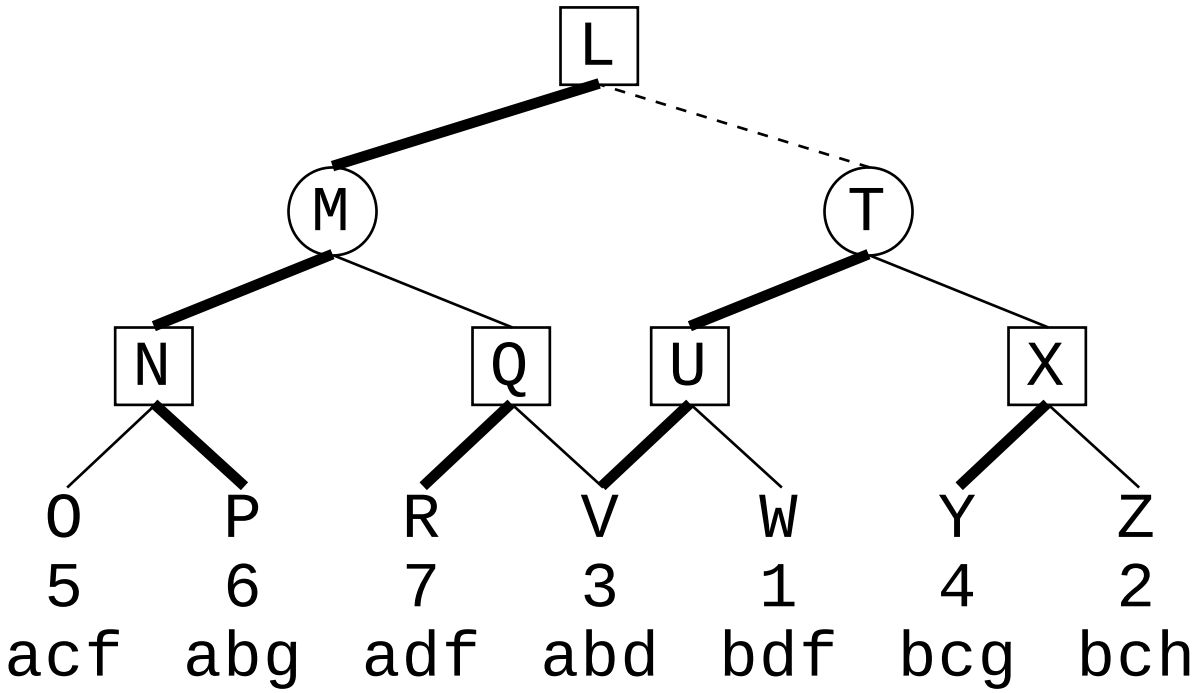
- Non-differentiable optimization
- Lagrange multplier methods
- Heuristic search
- Integer programming
- Linear programming with search
- Subgradient

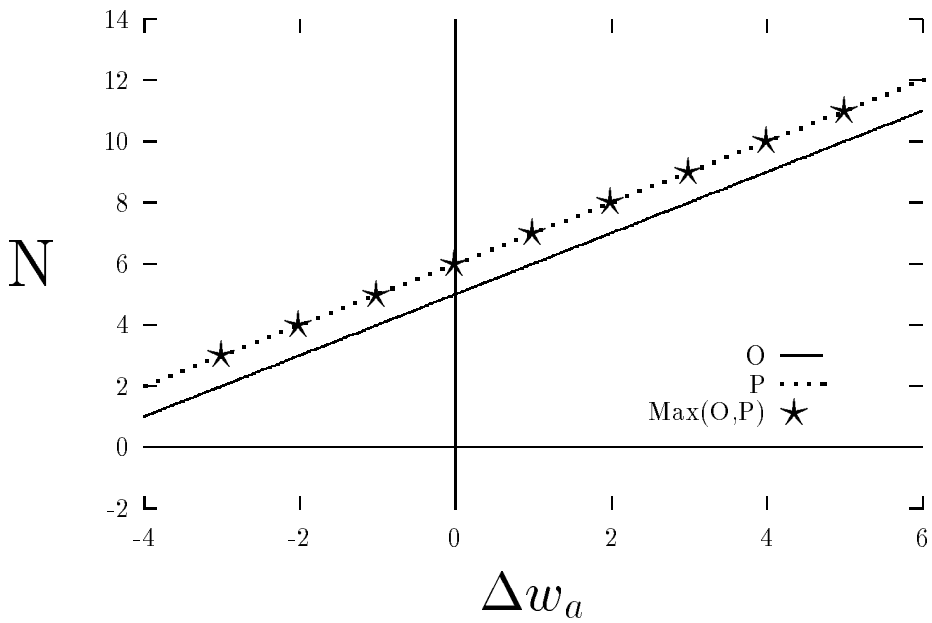

L
M
T
N
Q
U
X
O
P
R
V
W
Y
Z
5
6
7
3
1
4
2
acf
abg
adf
abd
bdf
bcg
bch

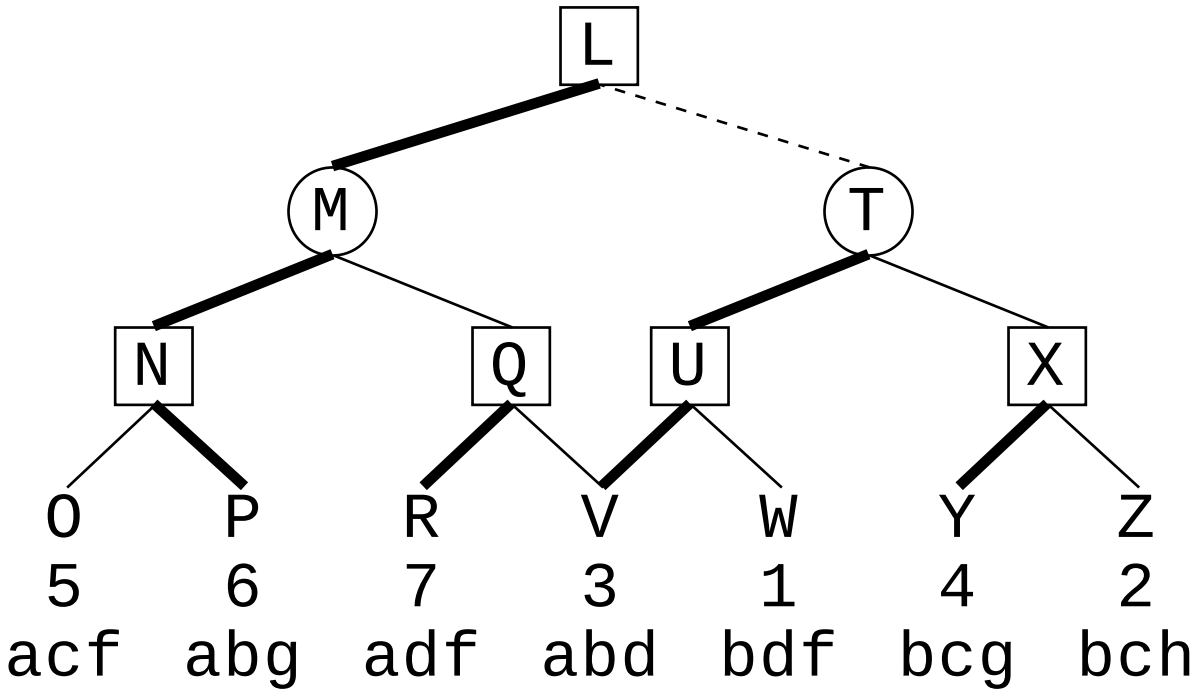

O
14
12
10
8
6
4
2
0
-2
-4
-2
0
2
4
6
$\Delta w_a, \Delta w_c, \Delta w_f$

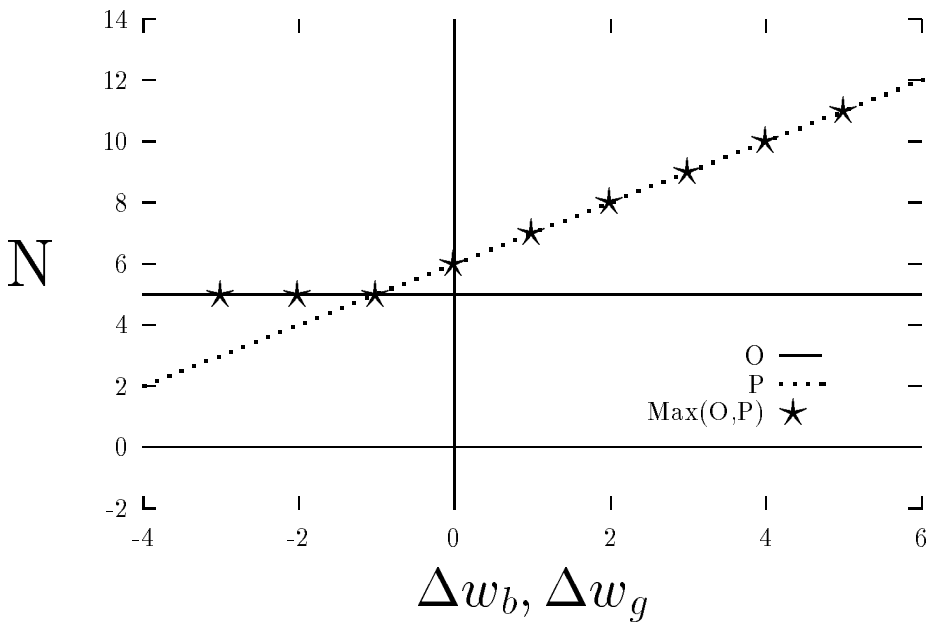

L
M
T
N
Q
U
X
O
P
R
V
W
Y
Z
5
6
7
3
1
4
2
acf
abg
adf
abd
bdf
bcg
bch

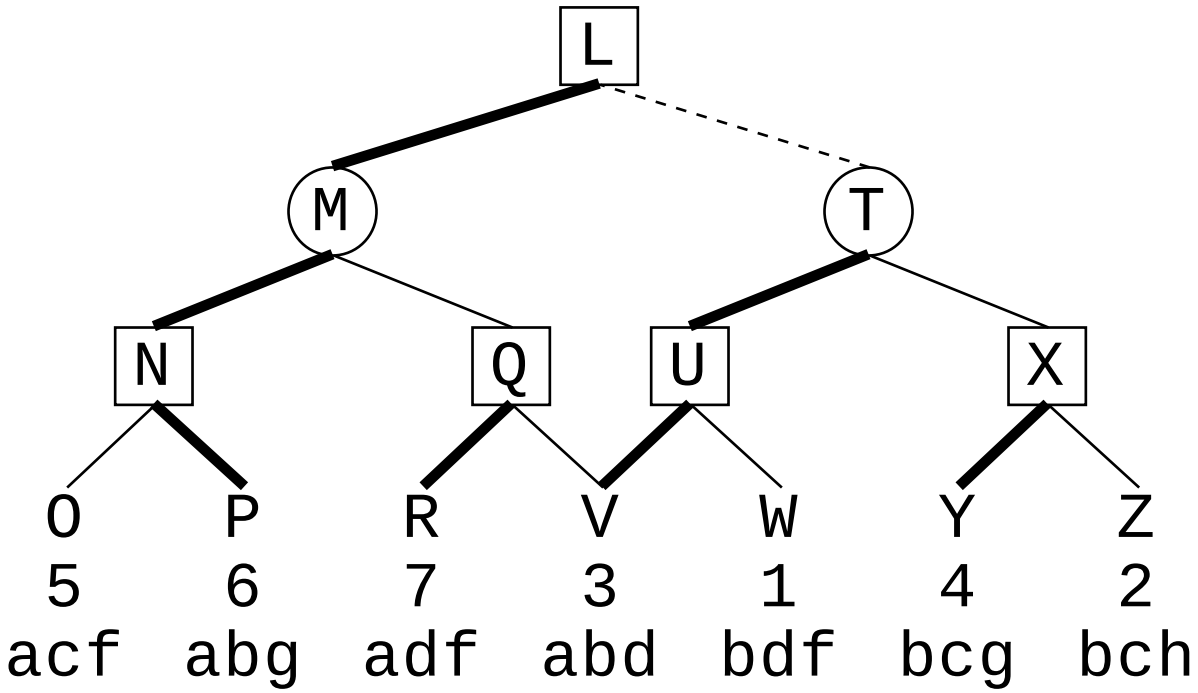

O
14
12
10
8
6
4
2
0
-2
-4
-2
0
2
4
6
$\Delta w_b, \Delta w_d, \Delta w_e, \Delta w_g, \Delta w_h$

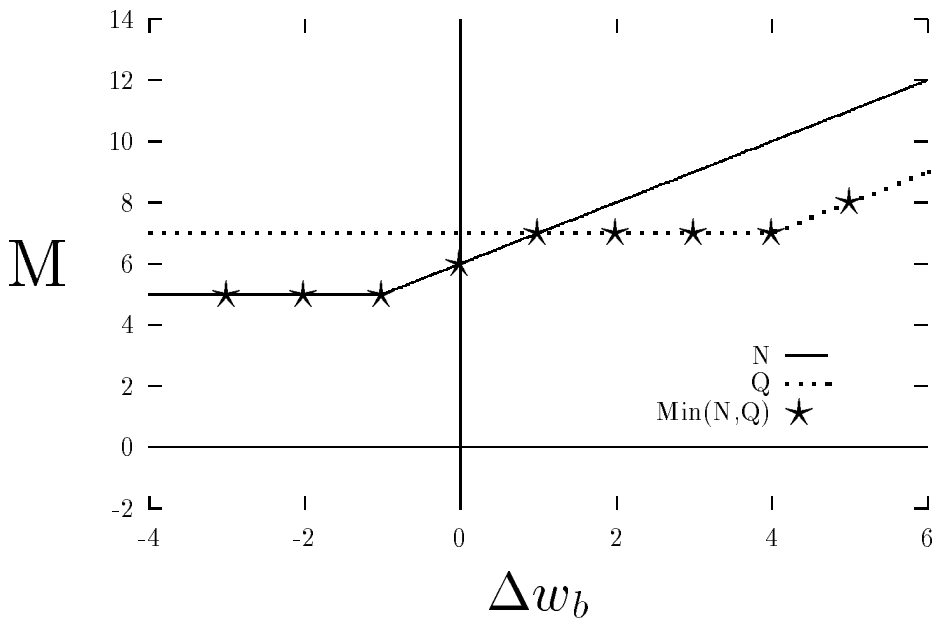

L
M
T
N
Q
U
X
O P R V W Y Z
5 6 7 3 1 4 2
acf abg adf abd bdf bcg bch

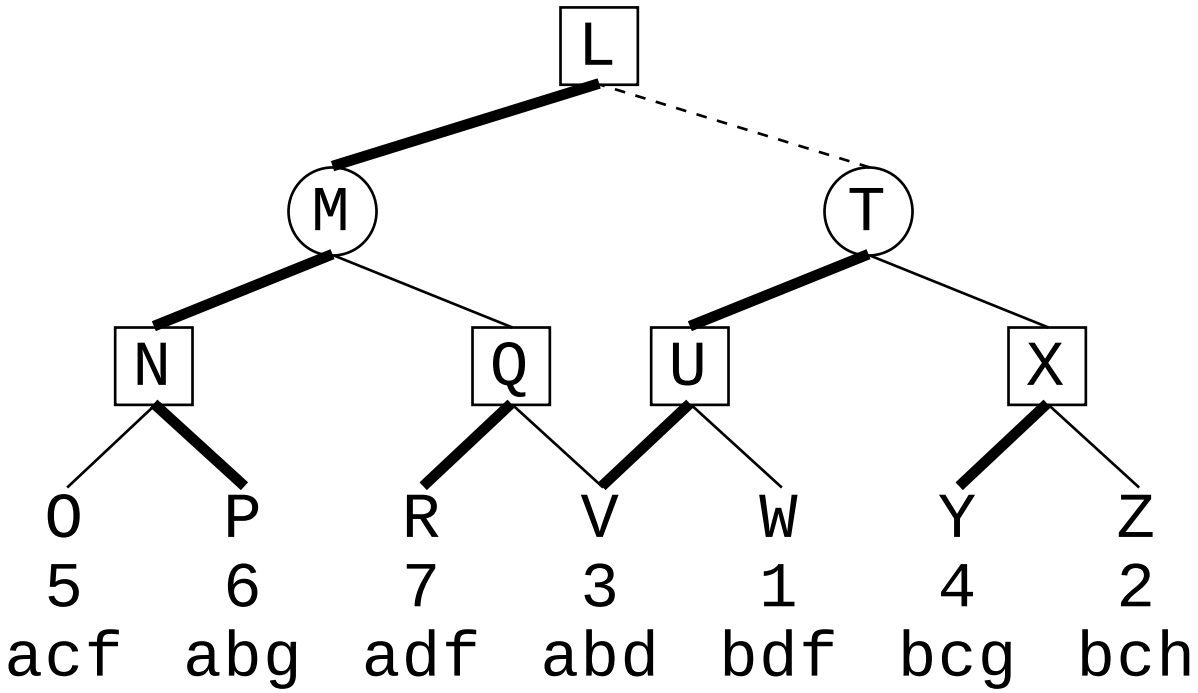

N
14
12
10
8
6
4
2
0
-2
-4
-2
0
2
4
6
O
P
Max(O,P)
Δw_a

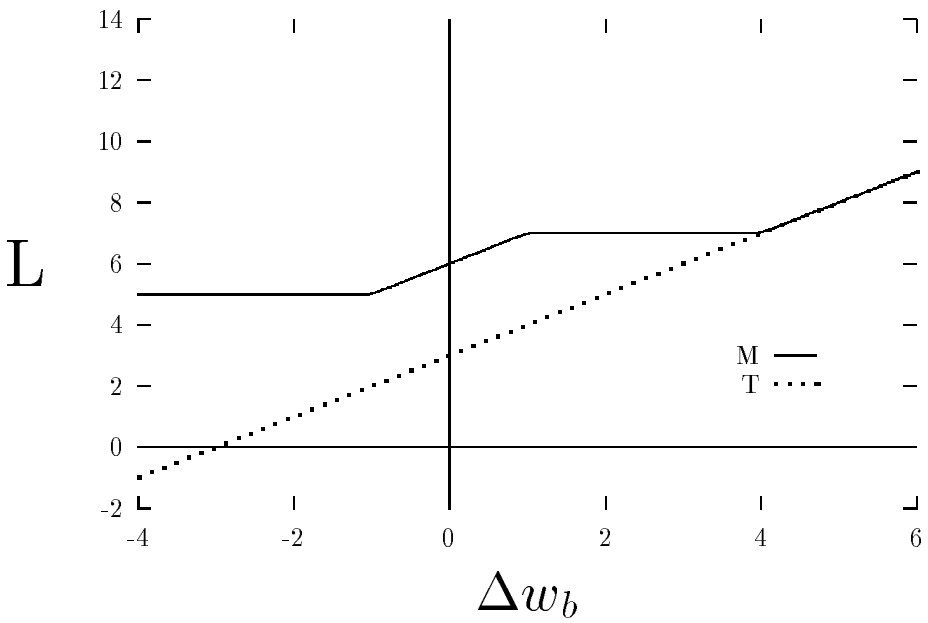

L
M
T
N
Q
U
X
O P R V W Y Z
5 6 7 3 1 4 2
acf abg adf abd bdf bcg bch

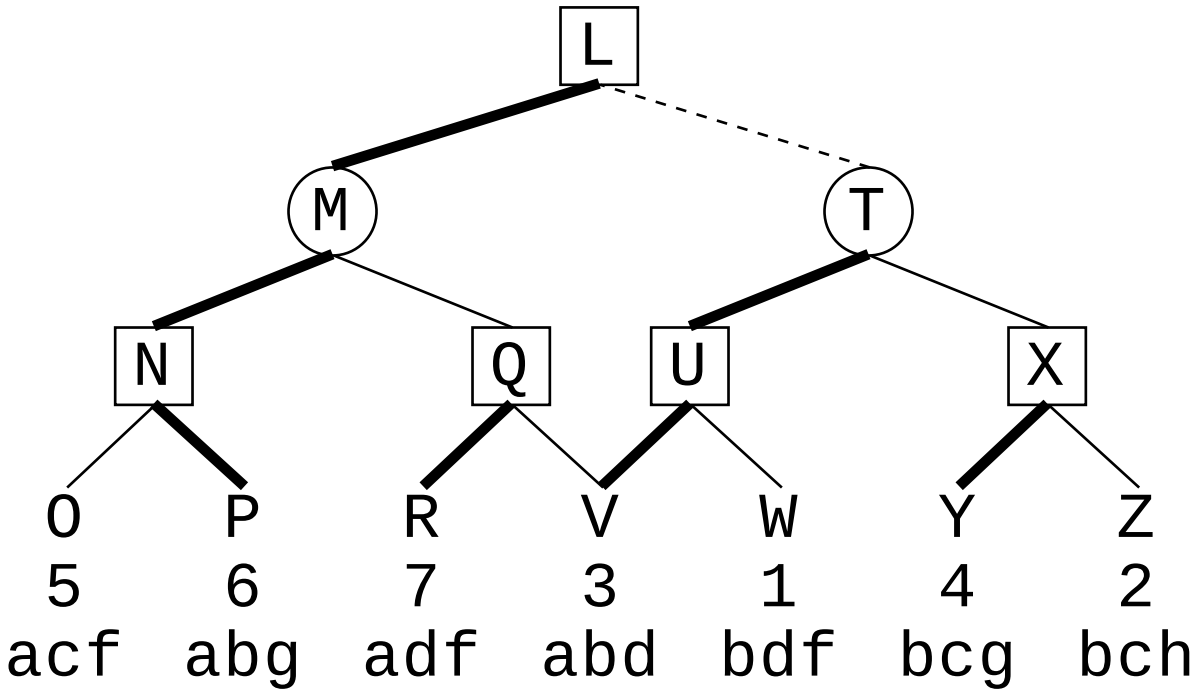

N
14
12
10
8
6
4
2
0
-2
-4
-2
0
2
4
6
O
P
Max(O,P)
$\Delta w_b, \Delta w_g$

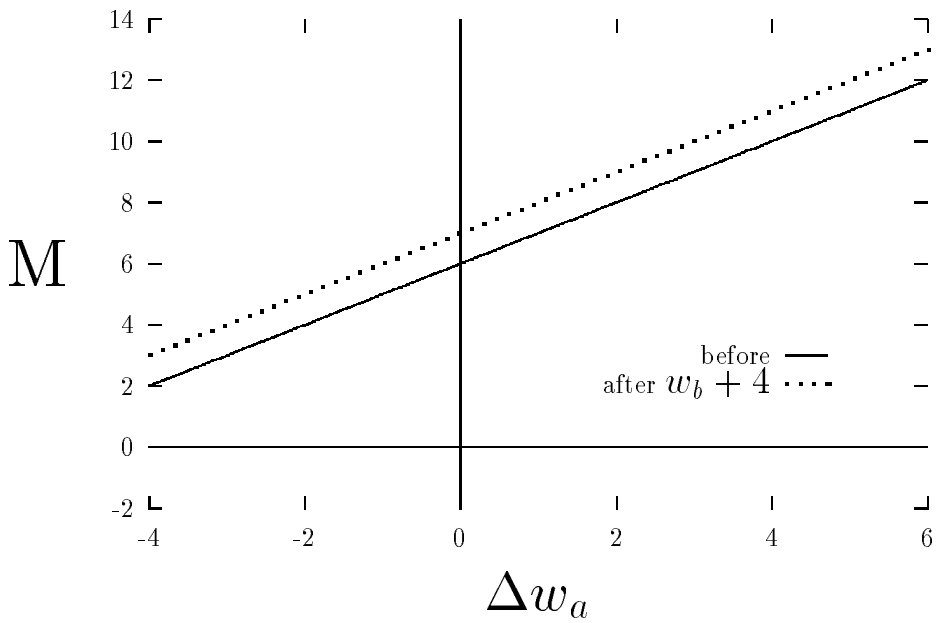

L
M
T
N
Q
U
X
O P R V W Y Z
5 6 7 3 1 4 2
acf abg adf abd bdf bcg bch

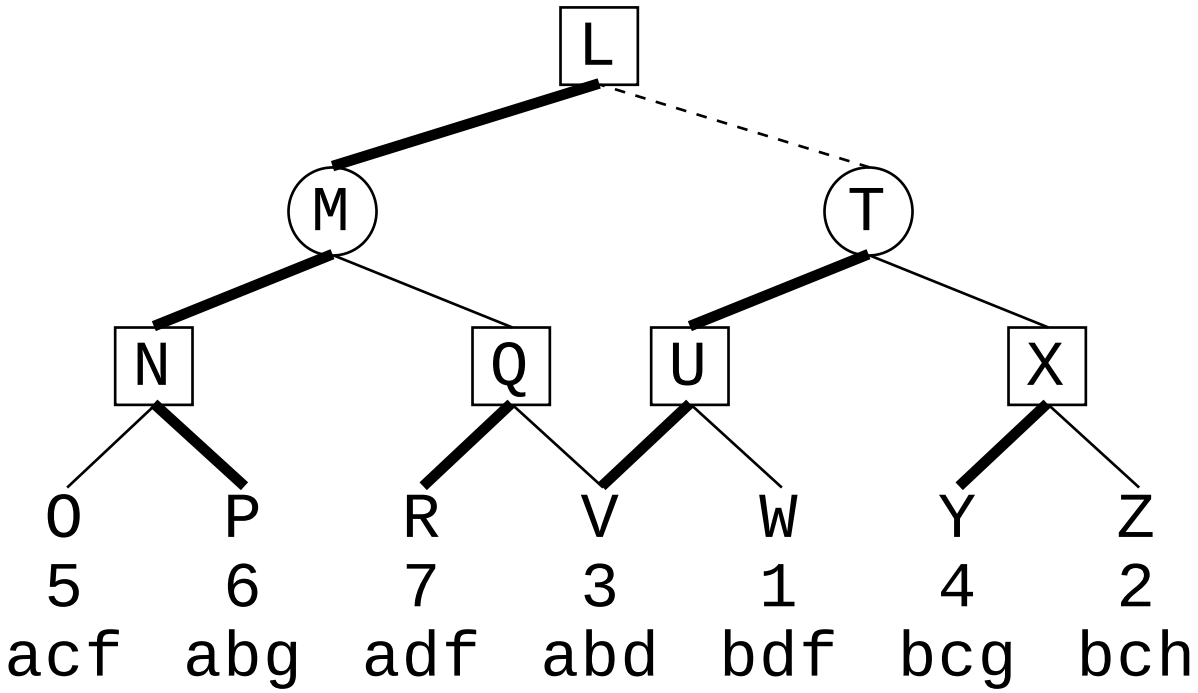

M
14
12
10
8
6
4
2
0
-2
-4 -2 0 2 4 6
N
Q
Min(N,Q)
Δw_b

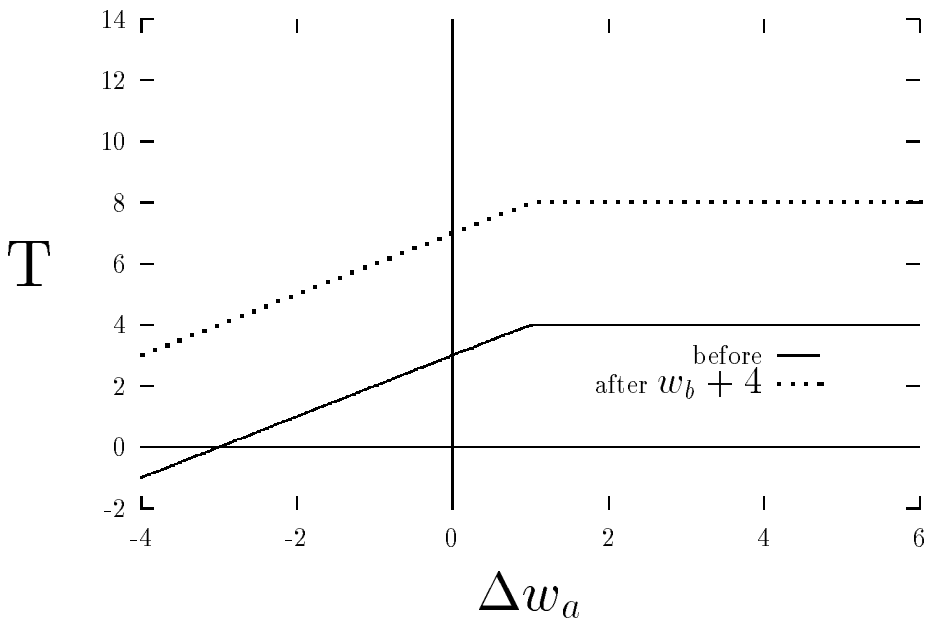

L
M
T
N
Q
U
X
O
P
R
V
W
Y
Z
5
6
7
3
1
4
2
acf
abg
adf
abd
bdf
bcg
bch

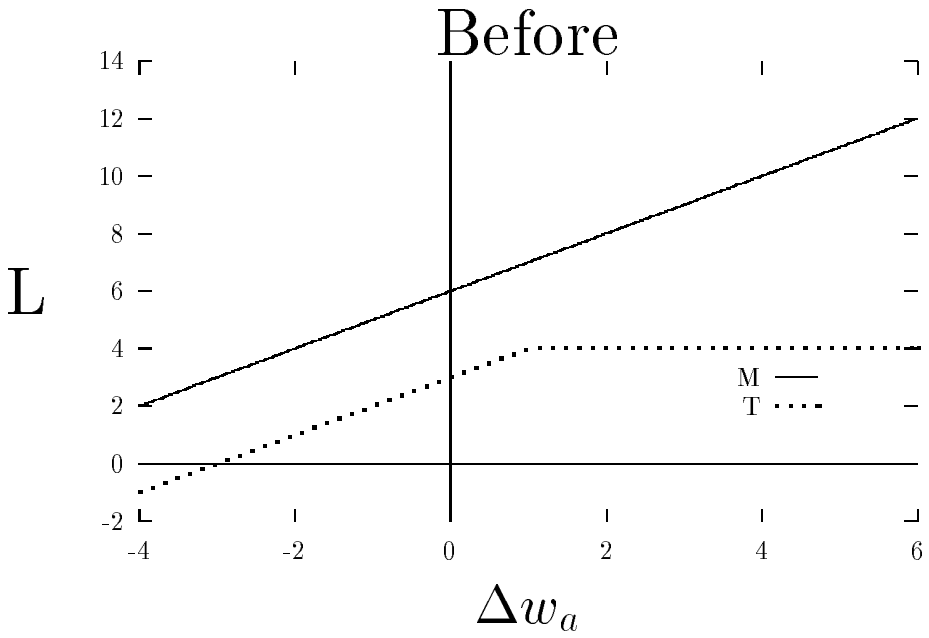

L
14
12
10
8
6
4
2
0
-2
-4
-2
0
2
4
6
M
T
Δw_b

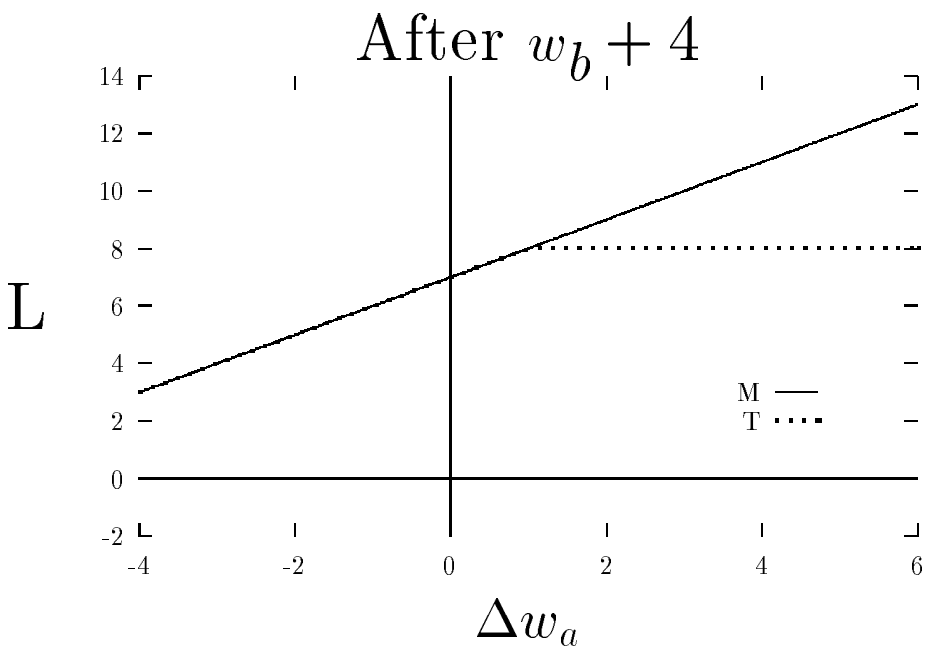

L
M
T
N
Q
U
X
O P R V W Y Z
5 6 7 3 1 4 2
acf abg adf abd bdf bcg bch


M
14
12
10
8
6
4
2
0
-2
-4
-2
0
2
4
6
before
after $w_b + 4$
Δw_a

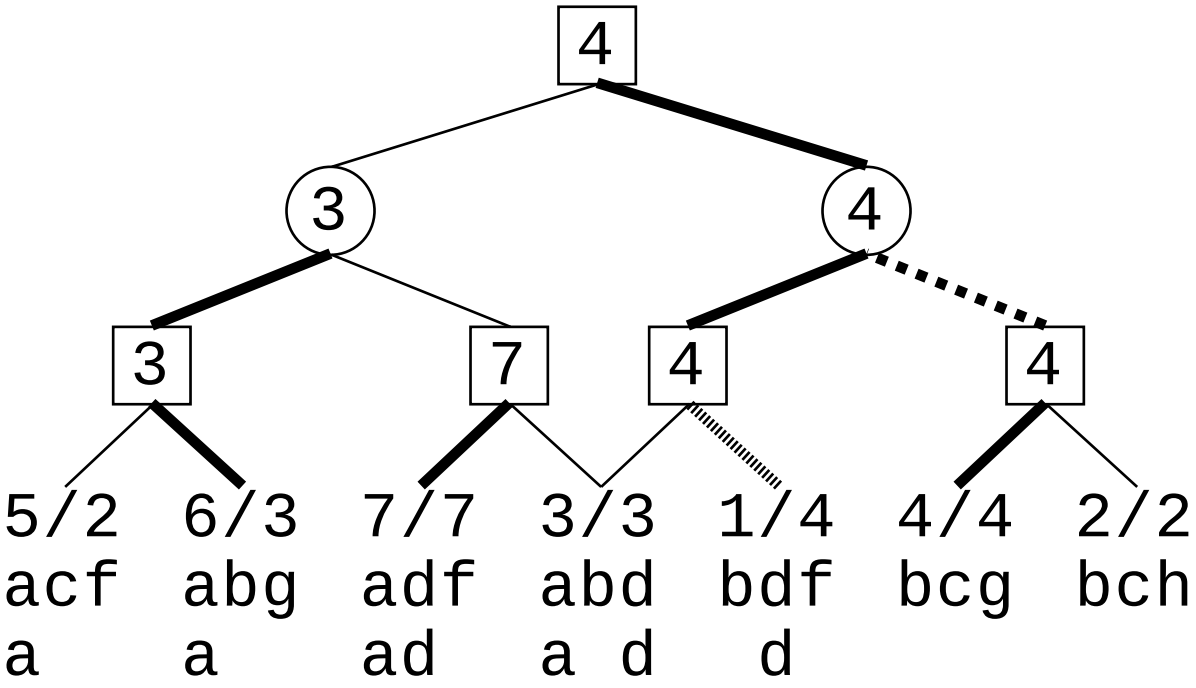

L
M
T
N
Q
U
X
O P R V W Y Z
5 6 7 3 1 4 2
acf abg adf abd bdf bcg bch


T
14
12
10
8
6
4
2
0
-2
-4 -2 0 2 4 6
before
after $w_b + 4$
Δw_a


Before
L
14
12
10
8
6
4
2
0
-2
-4
-2
0
2
4
6
M
T
Δw_a


After $w_b + 4$
L
14
12
10
8
6
4
2
0
-2
-4
-2
0
2
4
6
M
T
Δw_a

Final Solution



- $E(n) = 3a - 3b + c + 3d + f + 6g + 4h$
- $\Delta a = -3, \Delta d = 3$
- $E'(n) = 0a - 3b + c + 6d + f + 6g + 4h$
- $Cost = \mid \Delta a \mid + \mid \Delta d \mid = 6$

A Good Problem?

- Simple domain
  - Easy to control
  - Experts?
  - Quality of training data?
- Complex domain (Chess)
  - Hitech is fast
  - Lots of high quality data
  - Domain expert
  - Domain where lots of effort has been expended
  - Strategy is within context of material
  - There exist a few hand generated strategies
  - Lots more work!

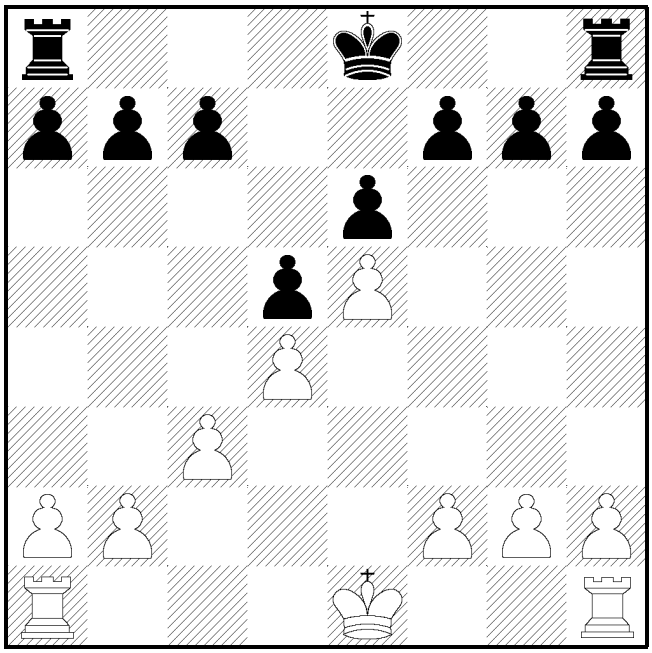
Hitech's Evaluation Function

- Recognizers $R(b)$ – nonlinear function of board
- Integer Piece/Square table $T(Piece/Square)$
- Pawn = 128
- $E(b) = Recognizers(b) + \Sigma_{p \in b} T(p)$
- Example Piece/Square table:

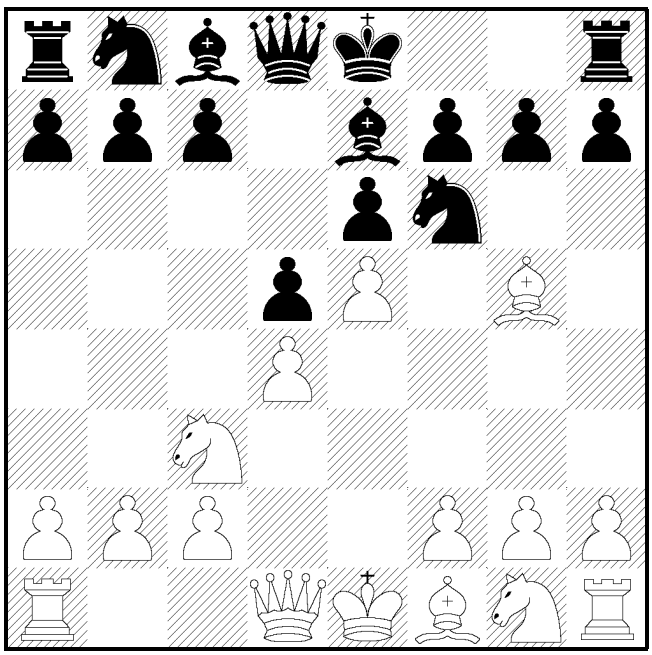
BP

8	128	128	128	128	128	128	128	128
7	128	130	132	134	134	132	130	128
6	128	130	132	140	140	132	130	128
5	128	130	137	148	148	137	130	128
4	131	133	142	156	156	142	133	131
3	134	137	146	164	164	146	137	134
2	216	222	232	248	248	232	222	216
1	0	0	0	0	0	0	0	0
	a	b	c	d	e	f	g	h

Pawntype



Sequence



Nd7 h4 Nc6 Nh3 f6

Training Signal

- $b = 35, d = 6, b^d = 10^{10}$
- Limit to $1/2$ pawn of best
- Initial constraints by best 2 $\alpha - \beta$
- Initial $b = 2, d = 6, b^d = 64$
- Final $b = 3, d = 6, b^d = 729$

Overall Picture

- Databases of human games
- Segment by *pawntype* into *sequences* of *positions*
- Learn *strategy* specific to one *pawntype*
- Random selection biased by frequency
- Initial trees found by best 2 $\alpha - \beta$
- Loop until no progress
  - Solve positions with subgradient
  - Check strategy against each position
  - Add unexpected lines
- Keep best solution for each pawntype
- Use in games

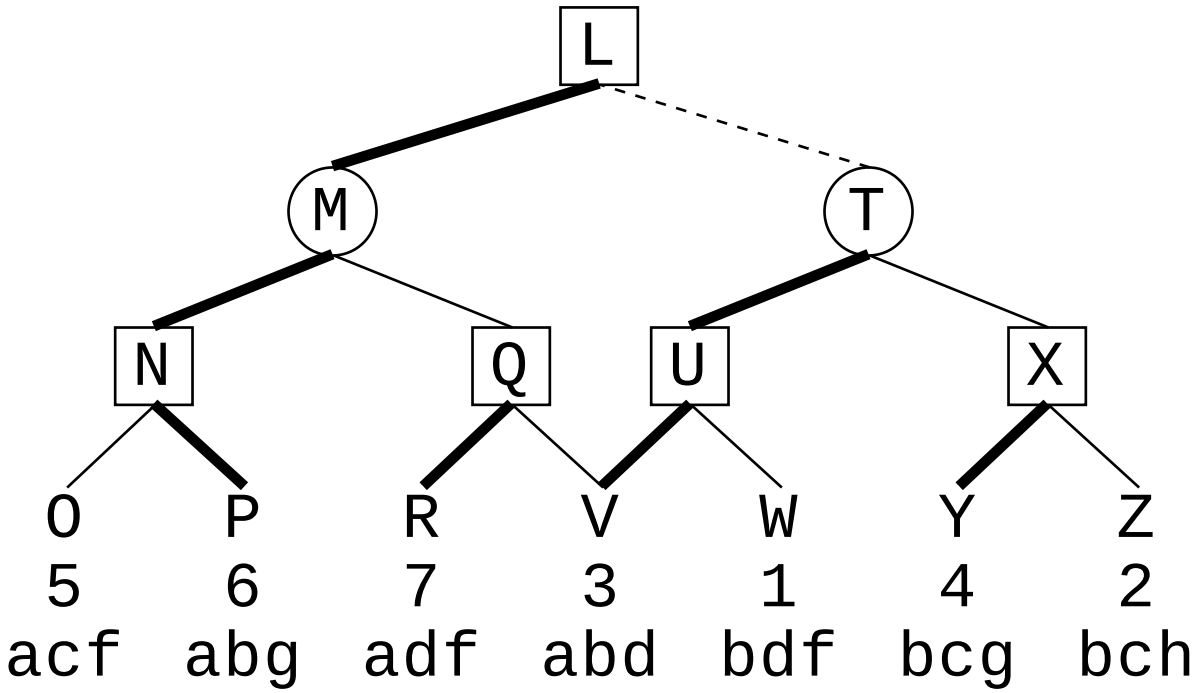
Sample Solution

- 170 sequences, 1524 positions in French defense
- 550 unique, random 189 → 163 positions
- Initial 232 vars, 3186 choices, 7727 leaves
- Right: 55 86 99 86 107 113
- Final 245 vars, 5198 choices, 13726 leaves
- 1 hour, cost = 172, max = 15
- Validation 63/173 → 97/173

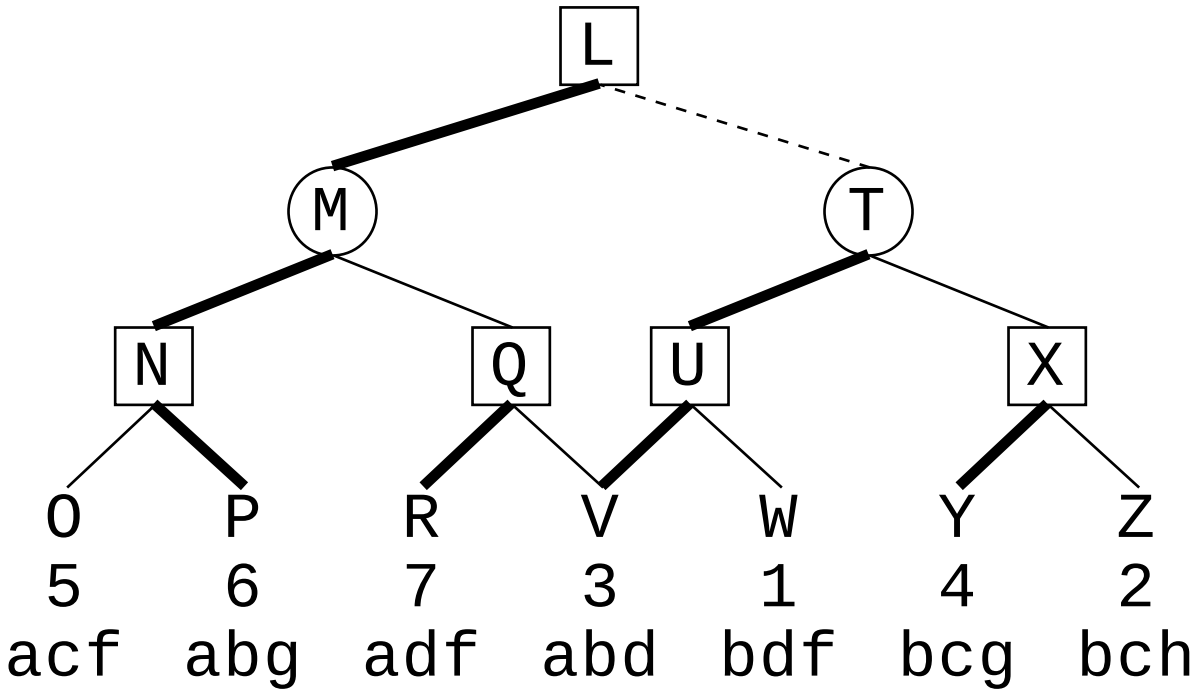
```
BB
   ~    ~    ~    ~    ~    ~    ~    ~
   ~    ~    ~    1    ~    ~    ~    ~
-15     ~    ~    ~    ~    ~    ~    ~
   ~    ~    ~    ~    ~    ~    ~    ~
 -5     ~   -2    ~    ~    ~    ~    ~
   ~    ~    ~    ~    ~    ~    ~    ~
   ~    ~    ~    ~    ~    ~    ~    ~
   ~    ~    ~    ~    ~    ~    ~    ~
```

Summary

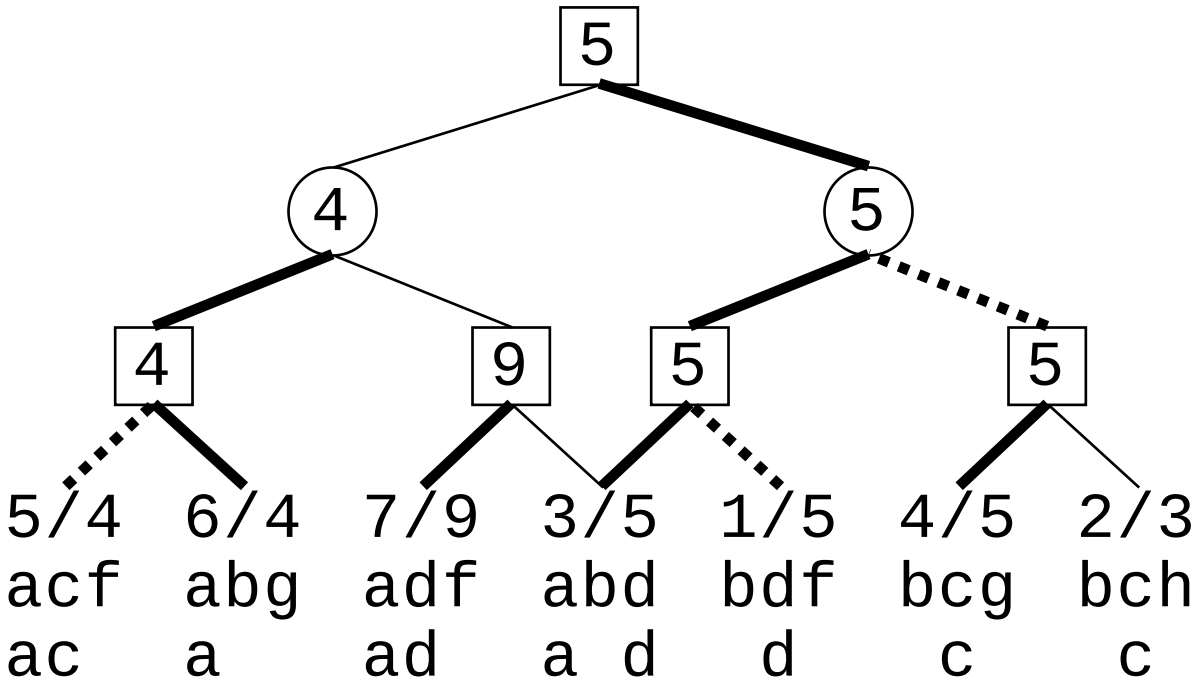
- Local
- Limited data
- Deep
- Can solve positions
- Reasonable by domain expert
- Generalization erratic
  - Quality and consistency of data
  - Margin of error
  - Base variability
  - Interference
  - Second order effects



$$O = \Delta a + \Delta c + \Delta f + 5$$
$$P = \Delta a + \Delta b + \Delta g + 6$$
$$N = P, N \geq O \rightarrow P \geq O$$
$$\Delta a + \Delta b + \Delta g + 6 \geq \Delta a + \Delta c + \Delta f + 5$$
$$\Delta b + \Delta g - \Delta c - \Delta f \geq -1$$



$$T \geq M + 1$$
$$T \geq M + 1 - \epsilon$$
$$\Delta a + \Delta b + \Delta d + 3 \geq \Delta a + \Delta b + \Delta g + 6 + 1 - \epsilon$$
$$\Delta d - \Delta g + \epsilon \geq 4$$



$$Minimize :| \Delta a \mid + \mid \Delta b \mid + \mid \Delta c \mid + \mid \Delta d \mid + \mid \Delta f \mid + \mid \Delta g \mid + \mid \Delta h \mid + \lambda\epsilon$$

$$\Delta d - \Delta g + \epsilon >= 4$$

$$\Delta b - \Delta c - \Delta f + \Delta g >= -1$$

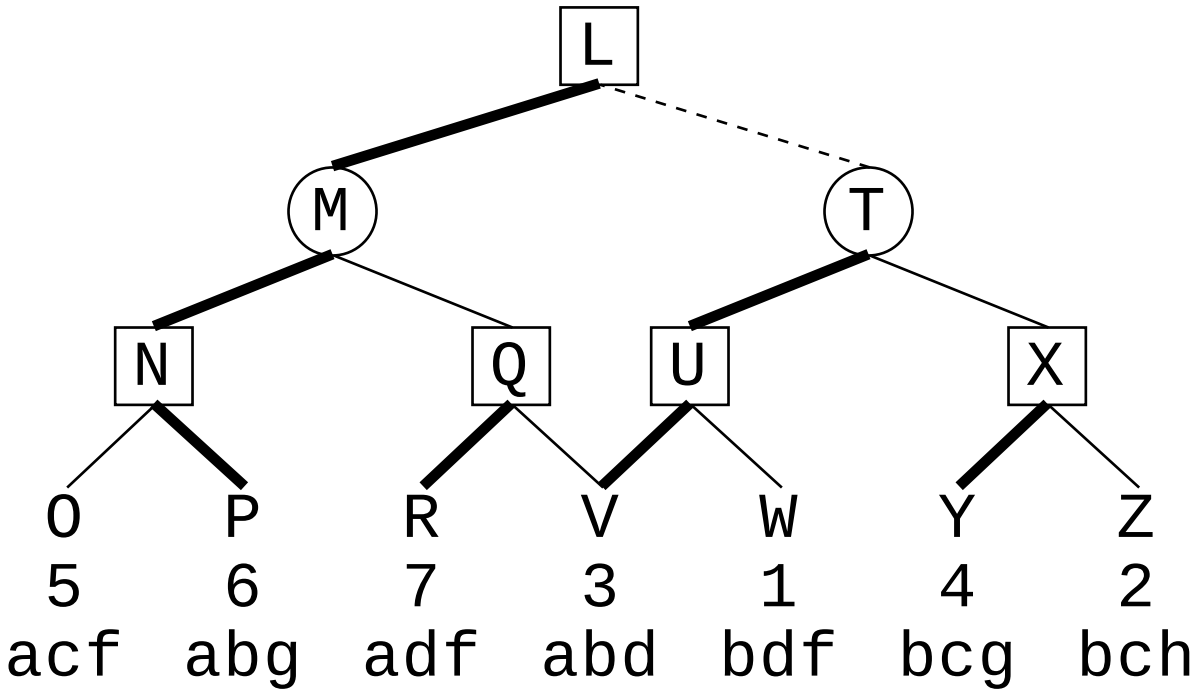
$$-\Delta b + \Delta f >= -4$$

$$\Delta a - \Delta f >= -2$$

$$\Delta g - \Delta h >= -2$$

$$\Delta b - \Delta d - \Delta f + \Delta g <= 1$$

$$\Delta a - \Delta c + \Delta d - \Delta g <= 1$$



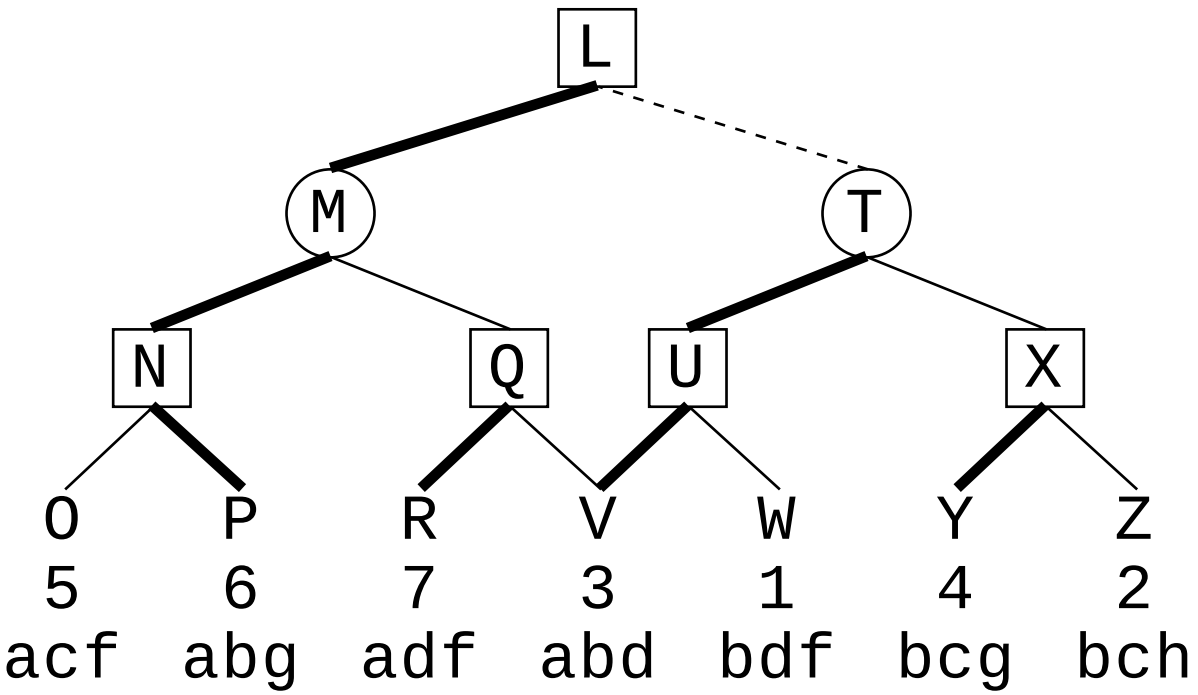
$$T = U \Rightarrow T = X$$

- $U \leq X \rightarrow V \leq Y \Rightarrow X \leq U \rightarrow Y \leq V$
  $\Delta a + \Delta d - \Delta c - \Delta g <= 1 \rightarrow$
  $-\Delta a - \Delta d + \Delta c + \Delta g <= -1$
- $T \geq M + 1 - \epsilon \rightarrow V \geq P + 1 - \epsilon \Rightarrow Y \geq P + 1 - \epsilon$
  $\Delta d - \Delta g + \epsilon >= 4 \rightarrow$
  $\Delta c - \Delta a + \epsilon >= 3$

λ and ϵ

- ϵ is error in goal constraint
- λ is penalty
  - Total Δ allowed / error
  - Example $\lambda > 2$
  - Tradeoff bewteen problems
  - G = gap that covers 80% of problems
  - $\lambda = 1.5G$ to force Δ overlap

Relaxed Problem



$$\Delta d - \Delta g + \epsilon >= 4$$

Multiple Problems

- Reduce problems to top level constraints
- Add one problem's constraints
- Solve constraints
- Use solution to choose in unexpanded problems
- Add all problem constraints
- Solve and keep best
- Repeat
- Throw out all never right problems
- Local search

Nonadversarial Search

- Applicable when:
  - $E(b) = \Sigma_i \, c_i F_i$
  - Way of defining locality
- λ is only combinatorial problem