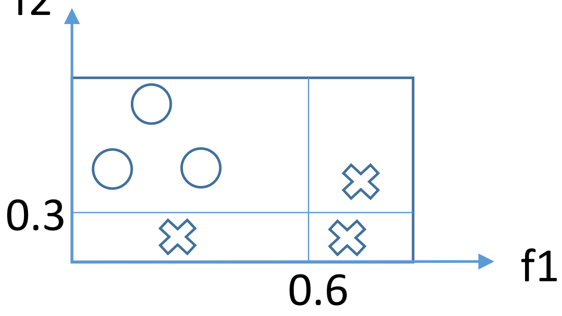
λ=0.1

F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1



λ=0.1

Initialization:

Root problem p [1,1,1,1,1,1]

Q = {[1,1,1,1,1,1]}

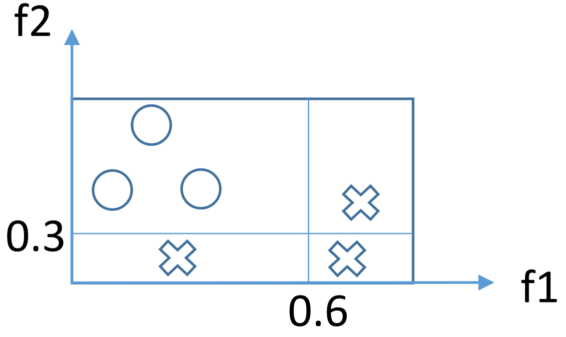
G

p.id=[1,1,1,1,1,1]

p.ub=3/6 + λ

p.lb = 2λ

F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1



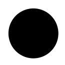
λ=0.1

Initialization:

Root problem p [1,1,1,1,1,1]

Q = {[1,1,1,1,1,1]}

G

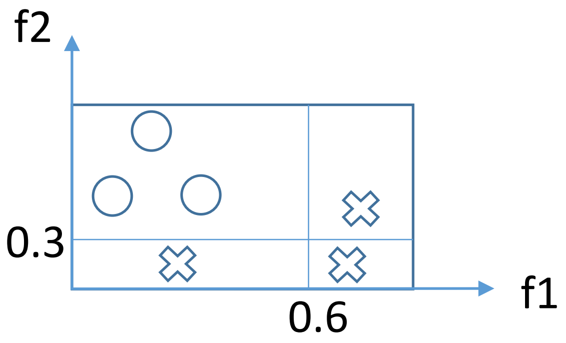


p.id=[1,1,1,1,1,1]

p.ub=3/6 + λ

p.lb = 2λ

F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1



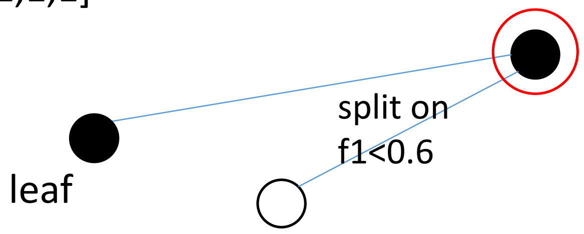
1[st] iteration

s=[1,1,1,1,1,1]

p.id=[1,1,1,1,1,1]

Q={}

G



p.id=[1,1,1,1,1,1]

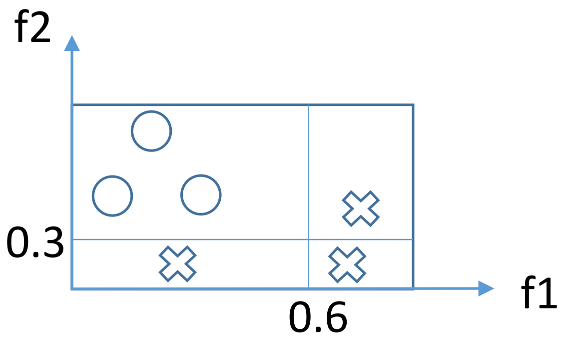
p.ub=3/6 + λ

p.lb = 2λ

λ=0.1
Initialization:
Root problem p [1,1,1,1,1,1]
Q = {[1,1,1,1,1,1]}
G

p.id=[1,1,1,1,1,1]
p.ub=3/6 + λ
p.lb = 2λ

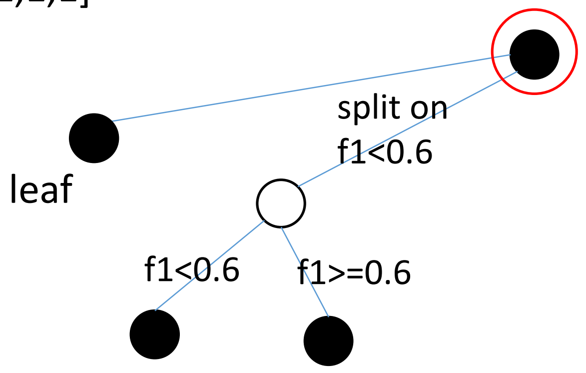
F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1



1^{st} iteration
s=[1,1,1,1,1,1]
p.id=[1,1,1,1,1,1]
Q={}

G

p.id=[1,1,1,1,1,1]
p.ub=3/6 + λ
p.lb = 2λ



[1,1,1,1,0,0]
p.ub=1/6+λ
p.lb=2λ

[0,0,0,0,1,1]
p.ub=λ
p.lb=2λ

λ=0.1

Initialization:

Root problem p [1,1,1,1,1,1]

Q = {[1,1,1,1,1,1]}

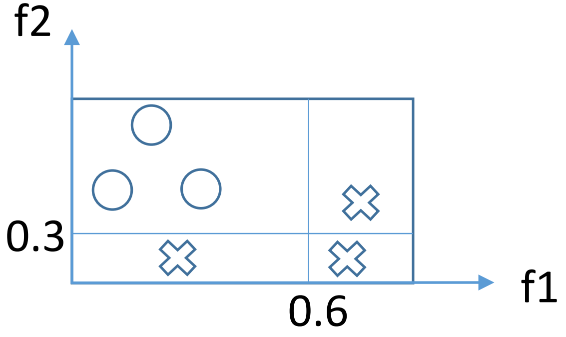
G

p.id=[1,1,1,1,1,1]

p.ub=3/6 + λ

p.lb = 2λ

F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1



1st iteration

s=[1,1,1,1,1,1]

p.id=[1,1,1,1,1,1]

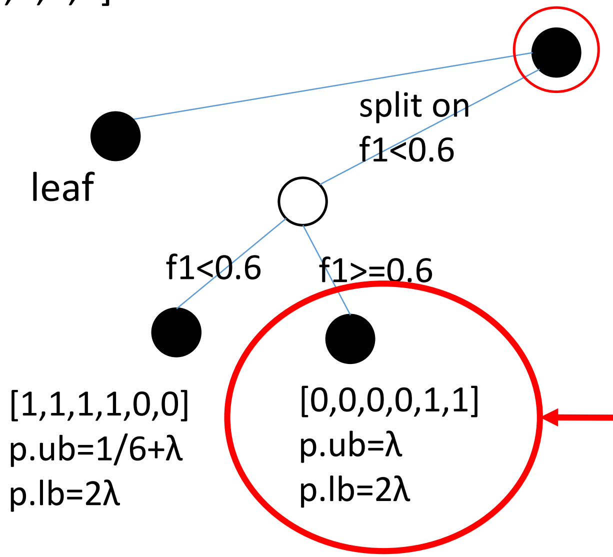
Q={}

G

p.id=[1,1,1,1,1,1]

p.ub=3/6 + λ

p.lb = 2λ



Since the initialized lower bound is already larger than initialized upper bound,

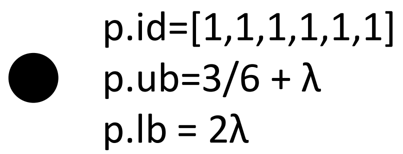
λ=0.1

Initialization:

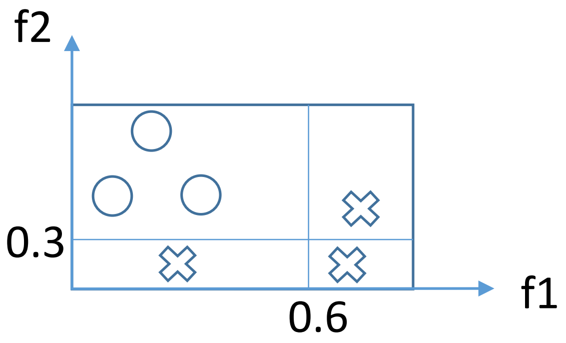
Root problem p [1,1,1,1,1,1]

Q = {[1,1,1,1,1,1]}

G



F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1

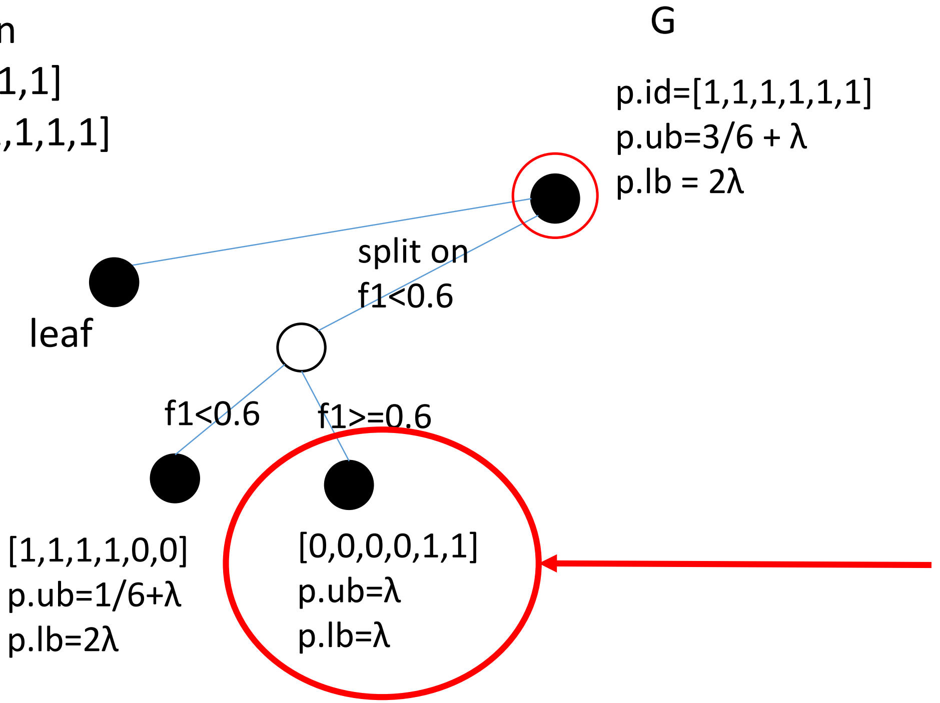


1[st] iteration

s=[1,1,1,1,1,1]

p.id=[1,1,1,1,1,1]

Q={}

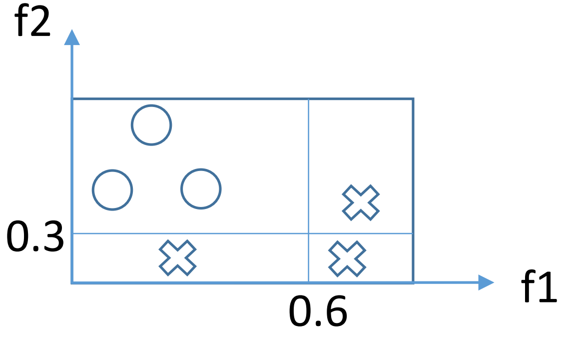


Since the initialized lower bound is already larger than initialized upper bound, we need to set the lower bound equal to the upper bound

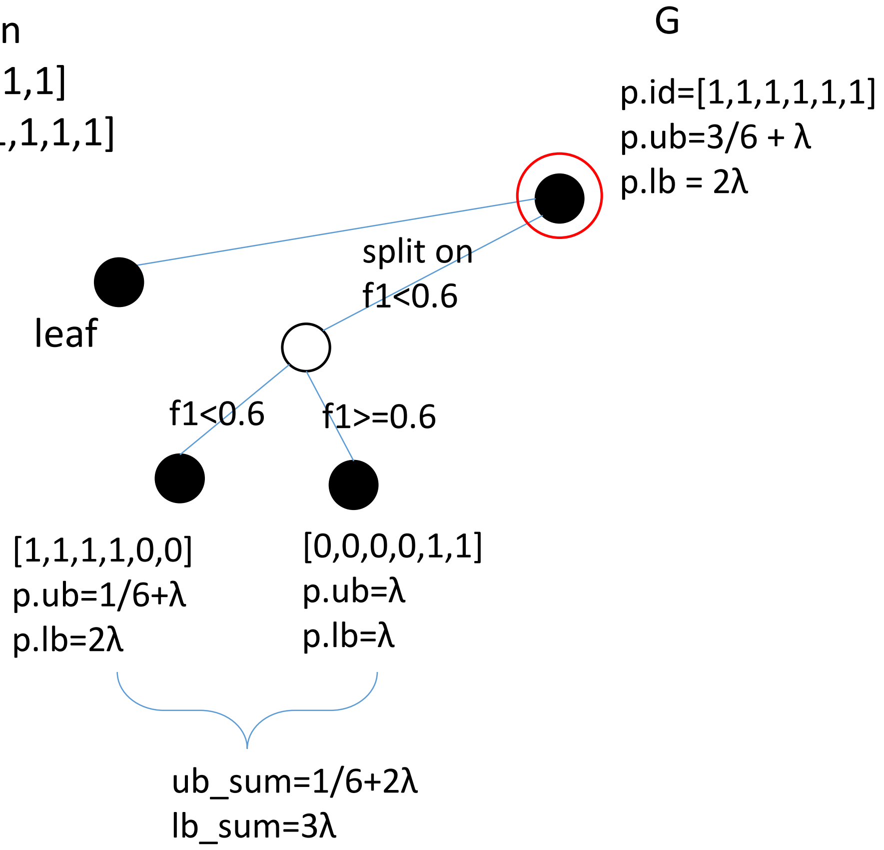
λ=0.1
Initialization:
Root problem p [1,1,1,1,1,1]
Q = {[1,1,1,1,1,1]}
G

p.id=[1,1,1,1,1,1]
p.ub=3/6 + λ
p.lb = 2λ

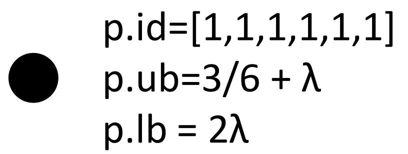
F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1



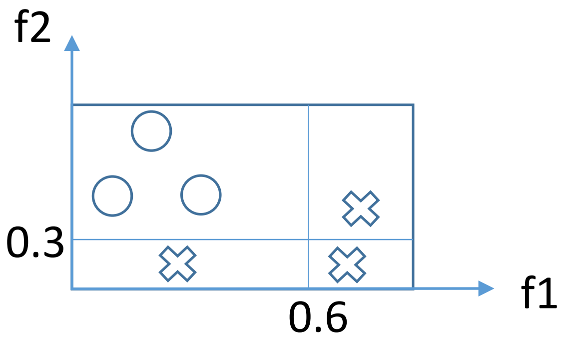
1st iteration
s=[1,1,1,1,1,1]
p.id=[1,1,1,1,1,1]
Q={}



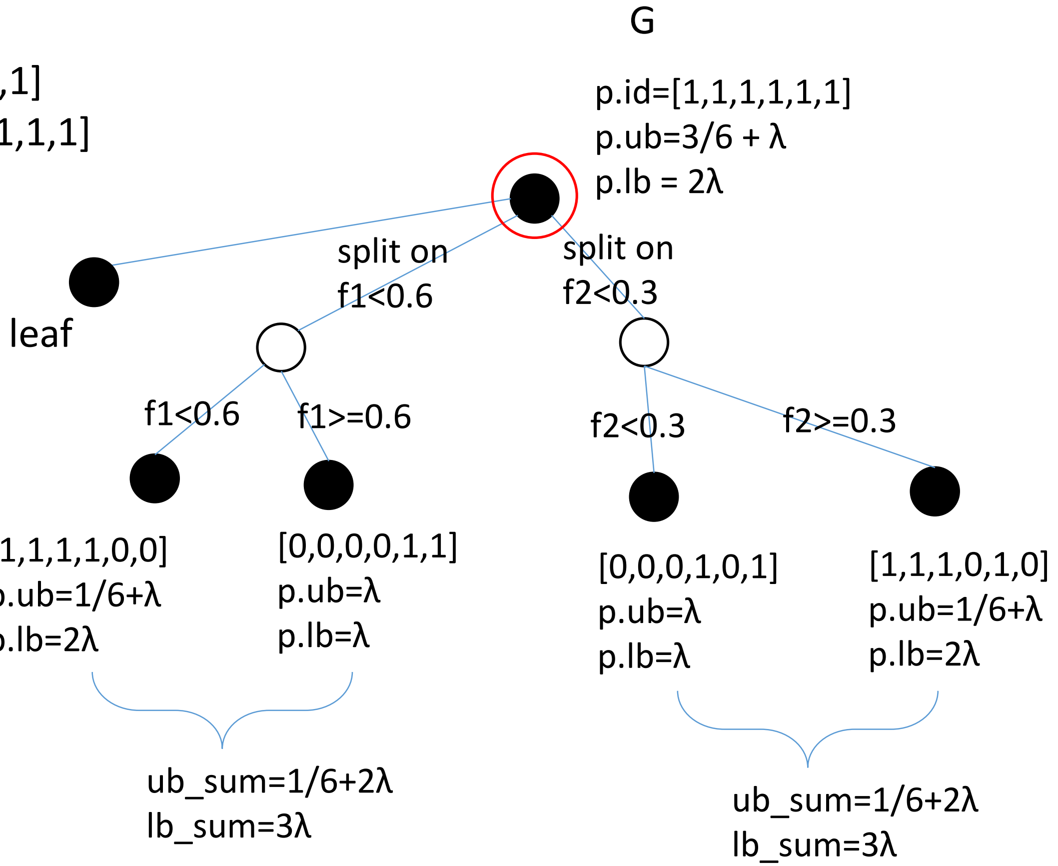
λ=0.1
Initialization:
Root problem p [1,1,1,1,1,1]
Q = {[1,1,1,1,1,1]}
G



F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1



1st iteration
s=[1,1,1,1,1,1]
p.id=[1,1,1,1,1,1]
Q={}



λ=0.1

Initialization:

Root problem p [1,1,1,1,1,1]

Q = {[1,1,1,1,1,1]}

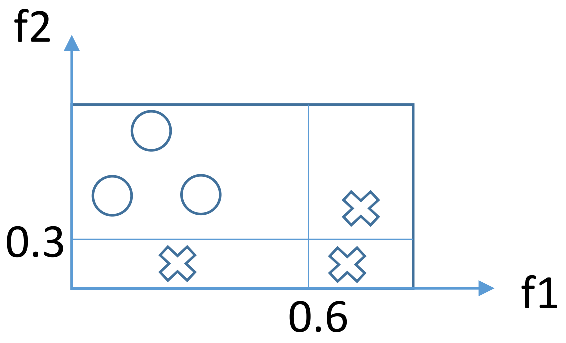
G

p.id=[1,1,1,1,1,1]

p.ub=3/6 + λ

p.lb = 2λ

F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1

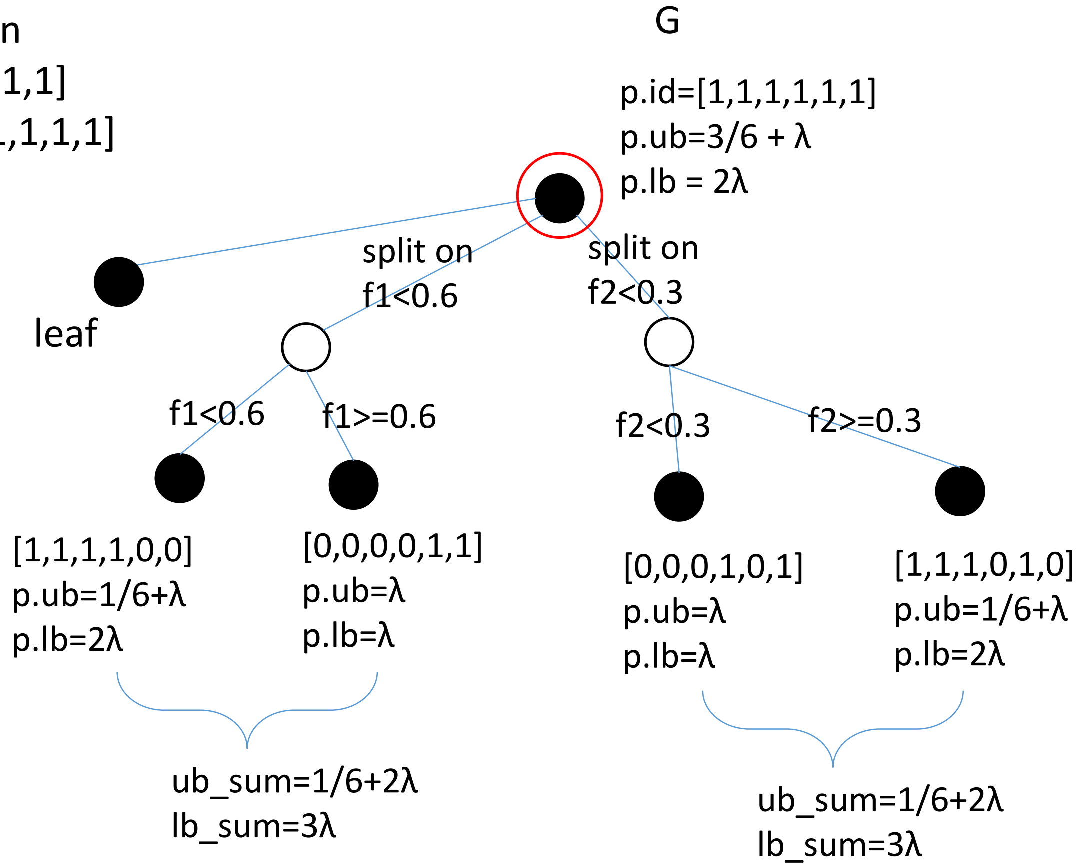


1st iteration

s=[1,1,1,1,1,1]

p.id=[1,1,1,1,1,1]

Q={}



Update the bound of p

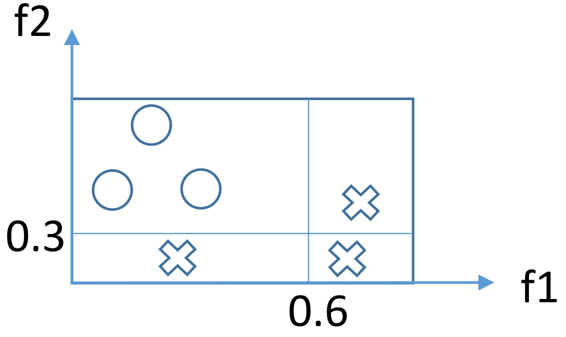
p.ub ← 1/6+2λ

p.lb ← 3λ

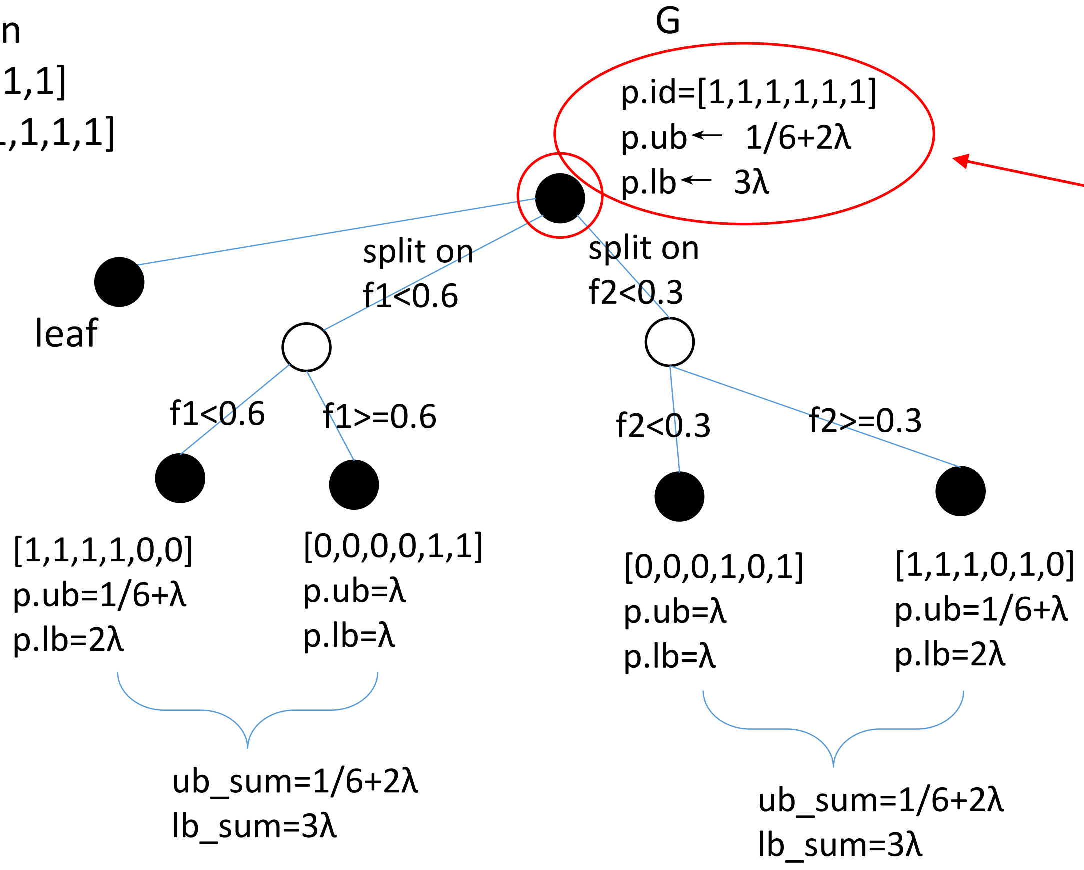
λ=0.1
Initialization:
Root problem p [1,1,1,1,1,1]
Q = {[1,1,1,1,1,1]}
G

p.id=[1,1,1,1,1,1]
p.ub=3/6 + λ
p.lb = 2λ

F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1



1st iteration
s=[1,1,1,1,1,1]
p.id=[1,1,1,1,1,1]
Q={}



Update the bound of p
p.ub← 1/6+2λ
p.lb← 3λ

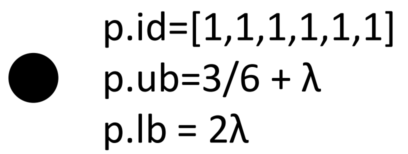
λ=0.1

Initialization:

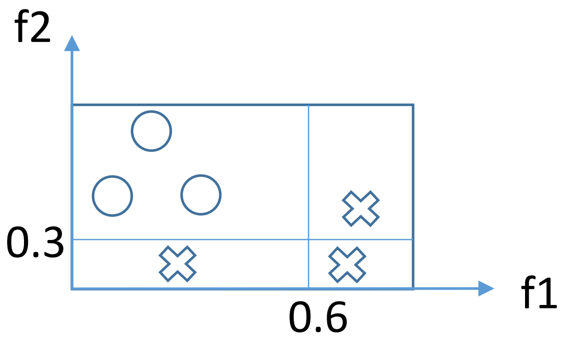
Root problem p [1,1,1,1,1,1]

Q = {[1,1,1,1,1,1]}

G



F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1

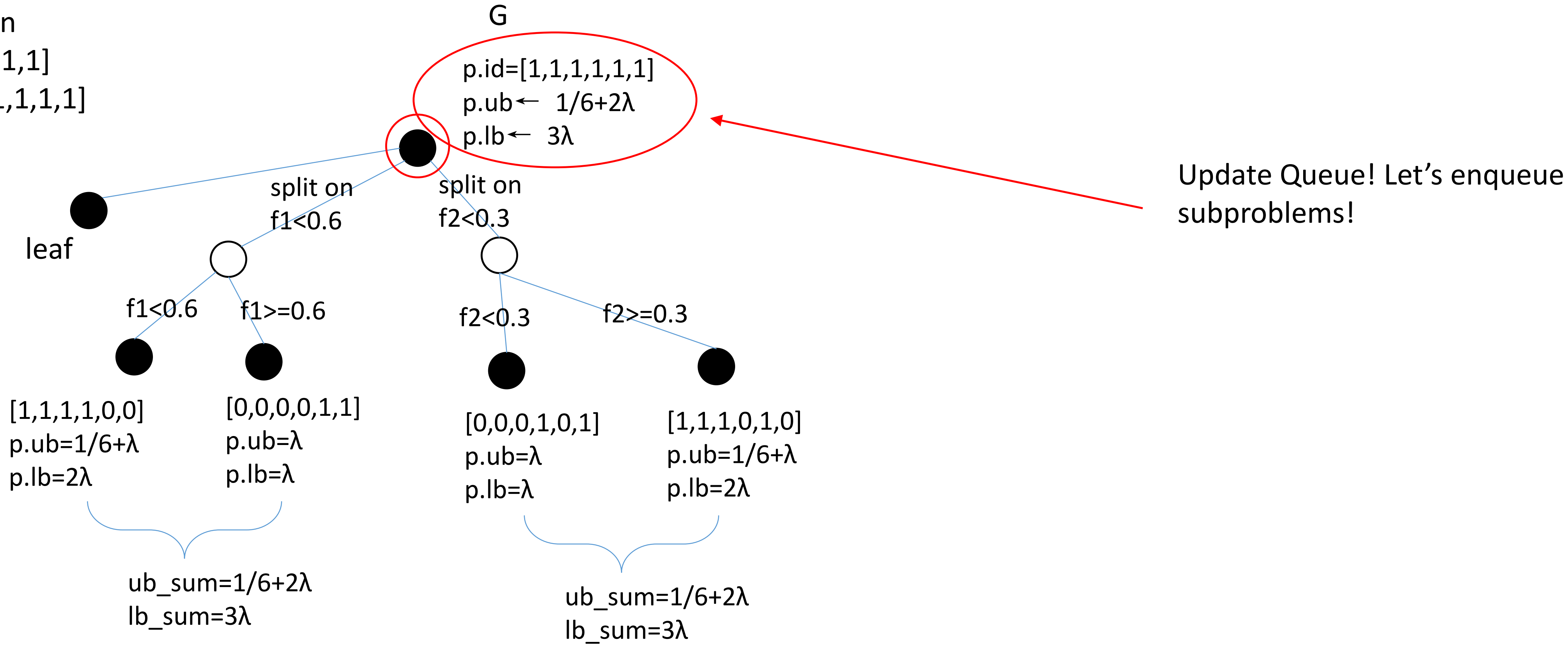


1st iteration

s=[1,1,1,1,1,1]

p.id=[1,1,1,1,1,1]

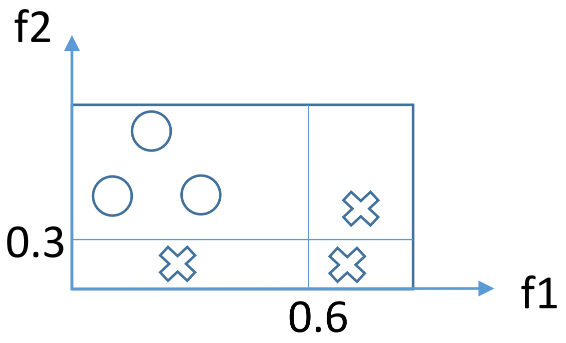
Q={}



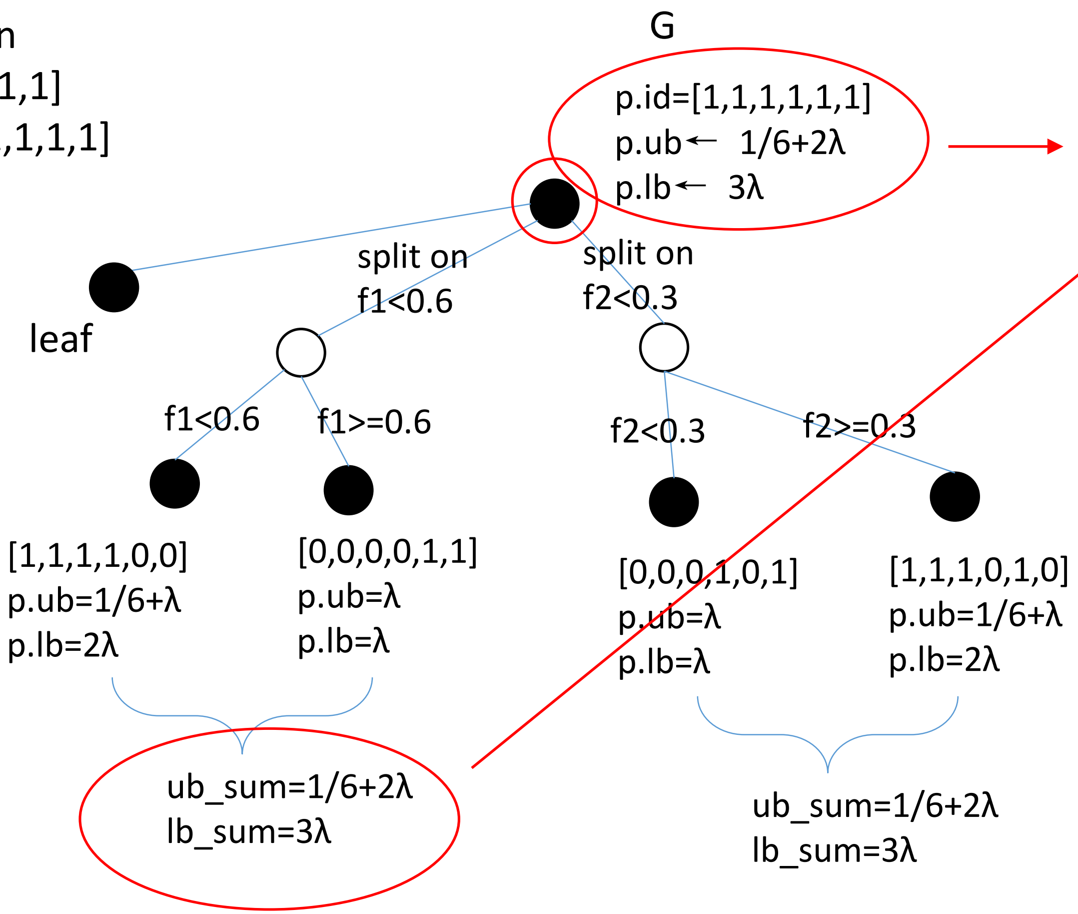
λ=0.1
Initialization:
Root problem p [1,1,1,1,1,1]
Q = {[1,1,1,1,1,1]}
G

p.id=[1,1,1,1,1,1]
p.ub=3/6 + λ
p.lb = 2λ

F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1



1st iteration
s=[1,1,1,1,1,1]
p.id=[1,1,1,1,1,1]
Q={}



lb_sum < ub_sum
and lb_sum <= p.ub?

Update Queue! Let's enqueue subproblems!

Conditions are satisfied. Enqueue this pair of subproblems.

Q = {[1,1,1,1,0,0], [0,0,0,0,1,1]}

λ=0.1

Initialization:

Root problem p [1,1,1,1,1,1]

Q = {[1,1,1,1,1,1]}

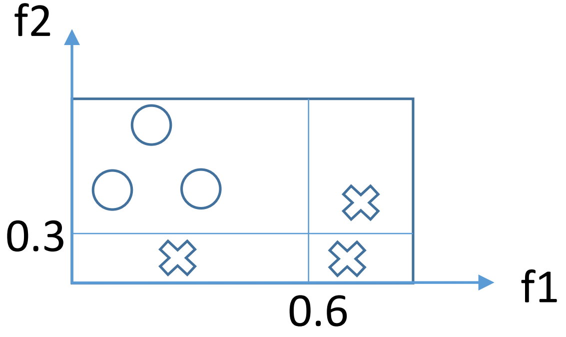
G

p.id=[1,1,1,1,1,1]

p.ub=3/6 + λ

p.lb = 2λ

F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1

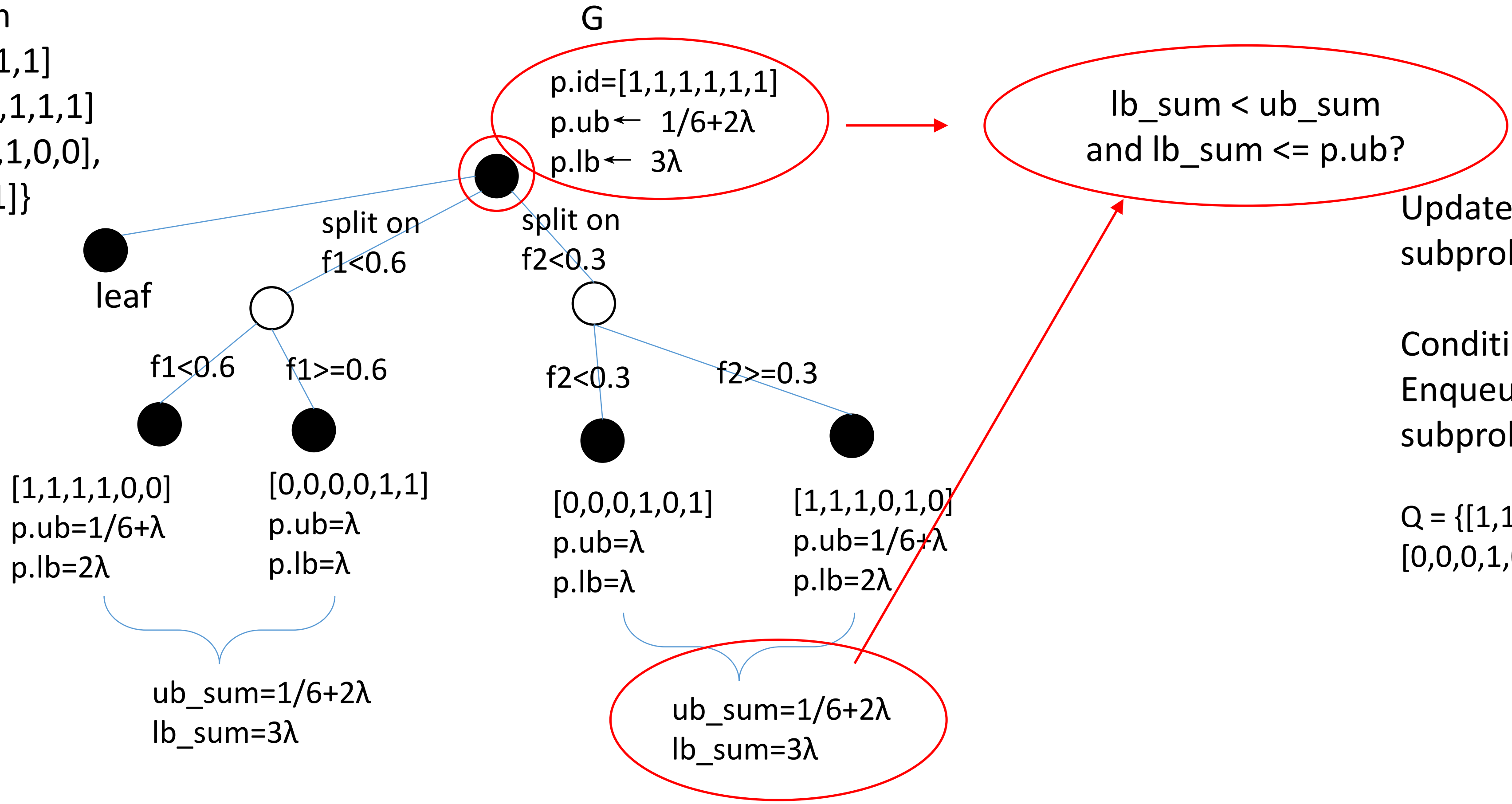


1st iteration

s=[1,1,1,1,1,1]

p.id=[1,1,1,1,1,1]

Q = {[1,1,1,1,0,0],
[0,0,0,0,1,1]}



Update Queue! Let's enqueue subproblems!

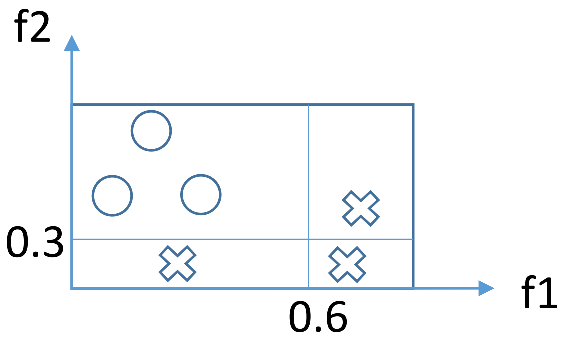
Conditions are satisfied. Enqueue this pair of subproblems.

Q = {[1,1,1,1,0,0], [0,0,0,0,1,1],
[0,0,0,1,0,1], [1,1,1,0,1,0]}

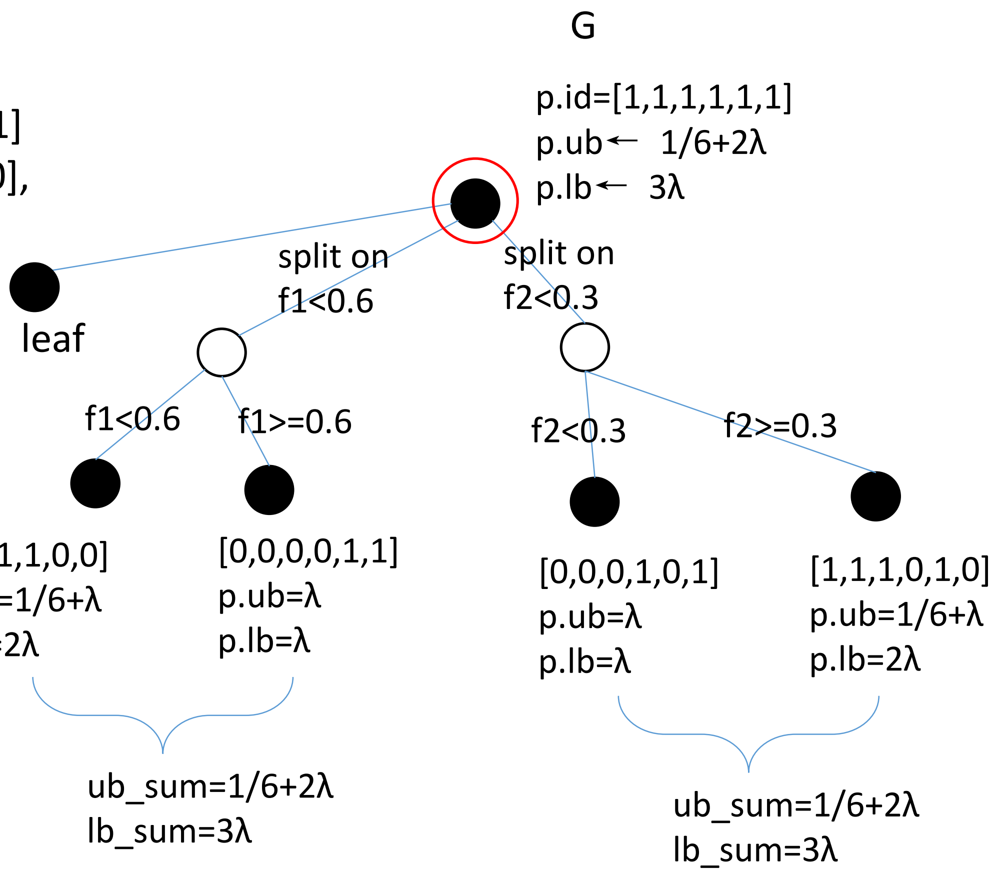
λ=0.1
Initialization:
Root problem p [1,1,1,1,1,1]
Q = {[1,1,1,1,1,1]}
G

p.id=[1,1,1,1,1,1]
p.ub=3/6 + λ
p.lb = 2λ

F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1



1[st] iteration
s=[1,1,1,1,1,1]
p.id=[1,1,1,1,1,1]
Q = {[1,1,1,1,0,0],
[0,0,0,0,1,1],
[0,0,0,1,0,1],
[1,1,1,0,1,0]}



Update Queue! Let's enqueue subproblems!

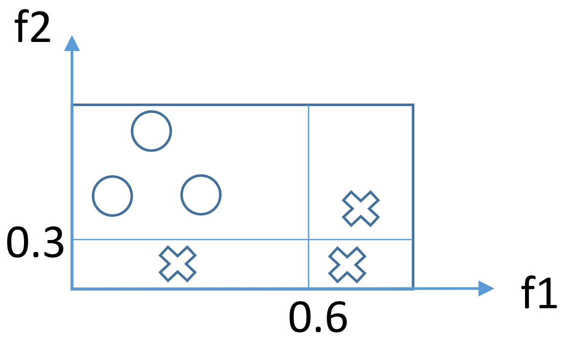
Q = {[1,1,1,1,0,0], [0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}

Jump back to check the while loop condition.

λ=0.1
Q = {[1,1,1,1,0,0], [0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}

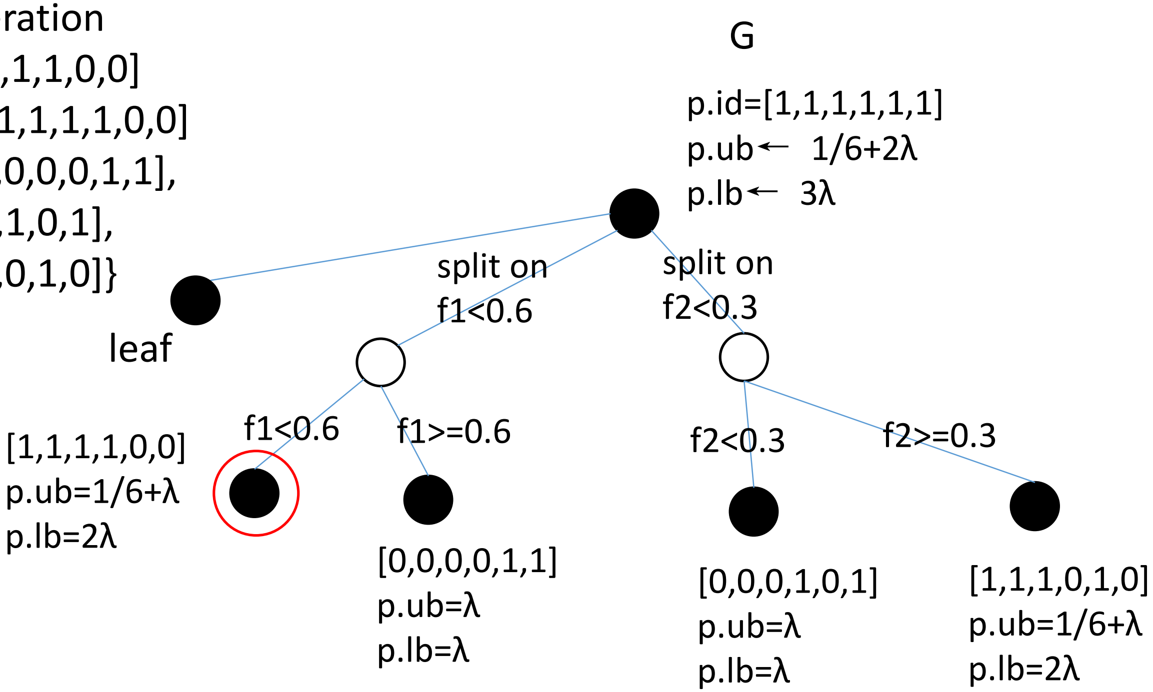
2nd iteration
s=[1,1,1,1,0,0]
p.id=[1,1,1,1,0,0]
Q={[0,0,0,0,1,1],
[0,0,0,1,0,1],
[1,1,1,0,1,0]}

F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1

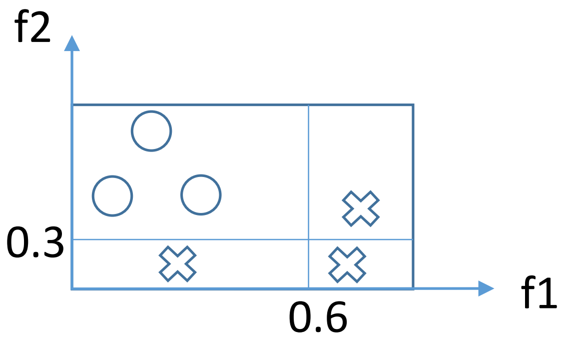


λ=0.1
Q = {[1,1,1,1,0,0], [0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}

2nd iteration
s=[1,1,1,1,0,0]
p.id=[1,1,1,1,0,0]
Q={[0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}



F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1



λ=0.1

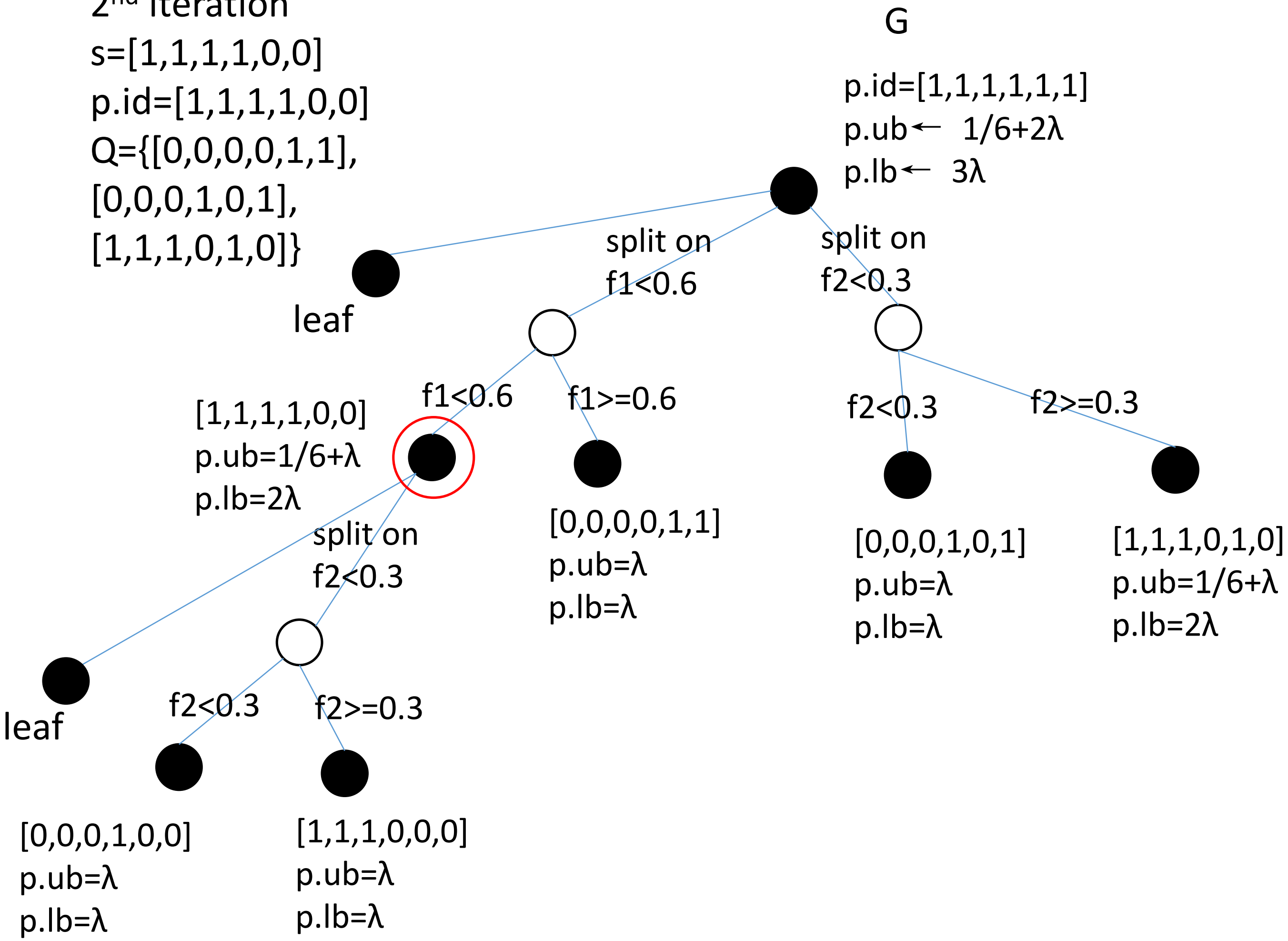
Q = {[1,1,1,1,0,0], [0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}

2^{nd} iteration

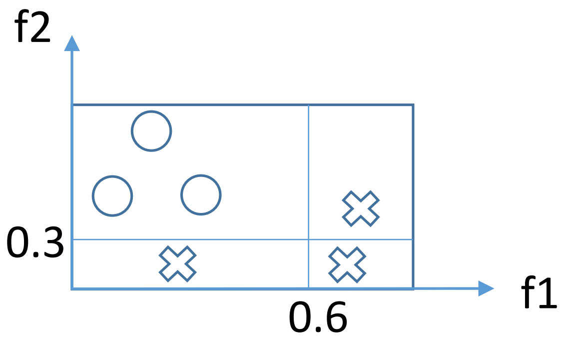
s=[1,1,1,1,0,0]

p.id=[1,1,1,1,0,0]

Q={[0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}



F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1



λ=0.1

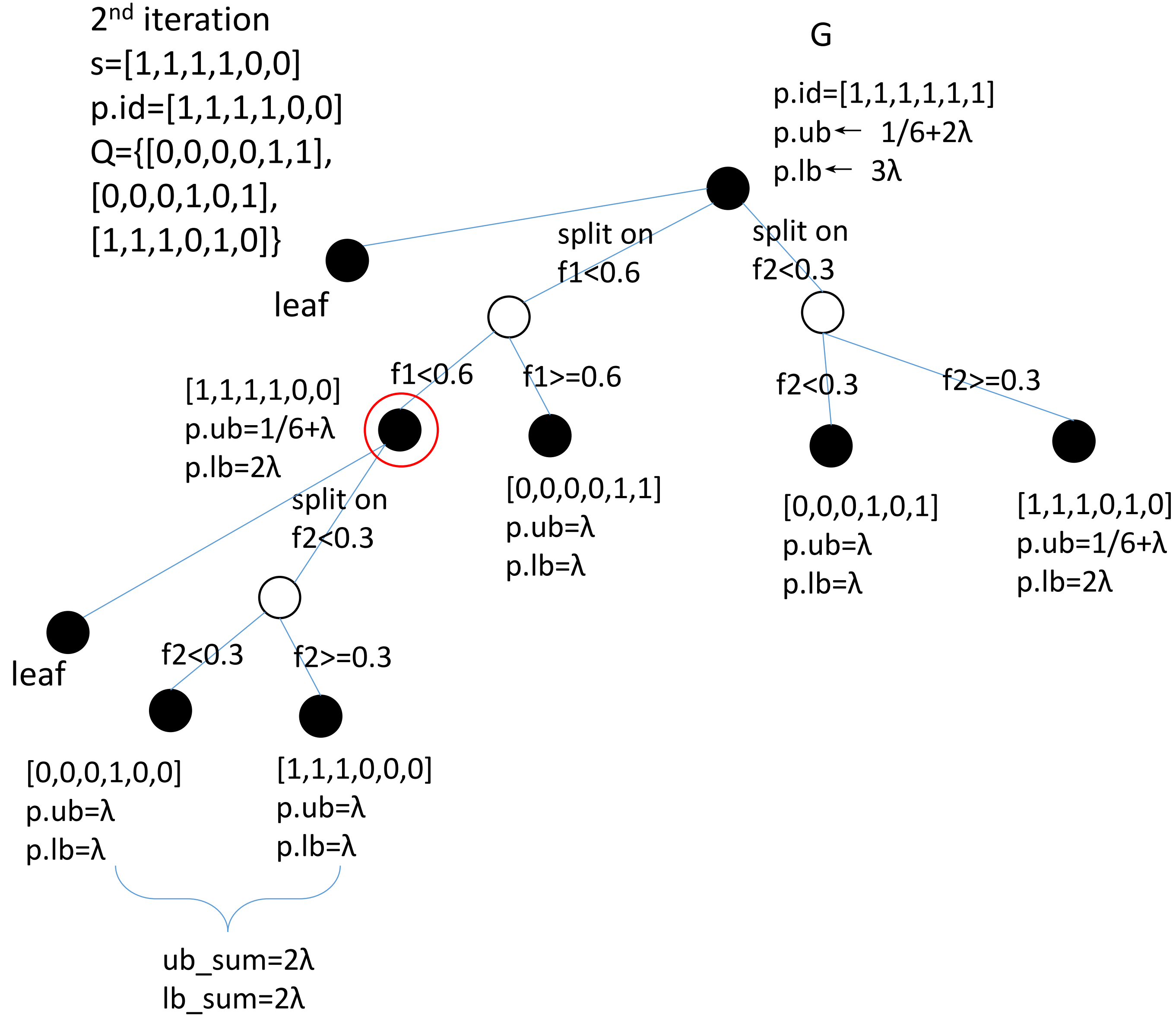
Q = {[1,1,1,1,0,0], [0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}

2nd iteration

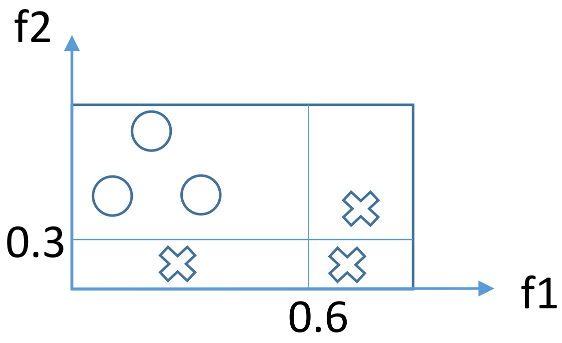
s=[1,1,1,1,0,0]

p.id=[1,1,1,1,0,0]

Q={[0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}



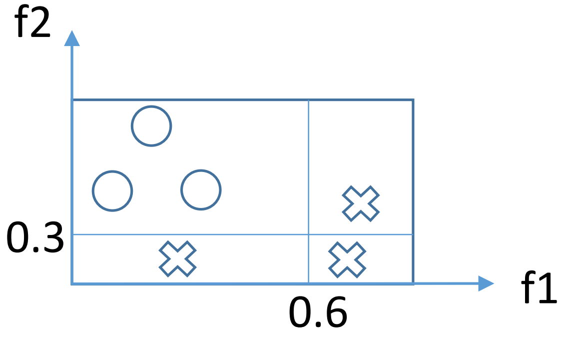
F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1



λ=0.1

Q = {[1,1,1,1,0,0], [0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}

F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1

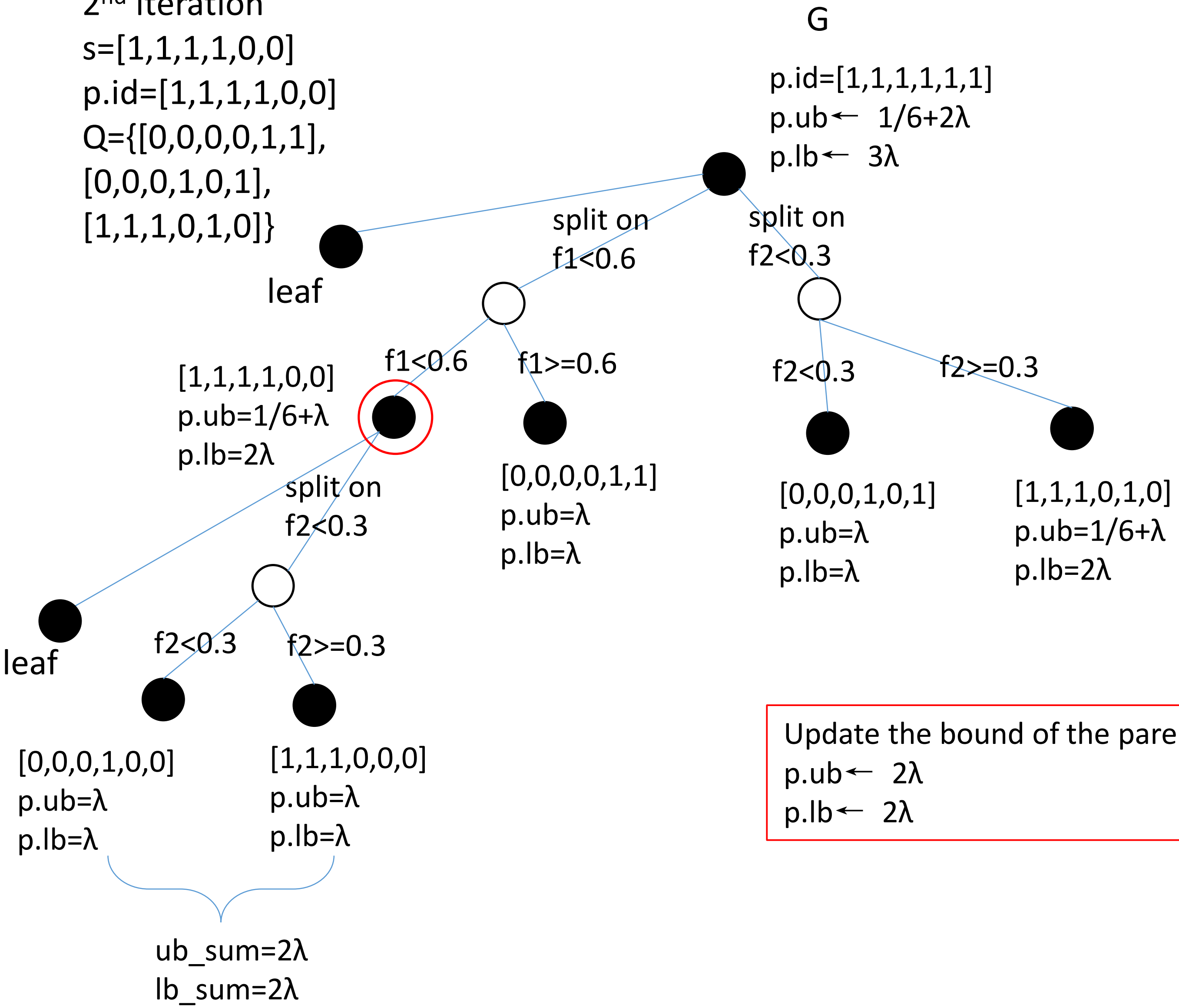


2nd iteration

s=[1,1,1,1,0,0]

p.id=[1,1,1,1,0,0]

Q={[0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}



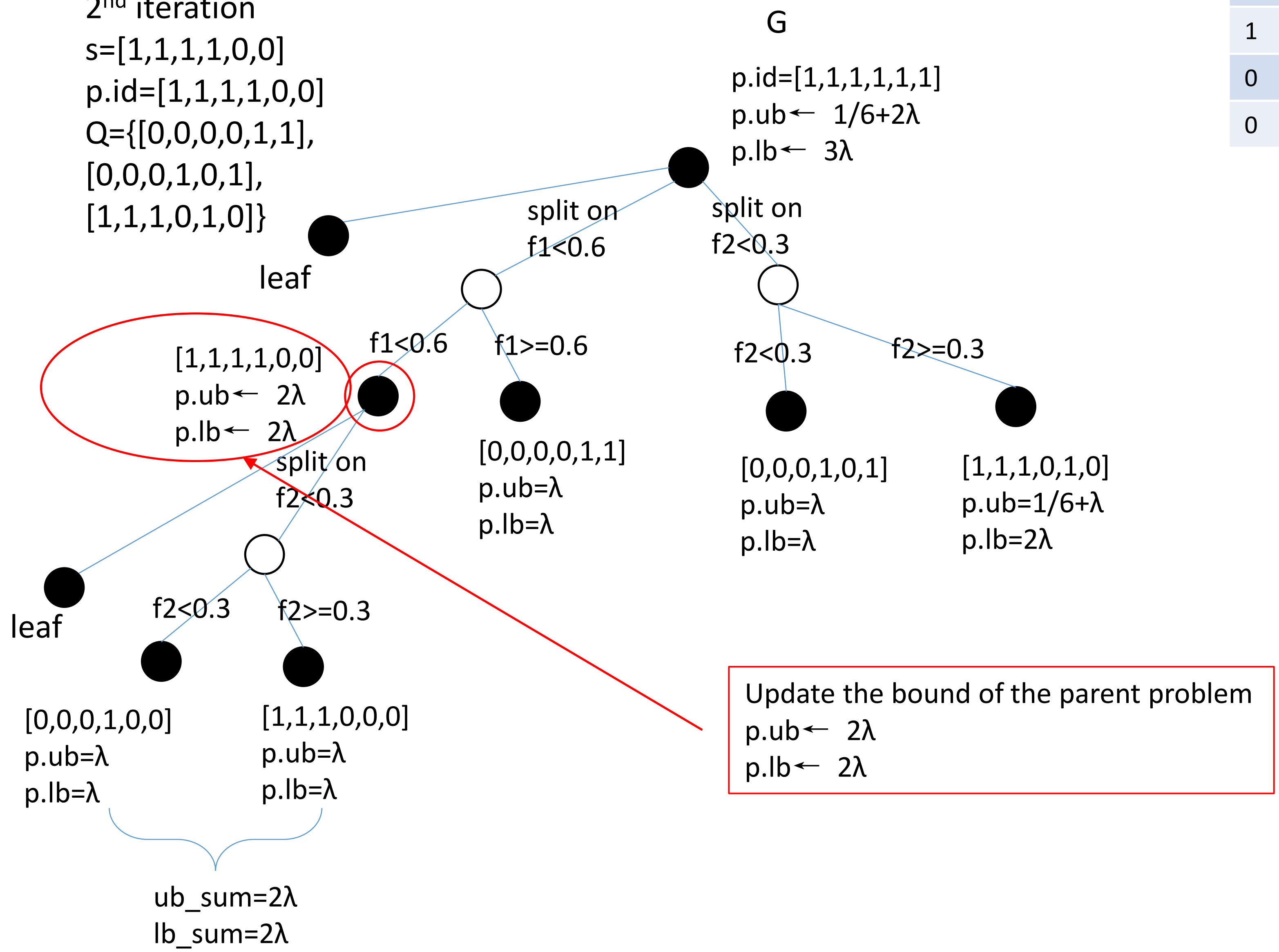
Update the bound of the parent problem

p.ub← 2λ

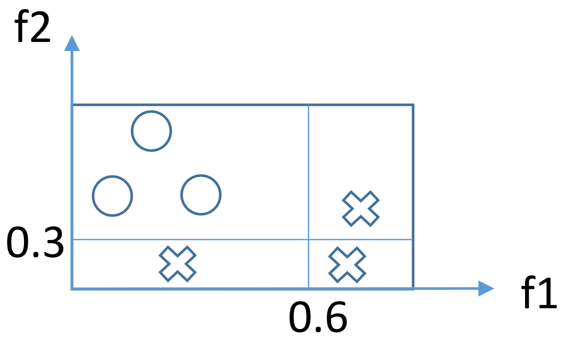
p.lb← 2λ

λ=0.1
Q = {[1,1,1,1,0,0], [0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}

2[nd] iteration
s=[1,1,1,1,0,0]
p.id=[1,1,1,1,0,0]
Q={[0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}

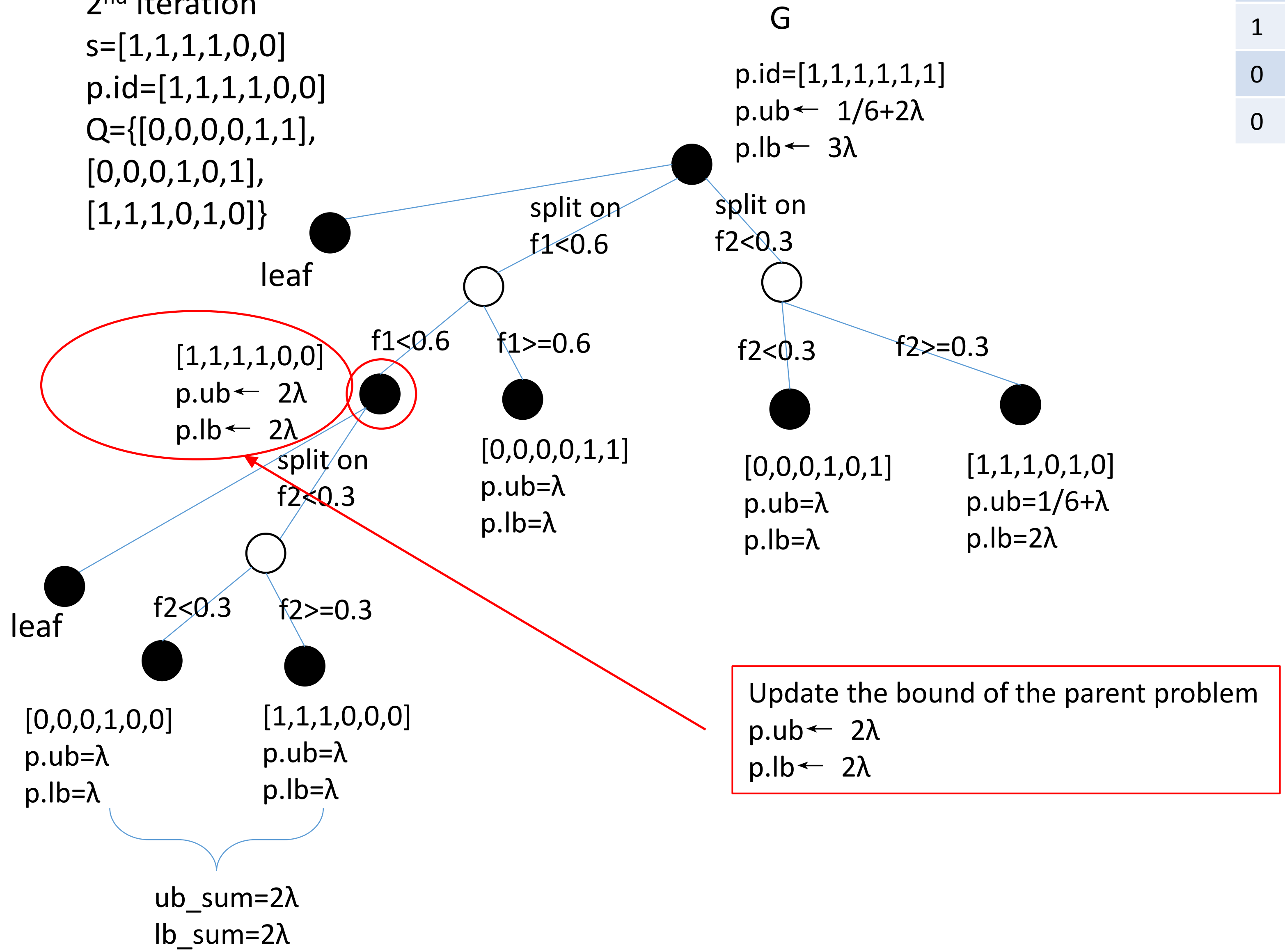


F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1



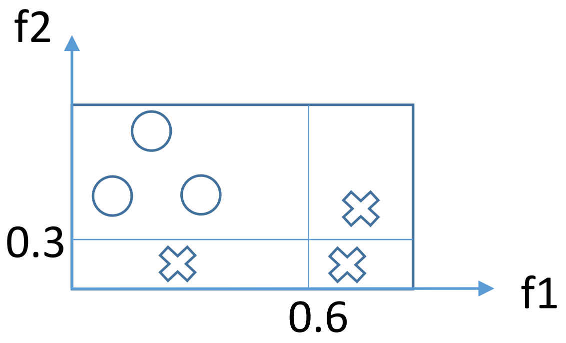
λ=0.1
Q = {[1,1,1,1,0,0], [0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}

2nd iteration
s=[1,1,1,1,0,0]
p.id=[1,1,1,1,0,0]
Q={[0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}



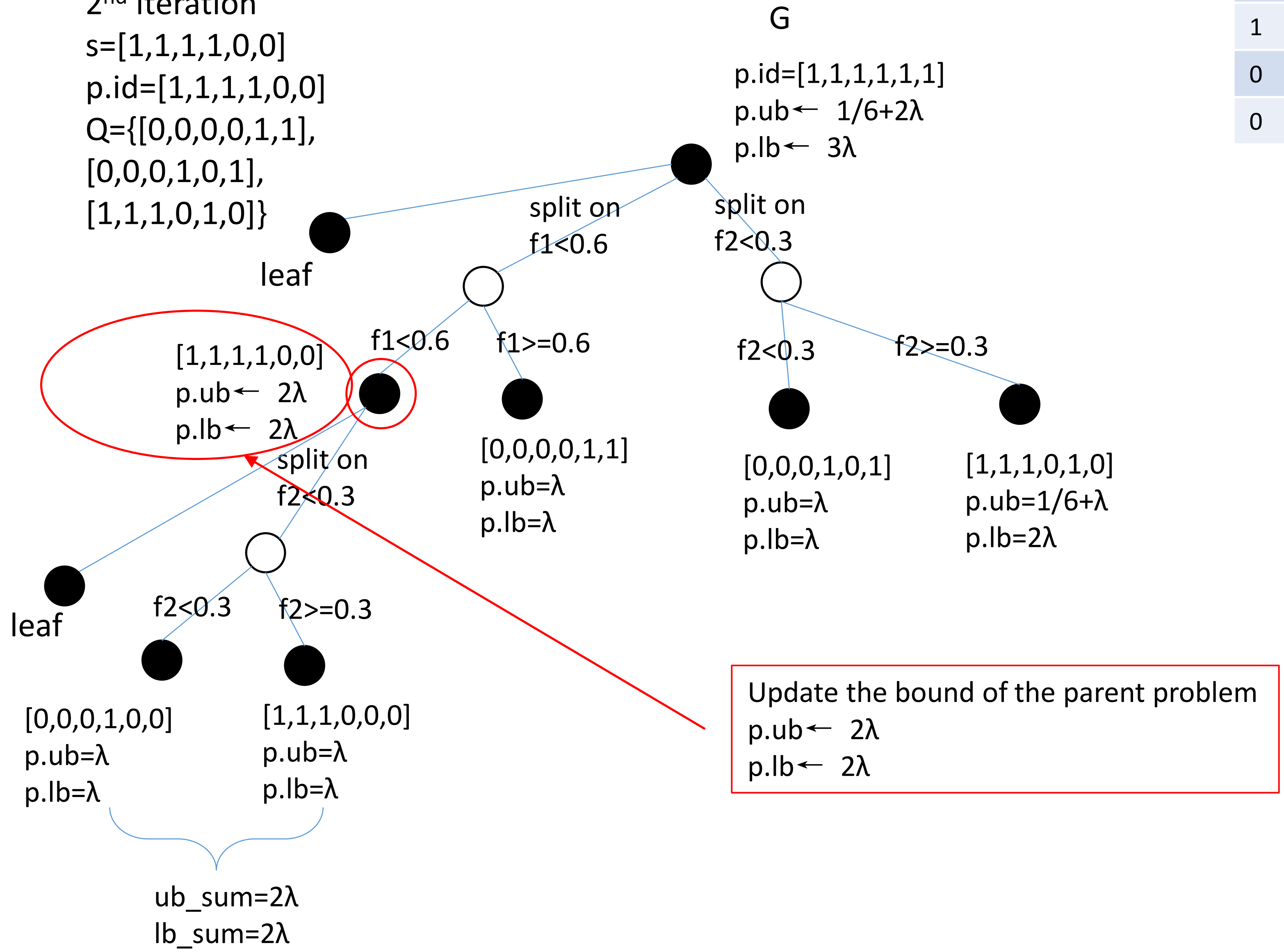
Since the problem is updated, we need to propagate this information to its parents

F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1

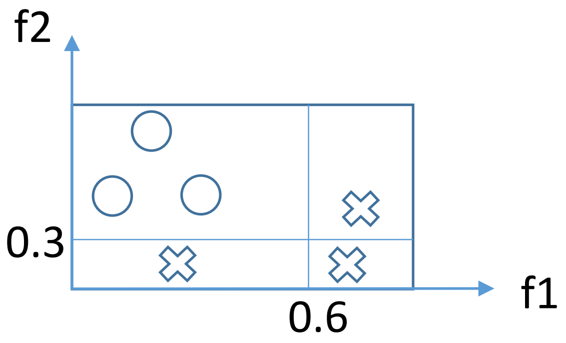


λ=0.1
Q = {[1,1,1,1,0,0], [0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}

2nd iteration
s=[1,1,1,1,0,0]
p.id=[1,1,1,1,0,0]
Q={[0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}



F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1



Update Queue! Let's enqueue its parents!

Q = {[1,1,1,1,1,1], [0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}

λ=0.1

Q ={[1,1,1,1,1,1], [0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}

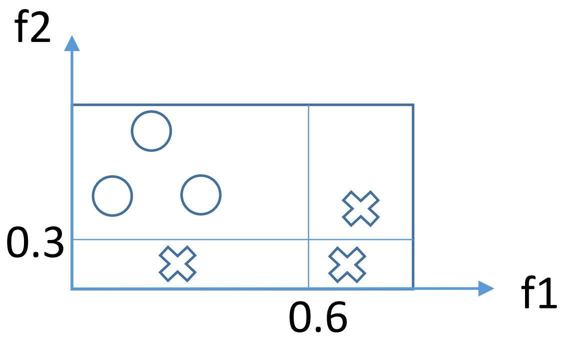
2^{nd} iteration

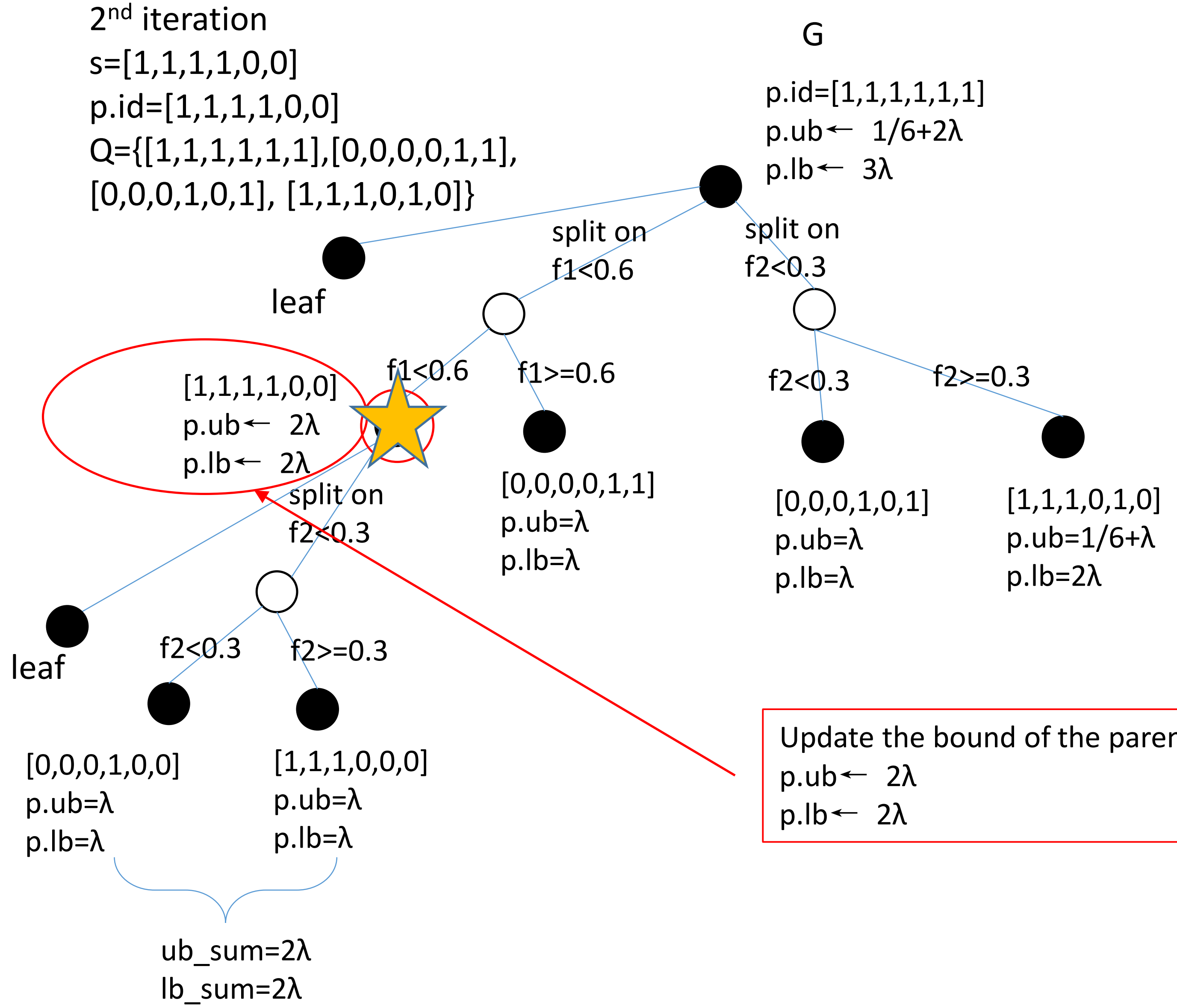
s=[1,1,1,1,0,0]

p.id=[1,1,1,1,0,0]

Q={[1,1,1,1,1,1],[0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}

F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1





Update the bound of the parent problem
p.ub← 2λ
p.lb← 2λ

Since the upper and lower of p is equal, this problem is solved. We jump back to the while loop.

$\lambda=0.1$

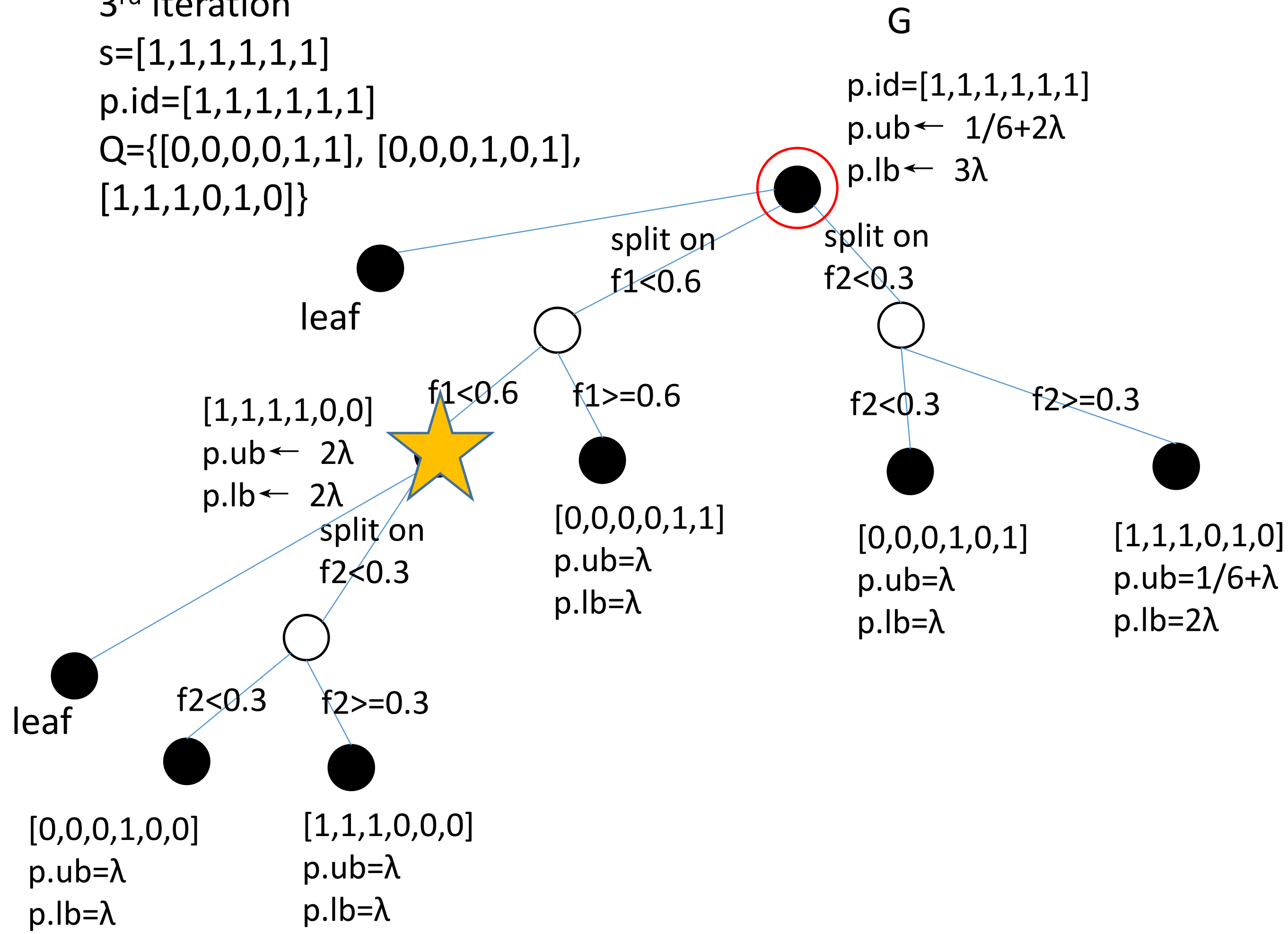
Q ={[1,1,1,1,1,1], [0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}

3rd iteration

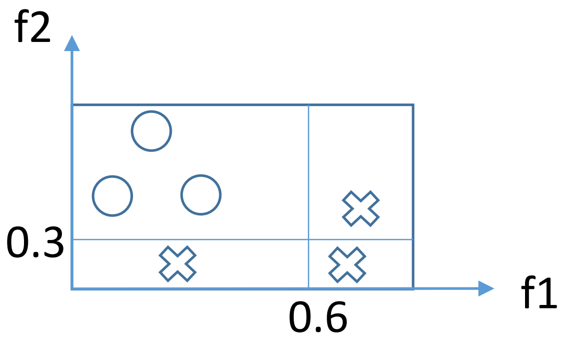
s=[1,1,1,1,1,1]

p.id=[1,1,1,1,1,1]

Q={[0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}



F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1



λ=0.1

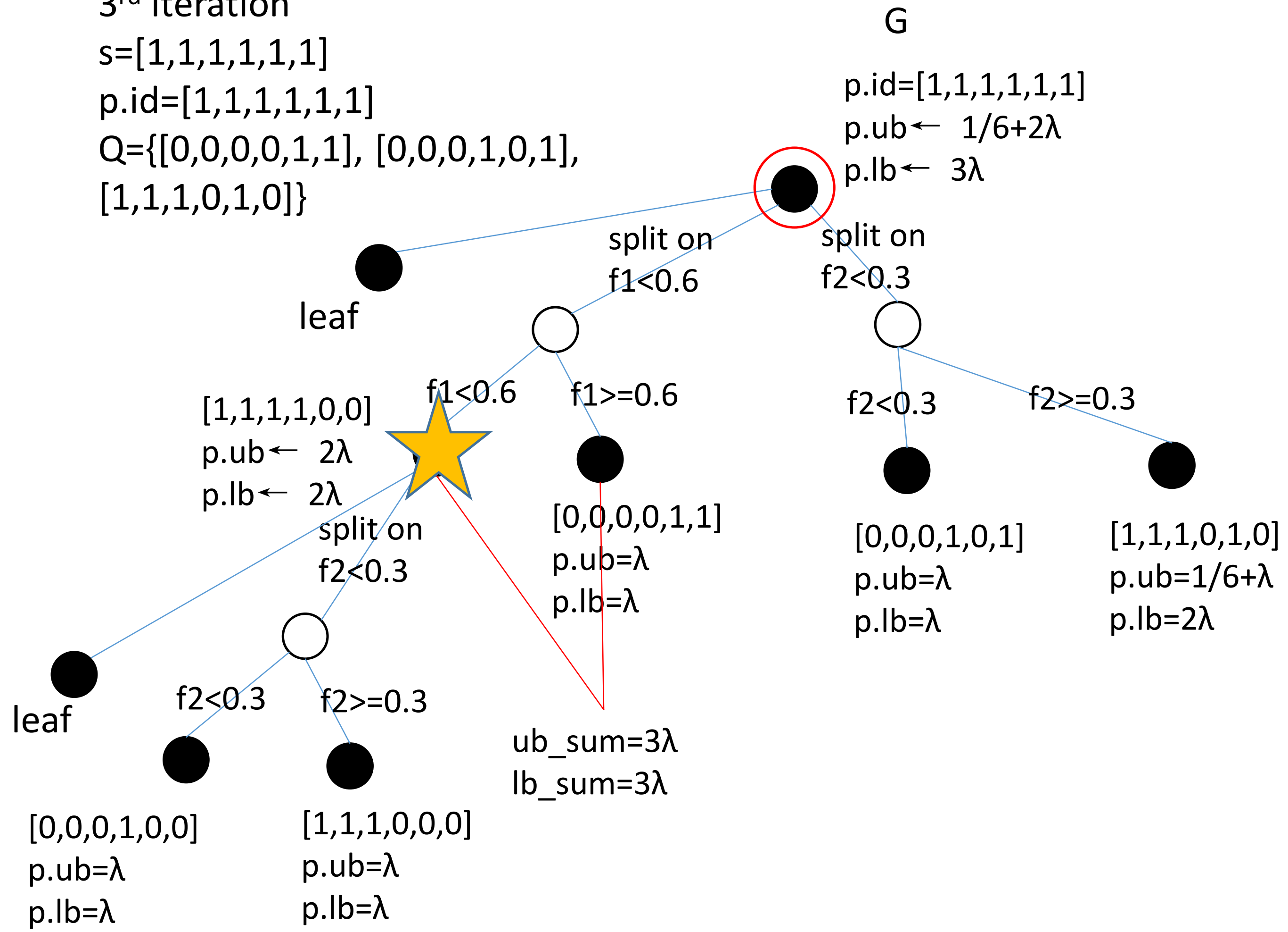
Q ={[1,1,1,1,1,1], [0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}

3[rd] iteration

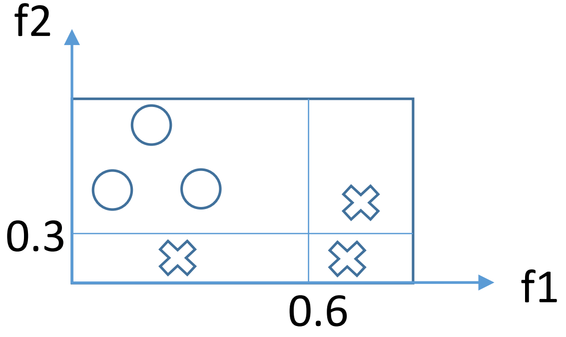
s=[1,1,1,1,1,1]

p.id=[1,1,1,1,1,1]

Q={[0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}



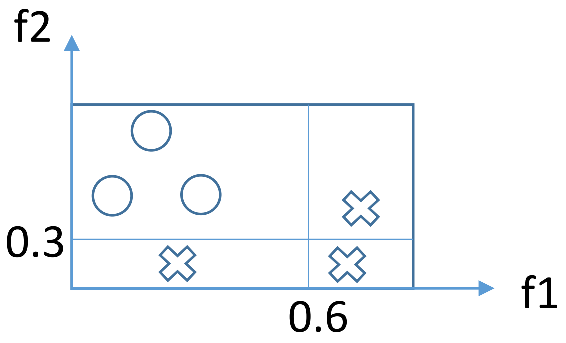
F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1



λ=0.1

Q ={[1,1,1,1,1,1], [0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}

F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1

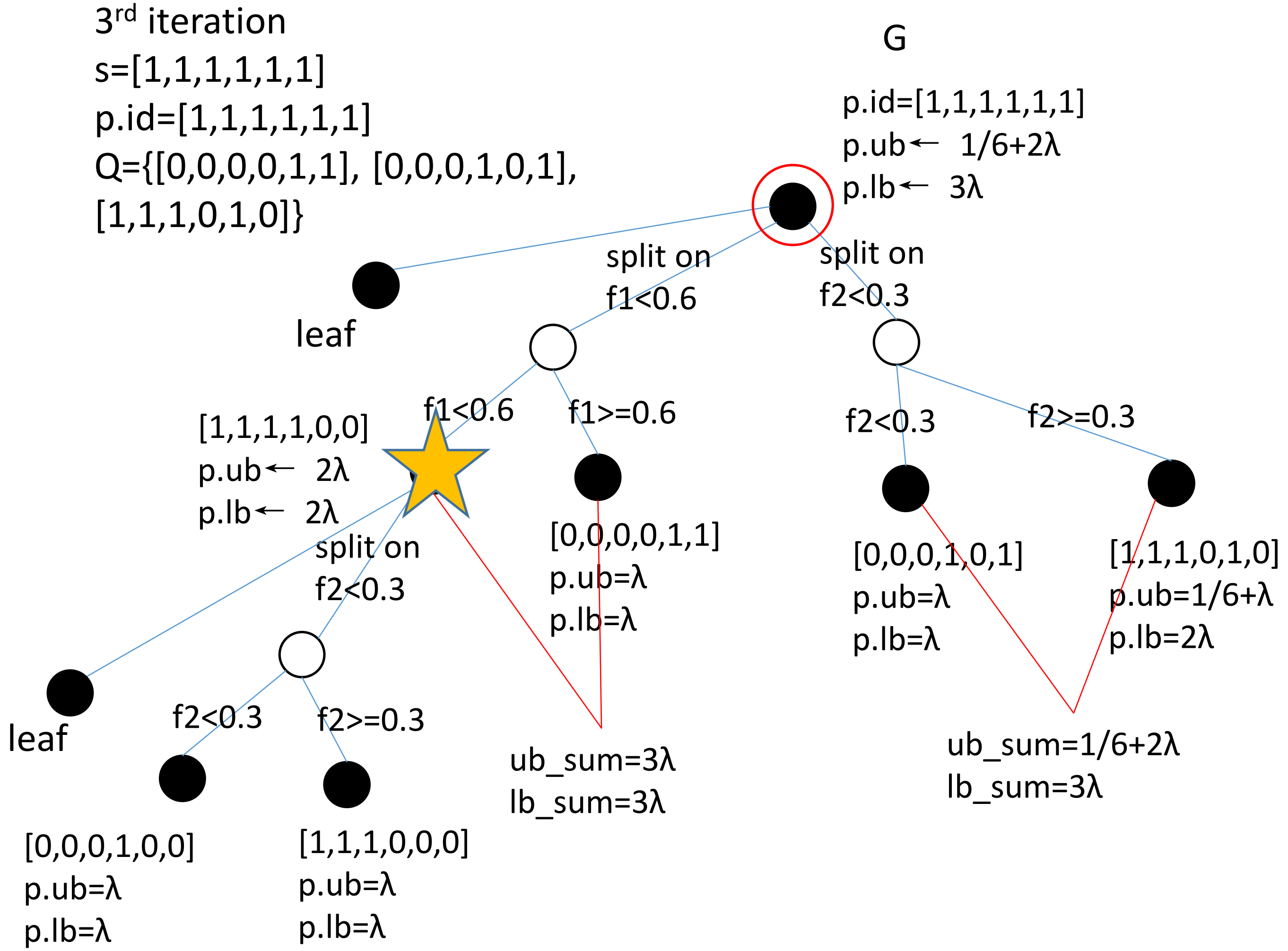


3[rd] iteration

s=[1,1,1,1,1,1]

p.id=[1,1,1,1,1,1]

Q={[0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}



Update the bound of p

p.ub← 3λ

p.lb← 3λ

λ=0.1

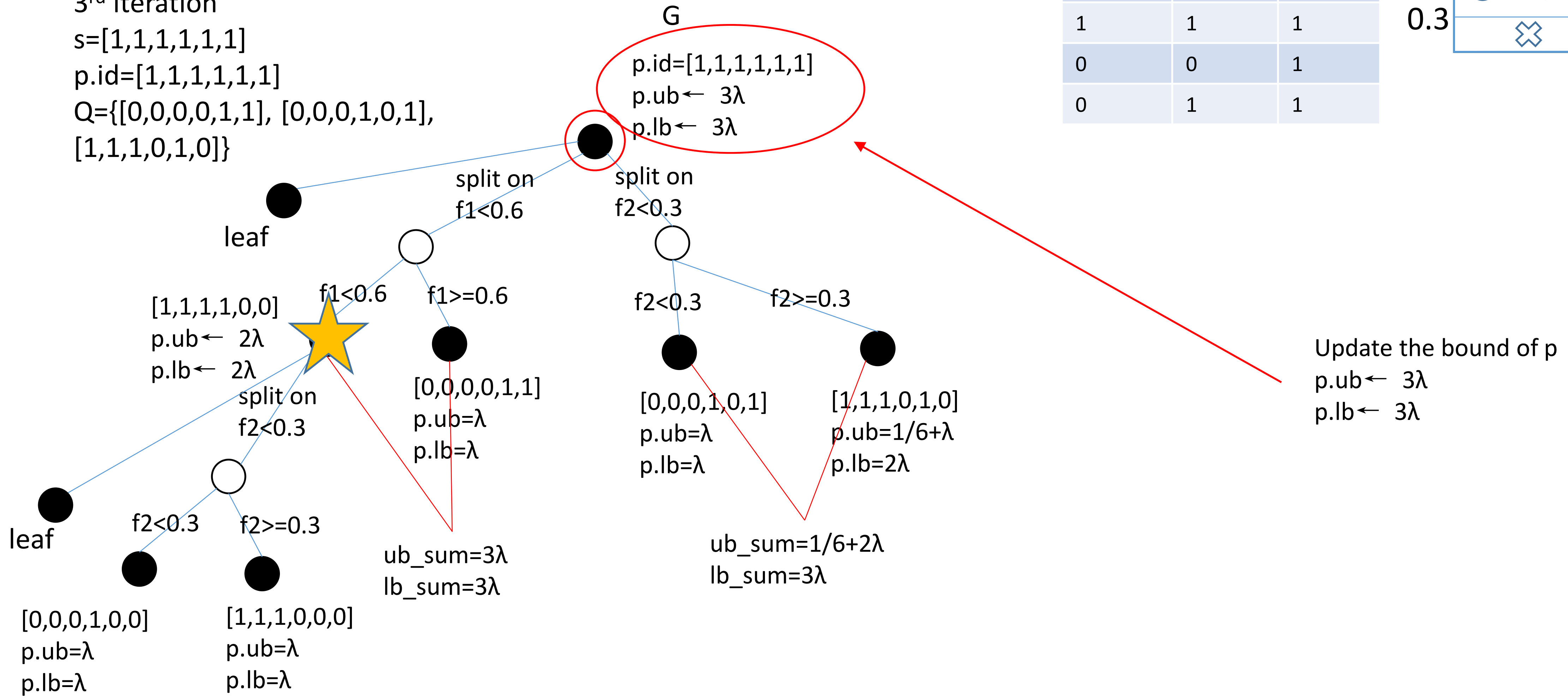
Q ={[1,1,1,1,1,1], [0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}

3rd iteration

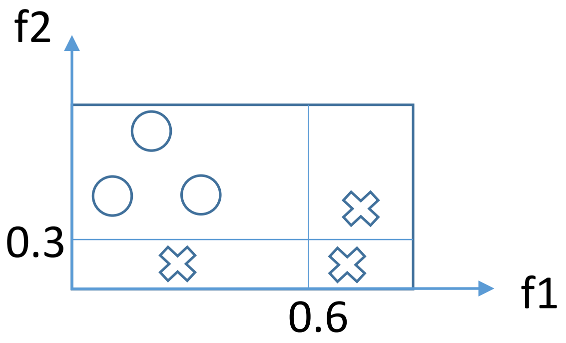
s=[1,1,1,1,1,1]

p.id=[1,1,1,1,1,1]

Q={[0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}



F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1



λ=0.1

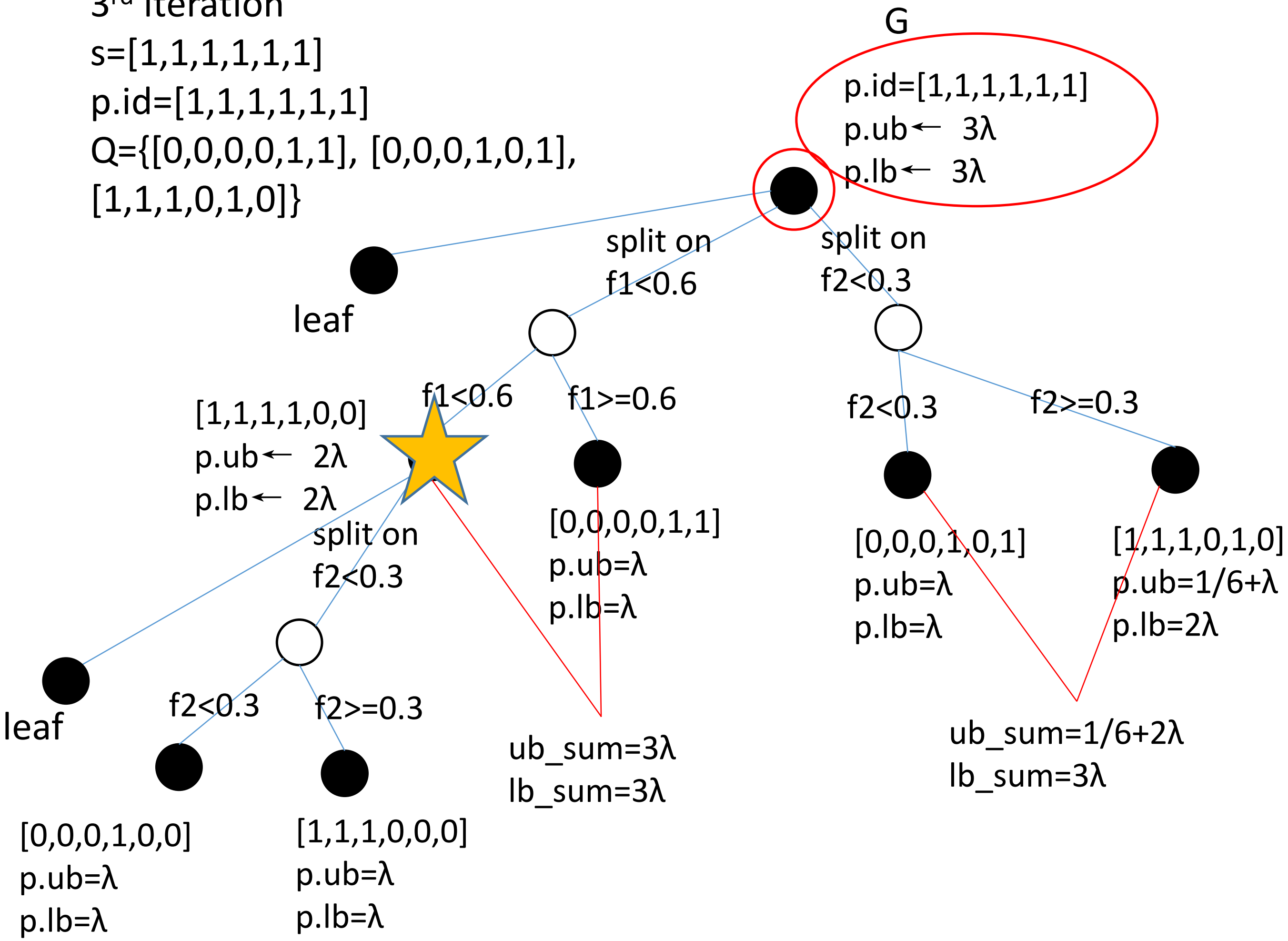
Q ={[1,1,1,1,1,1], [0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}

3rd iteration

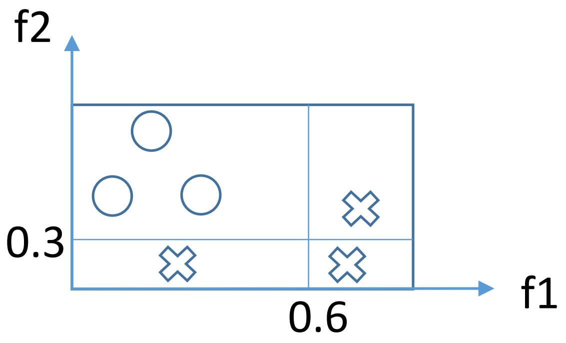
s=[1,1,1,1,1,1]

p.id=[1,1,1,1,1,1]

Q={[0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}



F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1

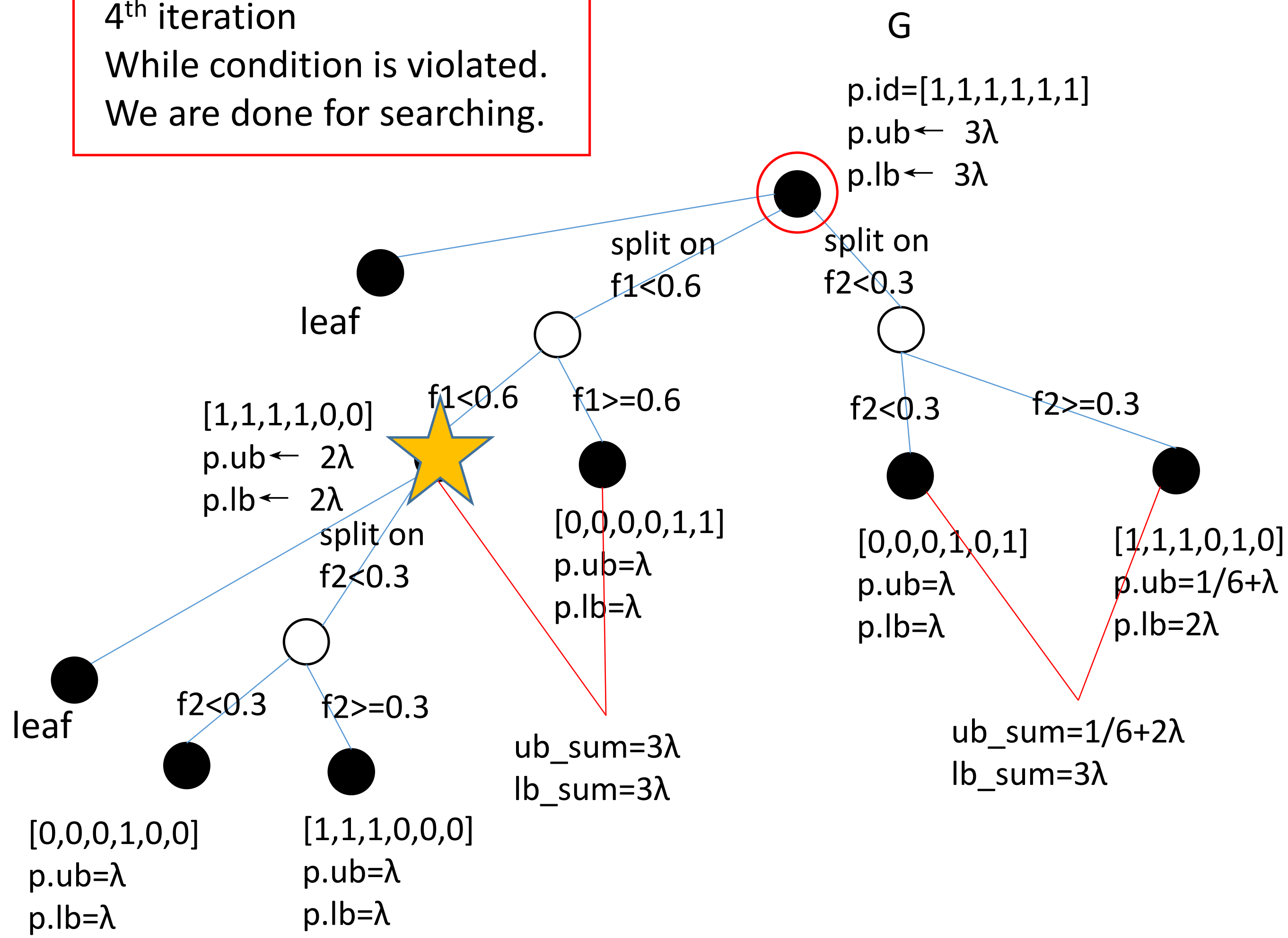


Since the lower and upper bounds are equal, jump back to the while condition.

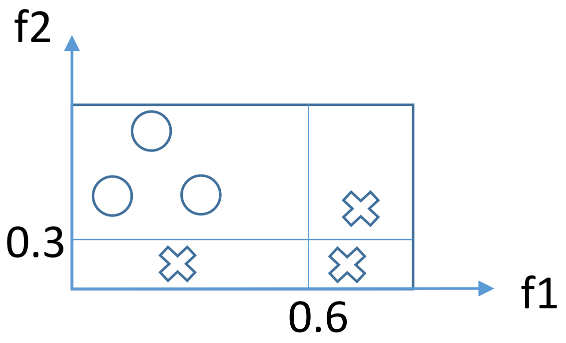
λ=0.1

Q ={[1,1,1,1,1,1], [0,0,0,0,1,1], [0,0,0,1,0,1], [1,1,1,0,1,0]}

4th iteration
While condition is violated.
We are done for searching.



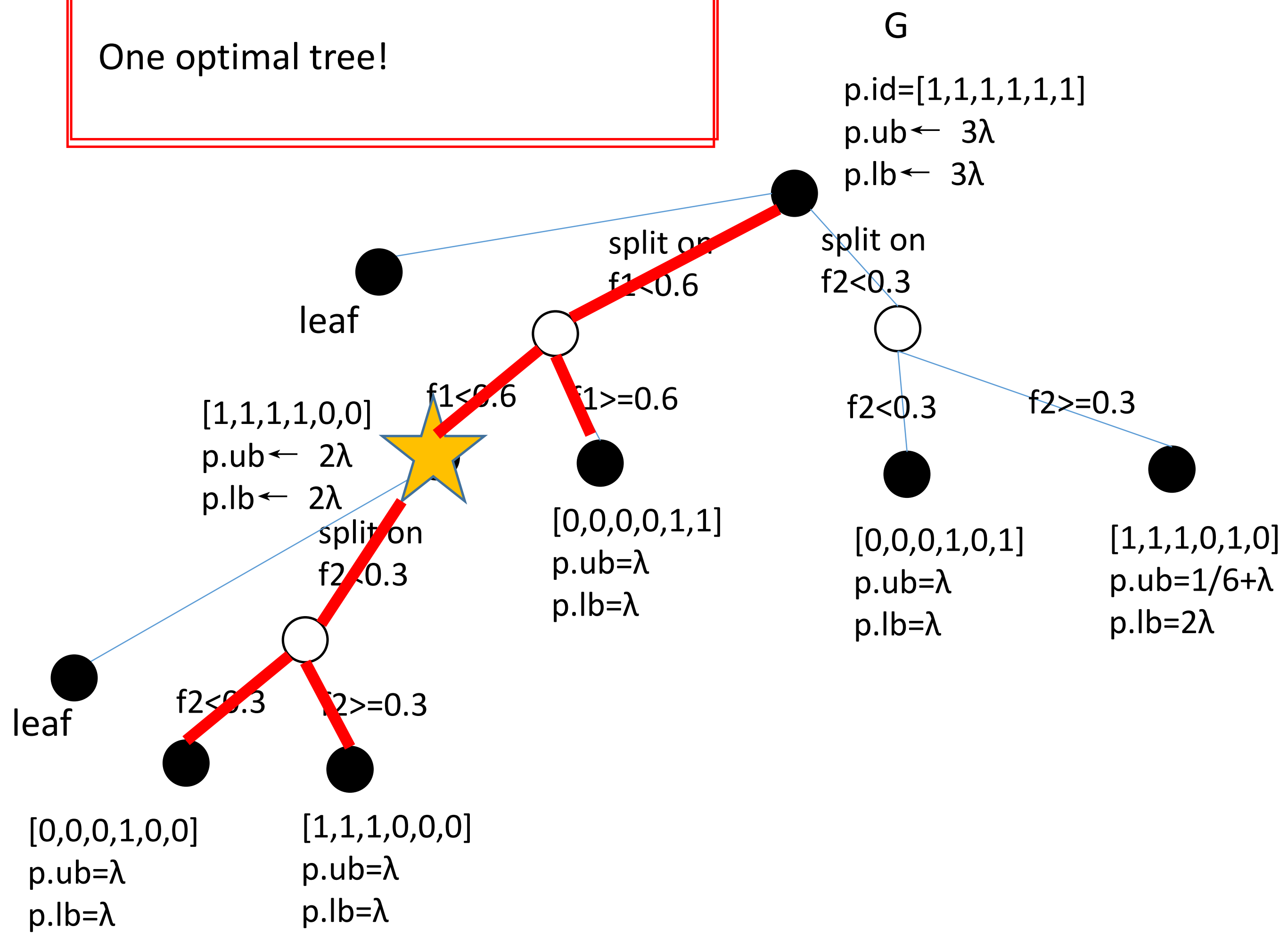
F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1



λ=0.1

Q {[0,0,0,1,0,1], [1,1,1,0,1,0]}

One optimal tree!



F1<0.6	F2<0.3	Label
1	0	0
1	0	0
1	0	0
1	1	1
0	0	1
0	1	1