Automatic Dominance Breaking for Constraint Optimization Problems

Jimmy H.M. Lee and Allen Z. Zhong
Computer Science and Engineering Department
The Chinese University of Hong Kong

Motivation

- Dominance Breaking (DB) constraints can speed up solving constraint optimization problems (COPs):
  - Permutation Problem (Kohler Jr et al. 1974)
  - Knapsack Problem (Poirriez et al. 2009)
  - Talent Scheduling (Qin et al. 2016, de la Banda et al. 2011)
  - Rectangle Packing (Korf 2004)
  - Minimization of Open Stack (Chu et al. 2009)
  - …
- <u>Most research works focus on problem specific DB constraints.</u>

Motivation

- Few studies on generic dominance breaking
  - Theoretical proof of that dominance breaking is effective (Ibaraki 1974)
  - Dichotomies of dominance rules (Jouglet and Carlier 2011)
  - A generic method for dominance breaking (Chu and Stuckey 2012)
    - Require manual derivation of dominance breaking constraints
    - Still require human insights into the problem
  - Semi-automatic dominance breaking by symmetry inference (Mears and de la Banda 2015)
    - Only generate constraints that can be manually found
    - Require manual selection of dominance breaking constraints

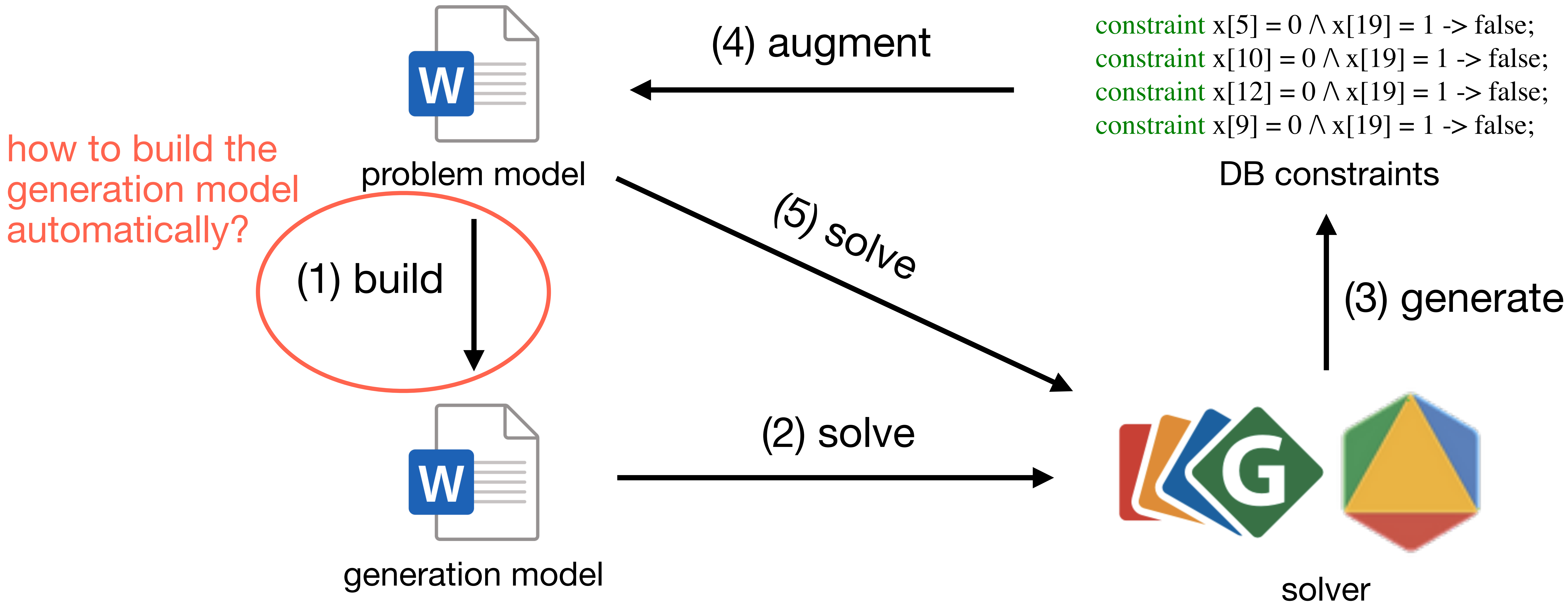
Contributions

- Full automation
- Solver independence
- More efficient than manual method

Contributions

- Full automation
- Solver independence
- More efficient than manual method

Method Overview



Agenda

- Background
- Automatic Dominance Breaking
- Experimental Results
- Q & A

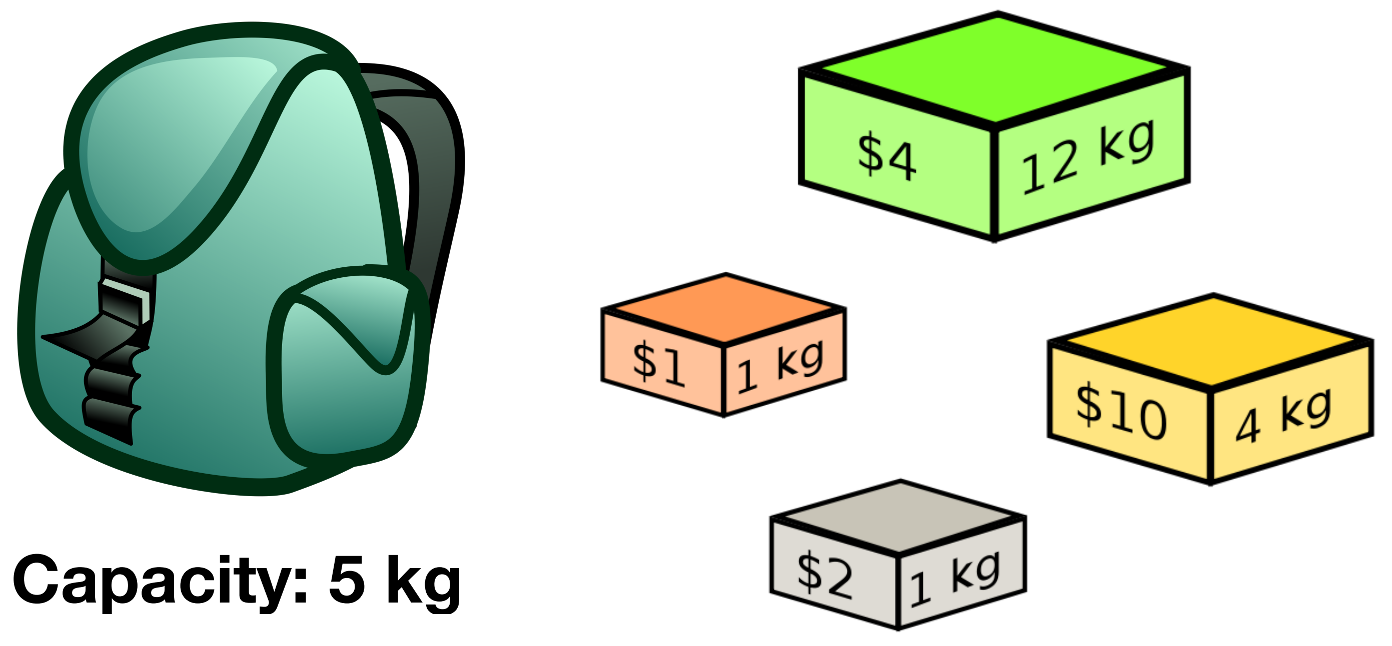
Background

- Constraint Satisfaction Problem (CSP): $P = (X, D, C)$
  - X: a set of variables $x_1, \ldots, x_n$
  - D: each variable has a domain $D(x_i)$
  - C: a set of constraints $\{c_i\}$
- Constraint Optimization Problem (COP): $P = (X, D, C, f)$
  - f: objective function to be minimized

Background

Constraint Optimizaition Problem

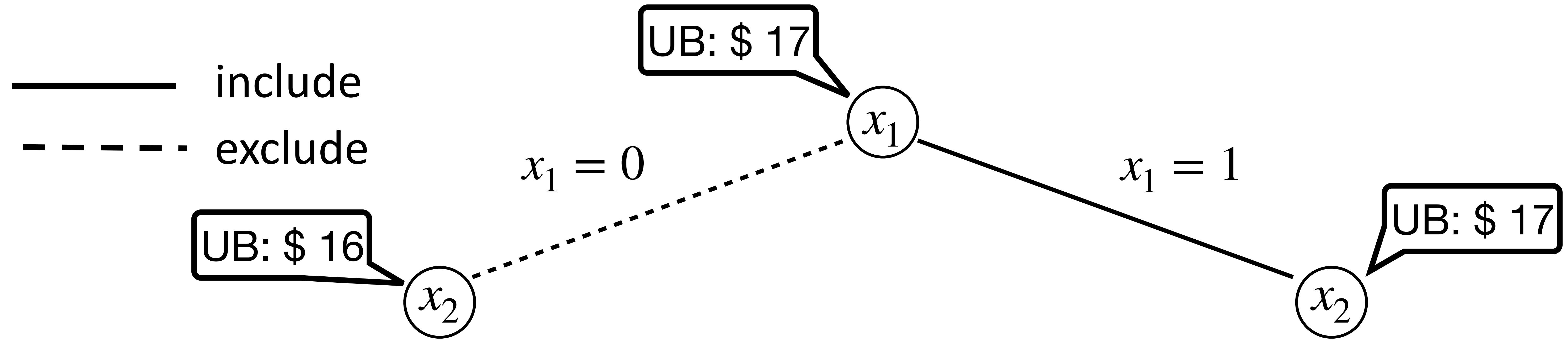
- Variables: x_i where $i = 1,\ldots,4$
- Domains: $x_i \in \{0,1\}$
- Contraint: $x_1 + x_2 + 4x_3 + 12x_4 \leq 5$
- Objective: $\min -(x_1 + 2x_2 + 10x_3 + 4x_4)$
- Goal: Find an optimal solution

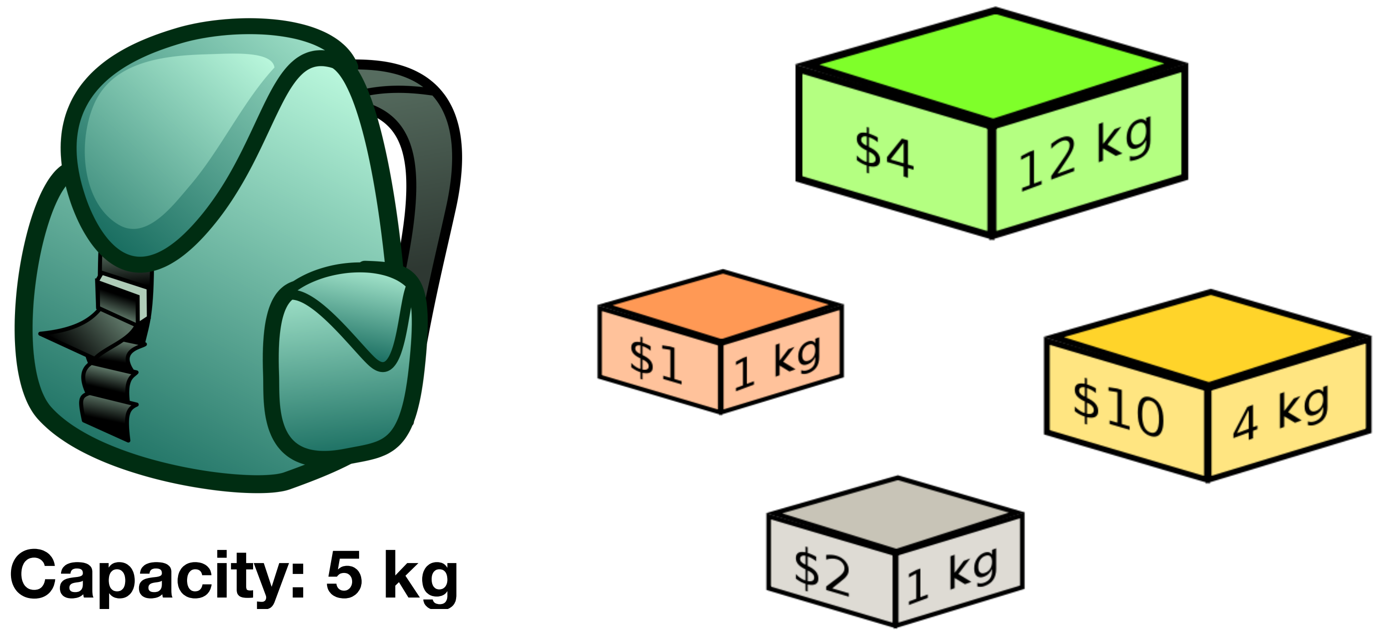


Capacity: 5 kg

Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4

Branch and Bound





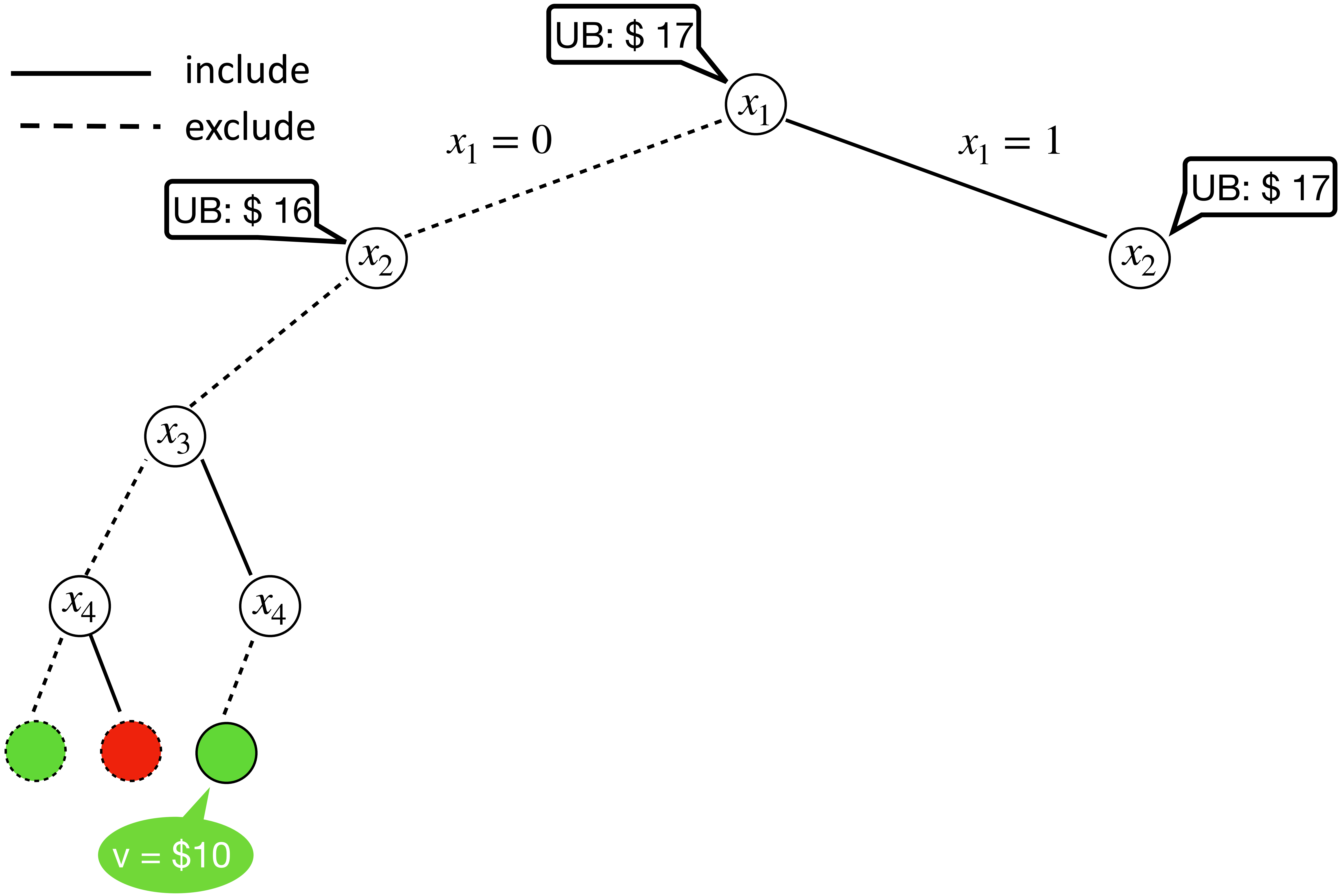
Capacity: 5 kg

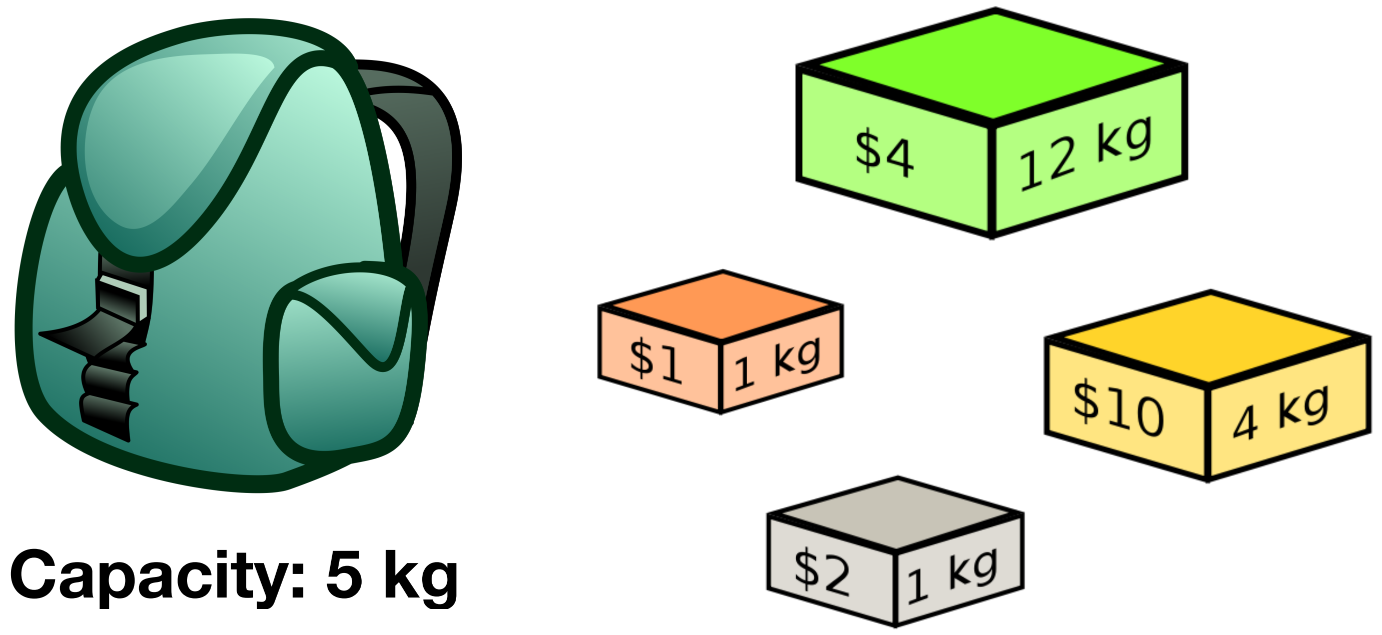
Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4

Tracking best value:

LB: $0

Branch and Bound



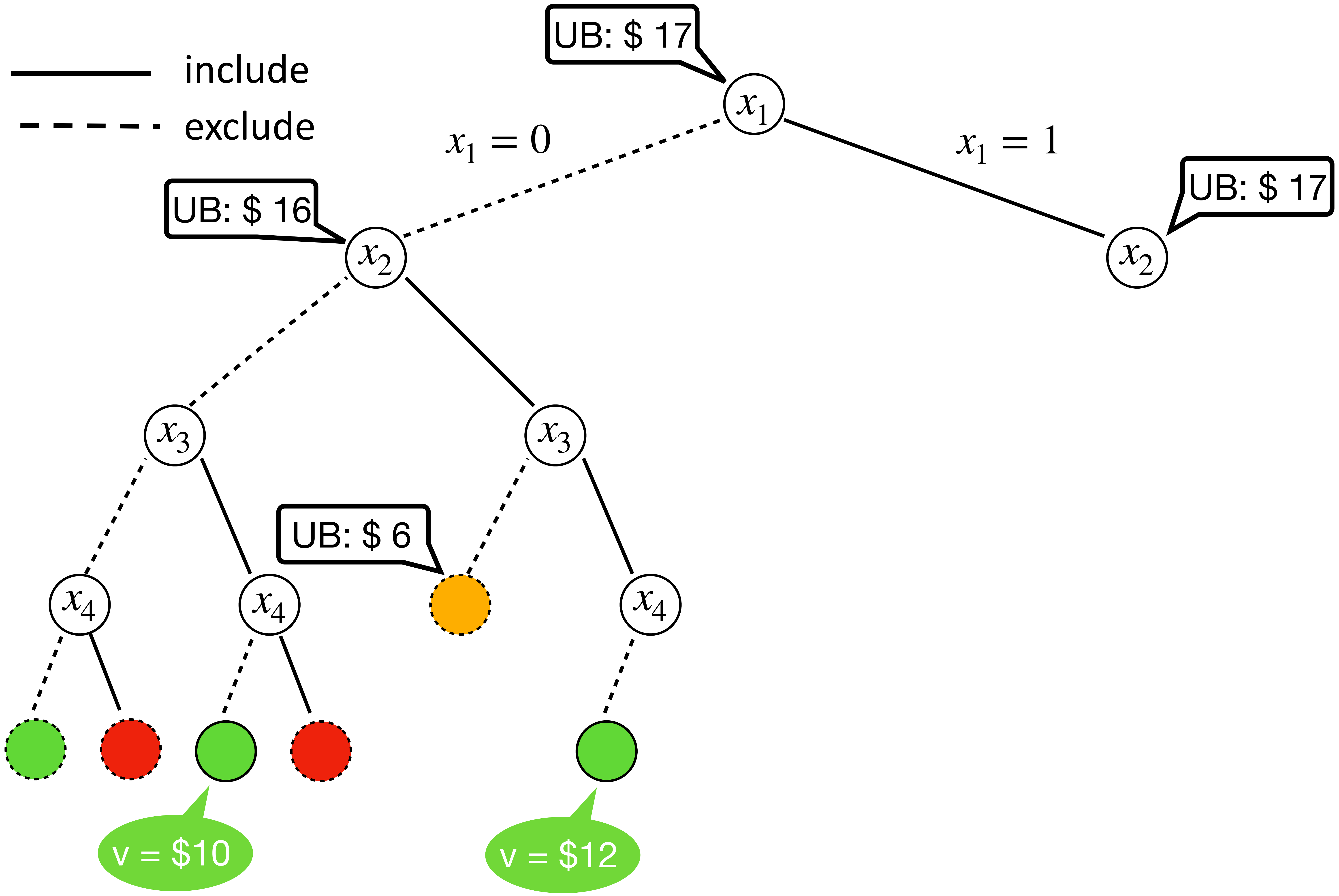


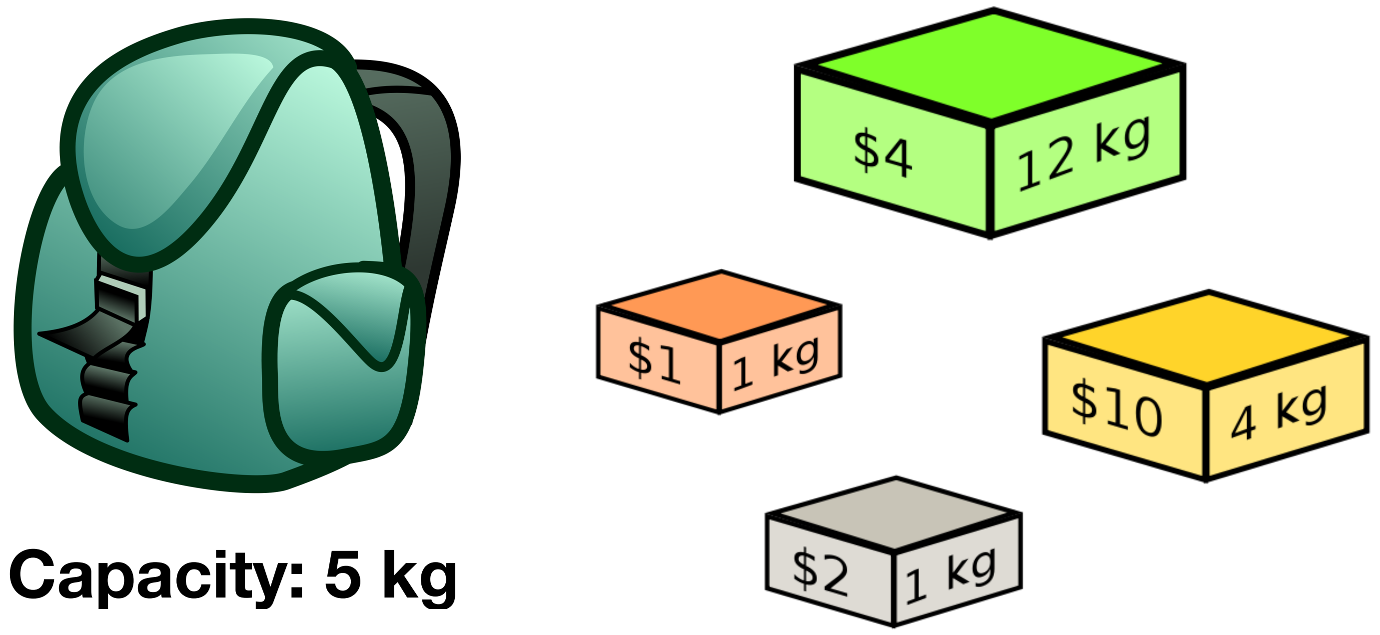
Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4

Tracking best value:

LB: $10

Branch and Bound





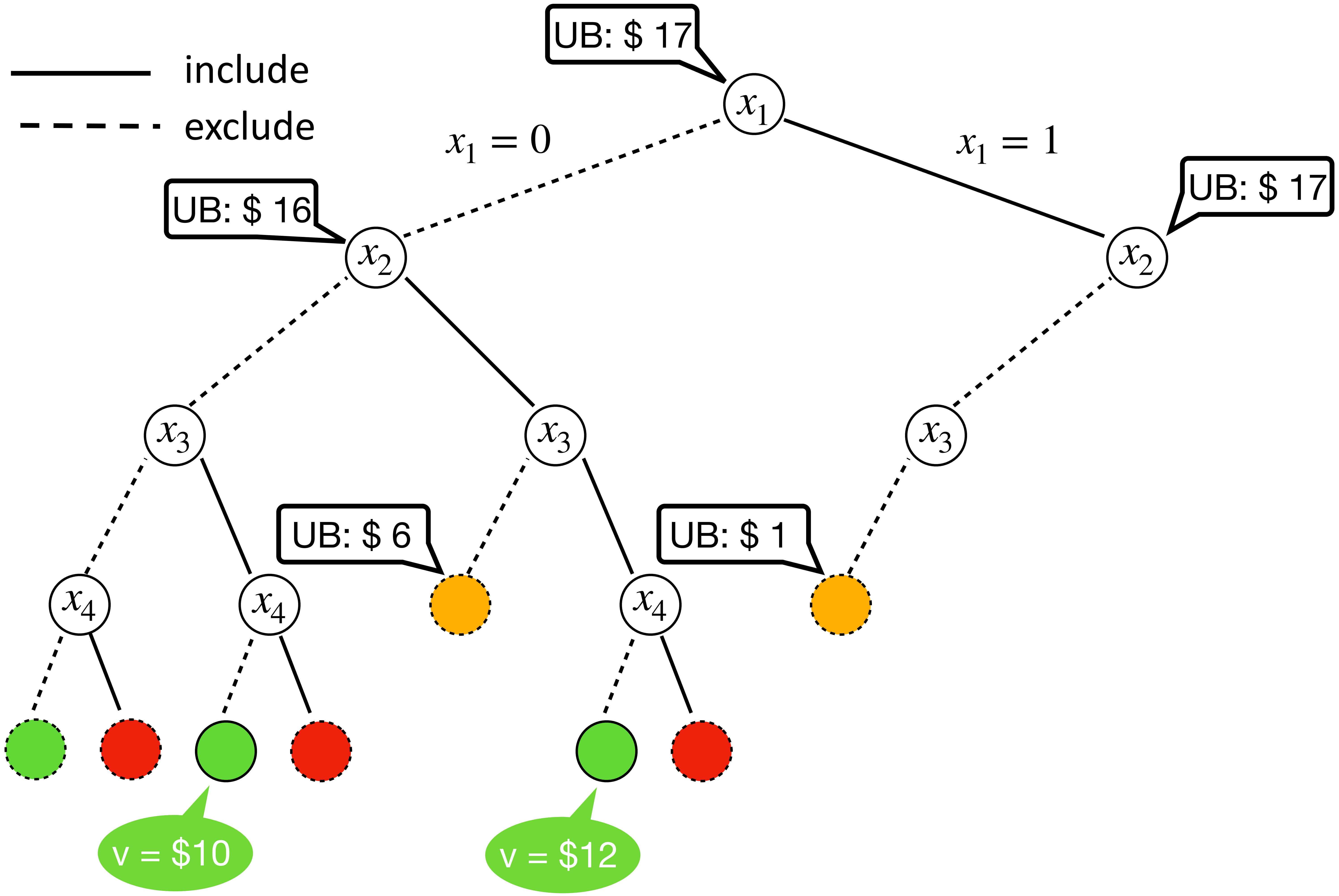
Capacity: 5 kg

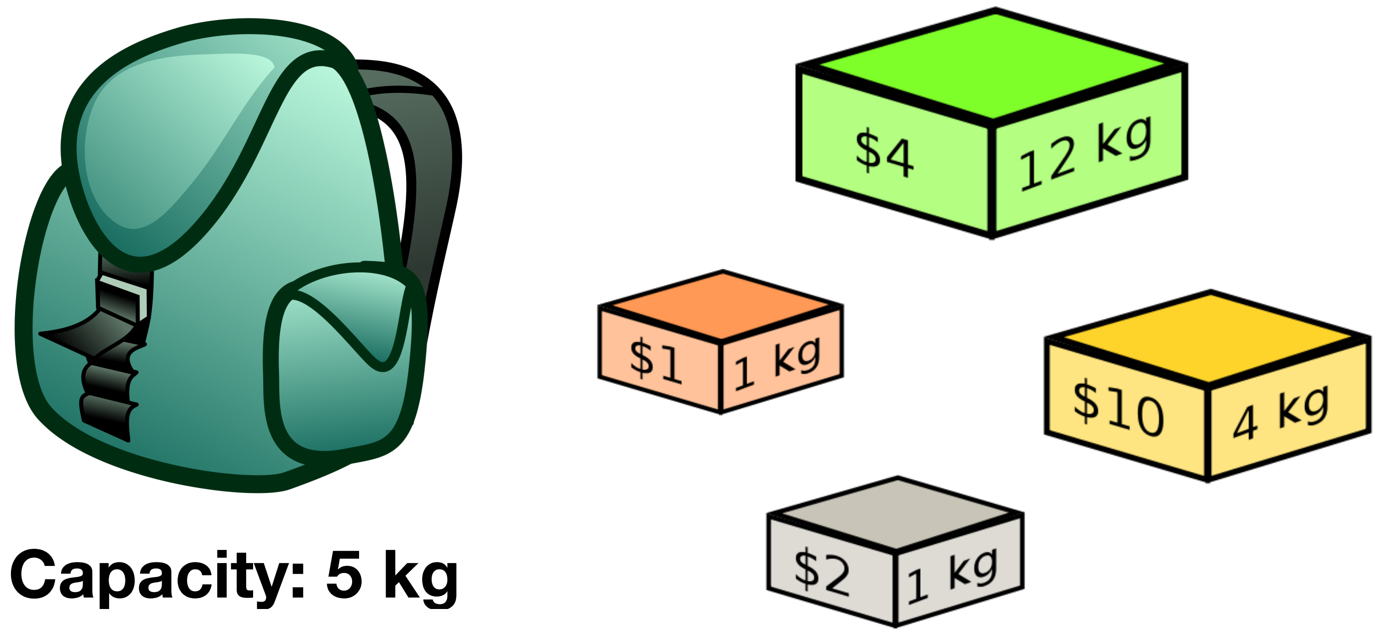
Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4

Tracking best value:

LB: $12

Branch and Bound



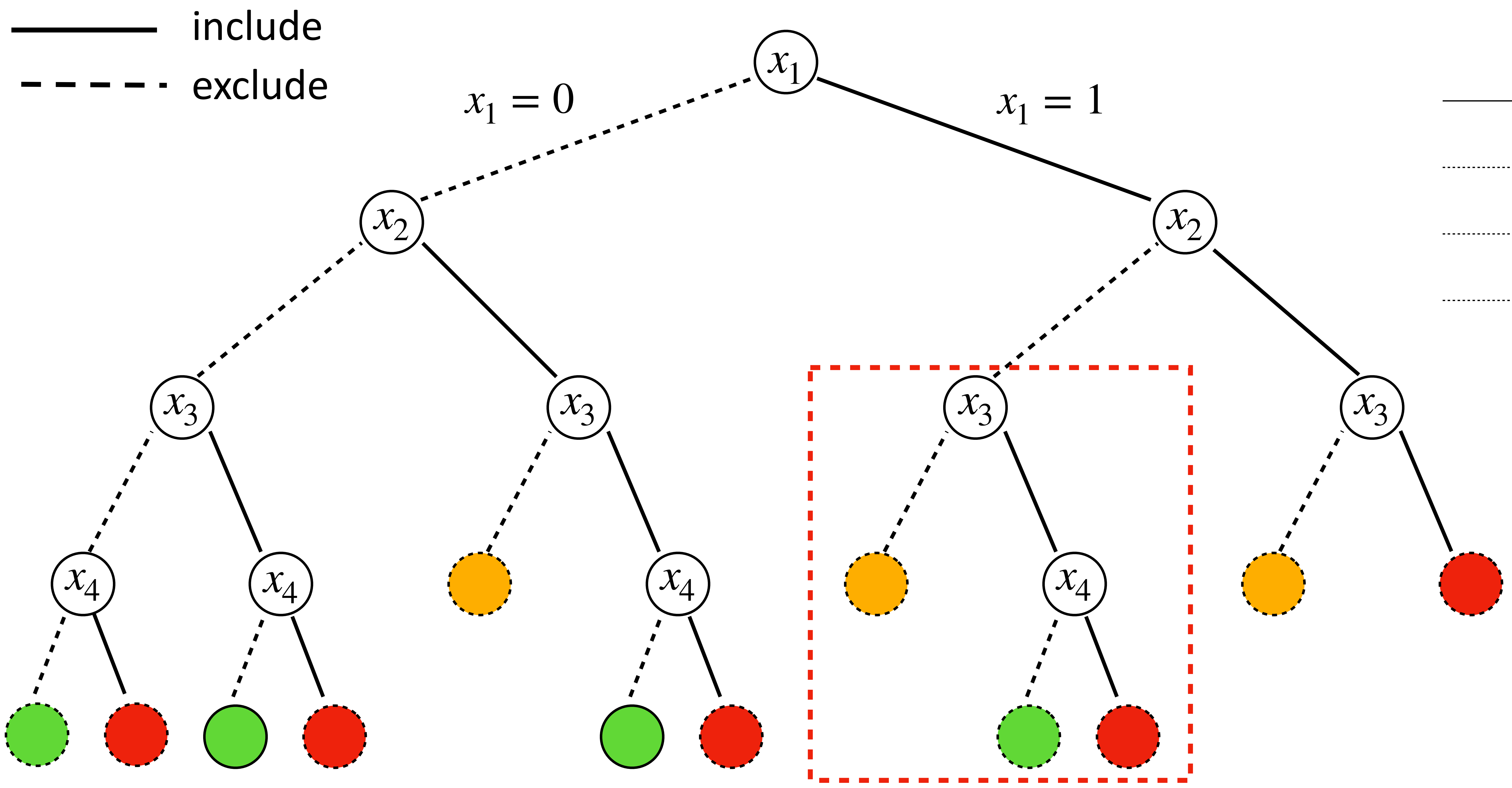


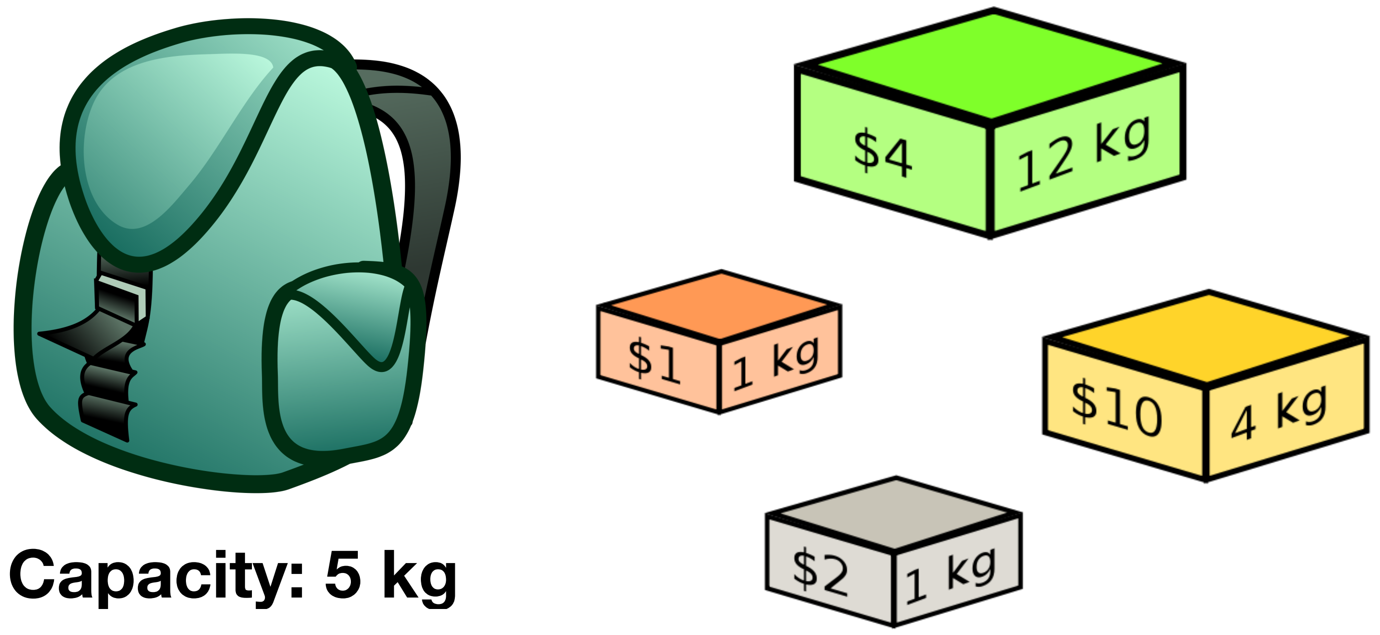
Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4

Tracking best value:

LB: $12

Dominance Breaking

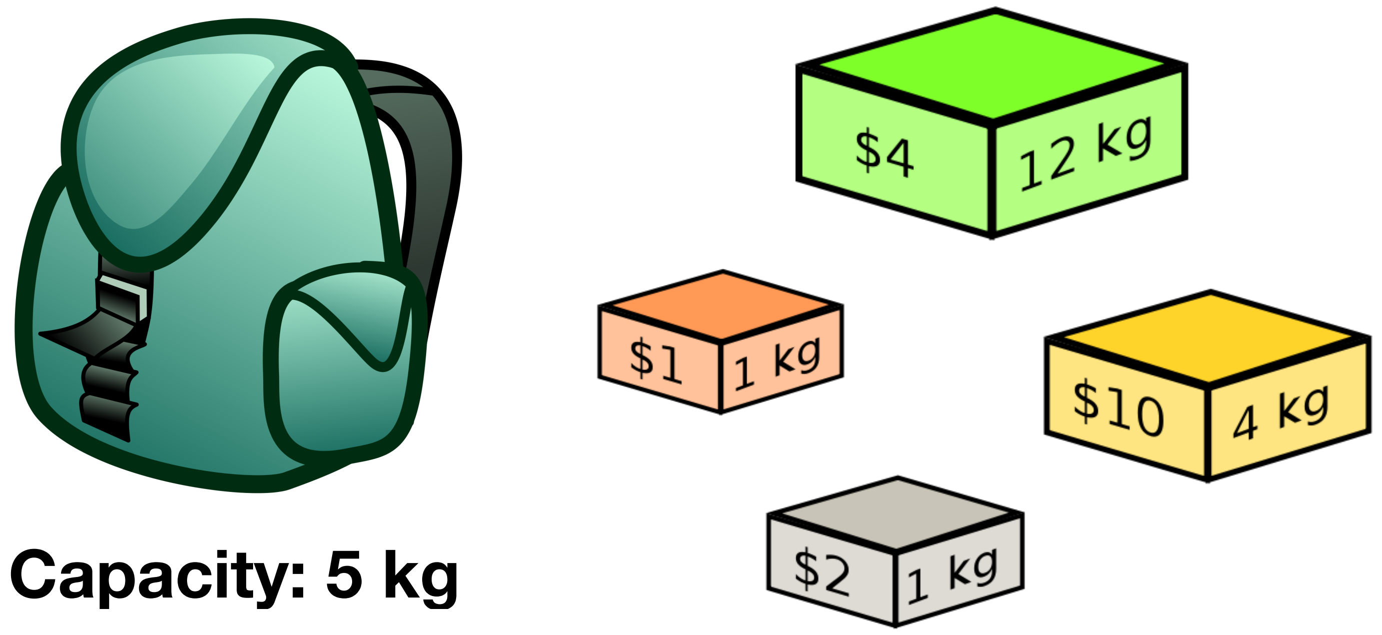




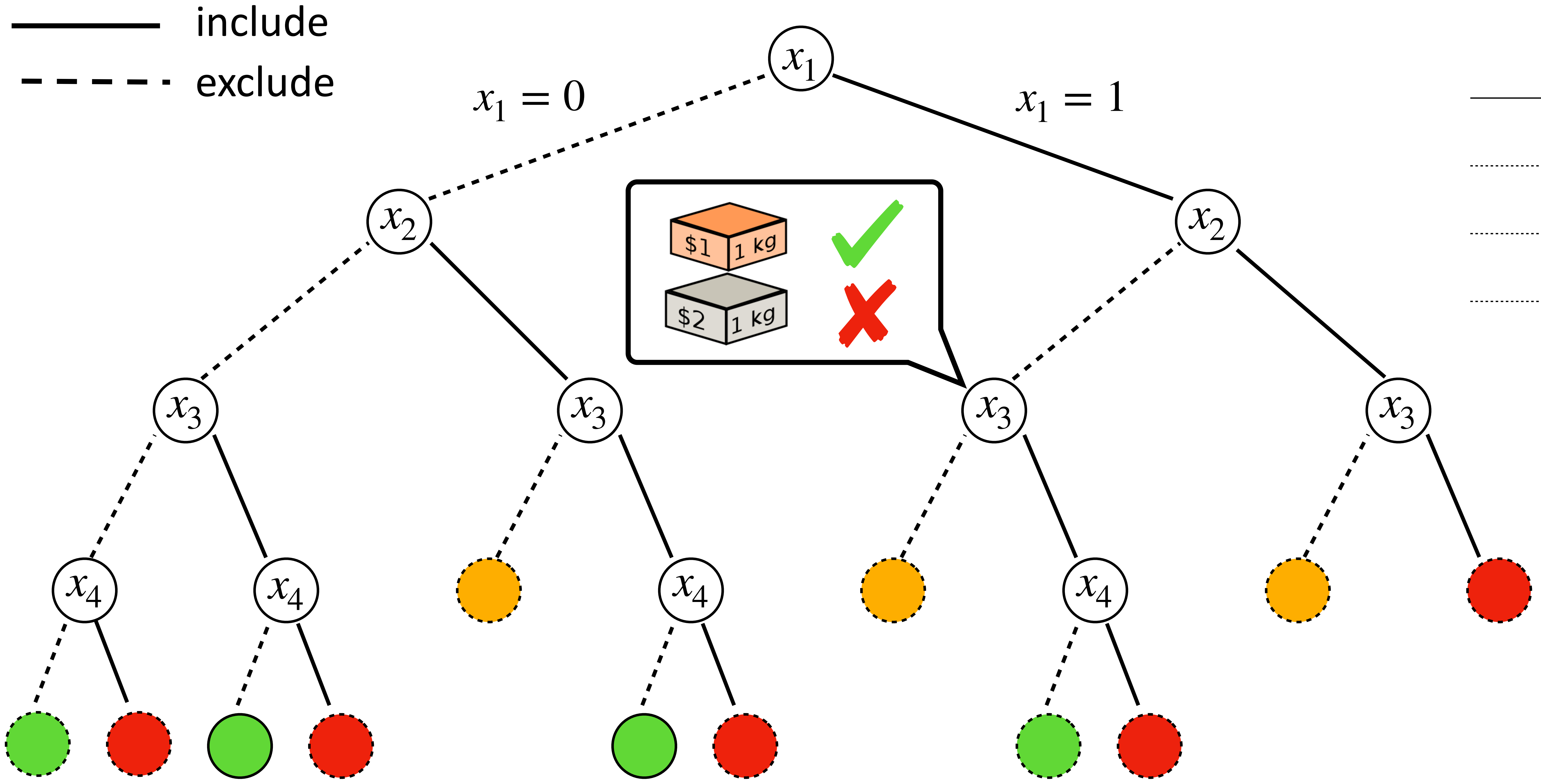
Capacity: 5 kg

Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4

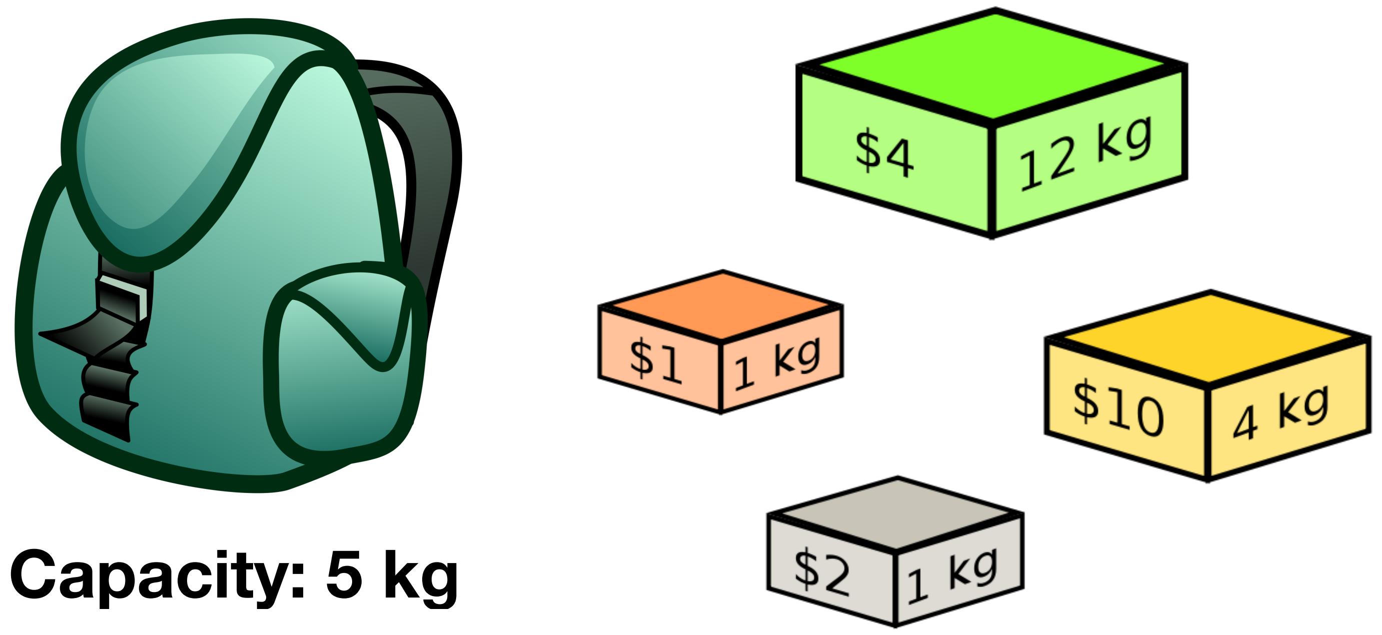
Dominance Breaking

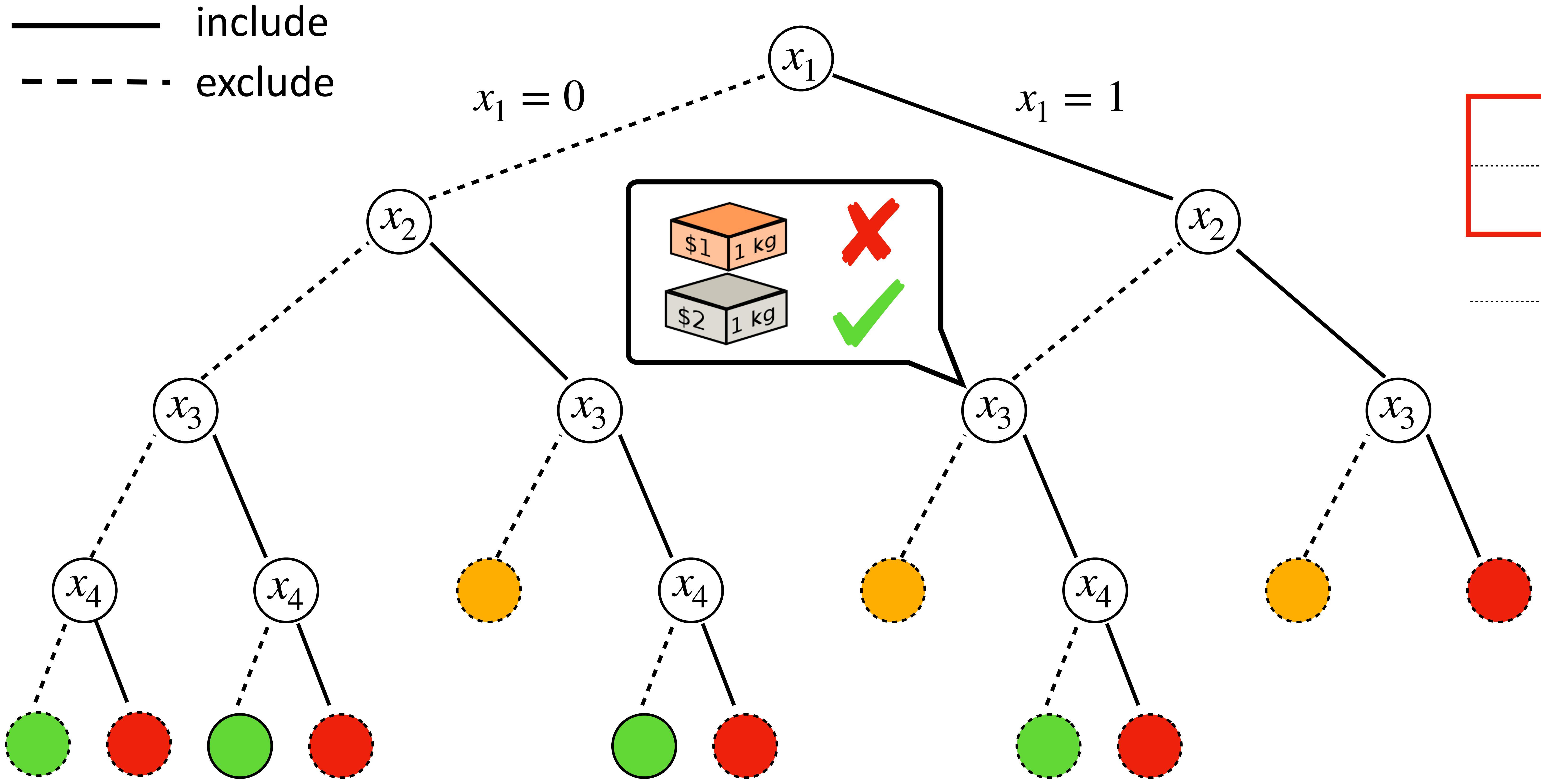


Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4



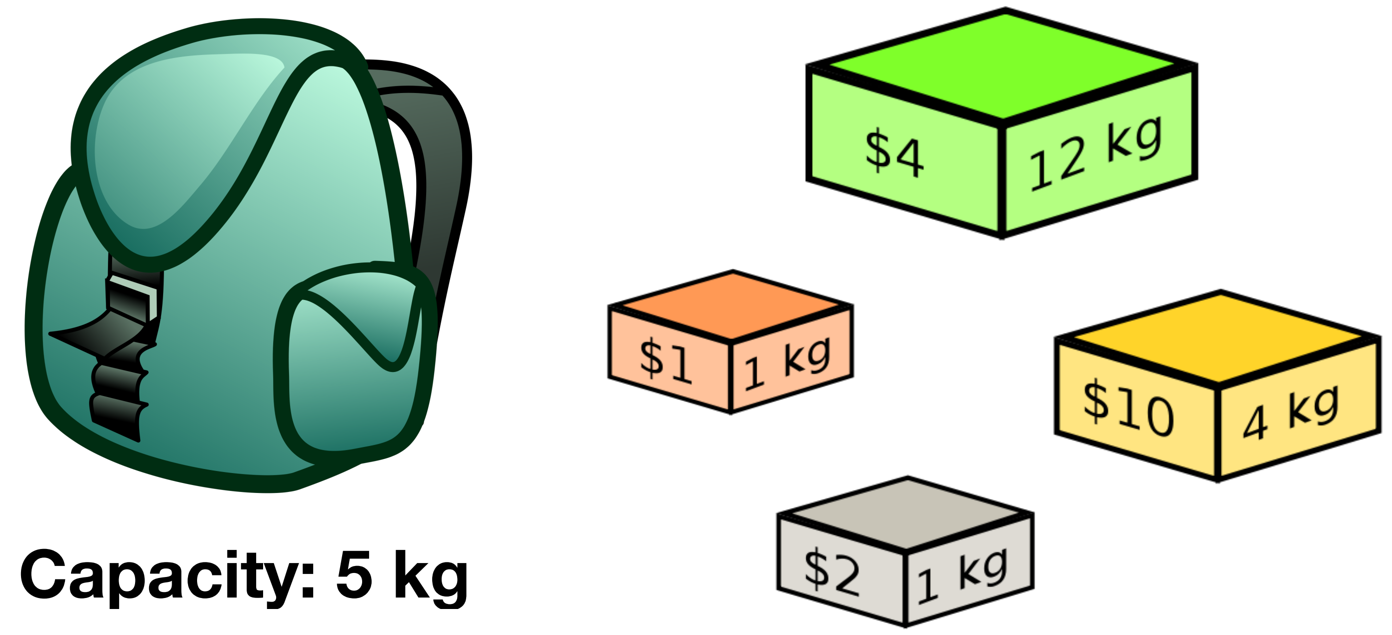
Dominance Breaking



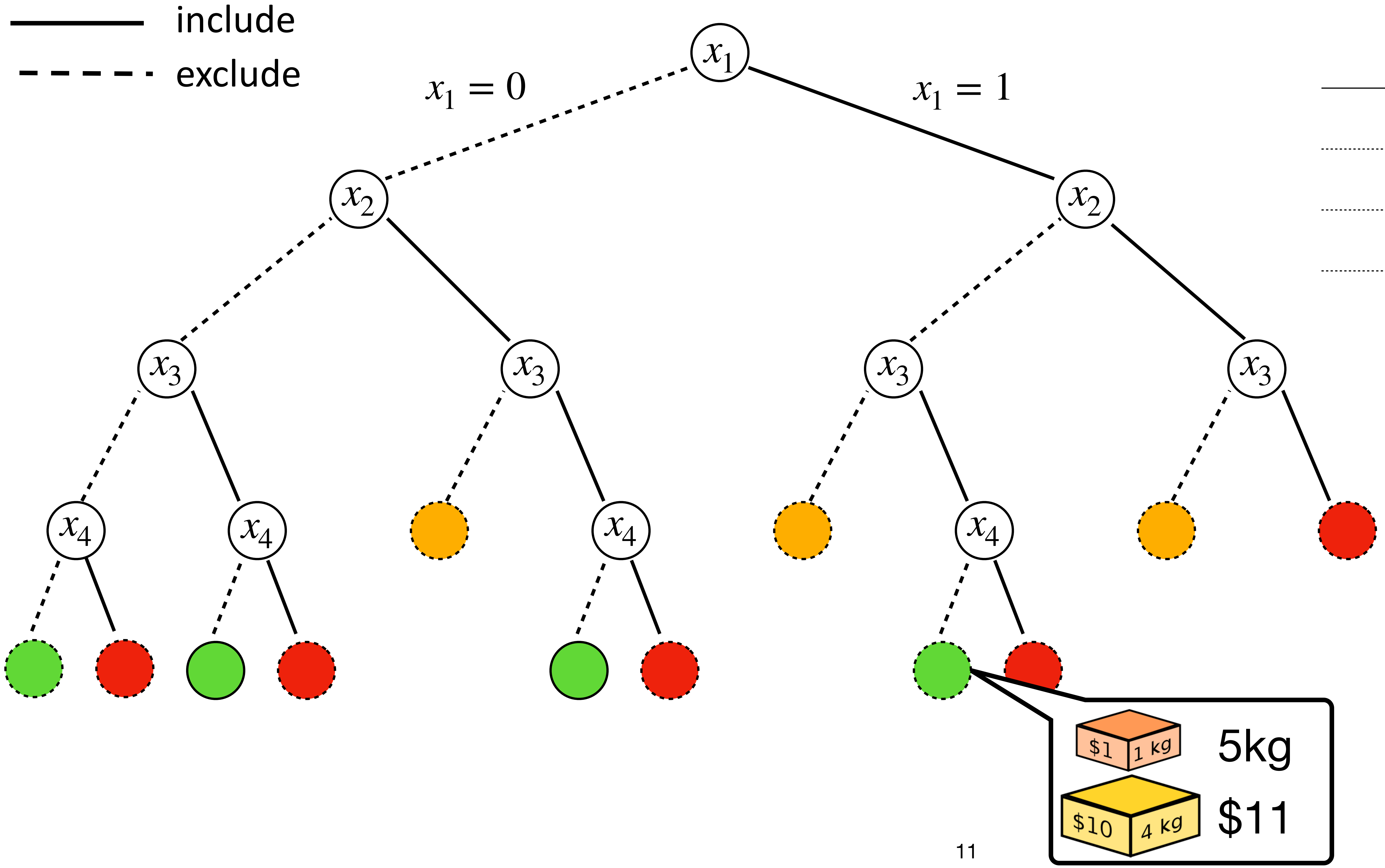


Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4

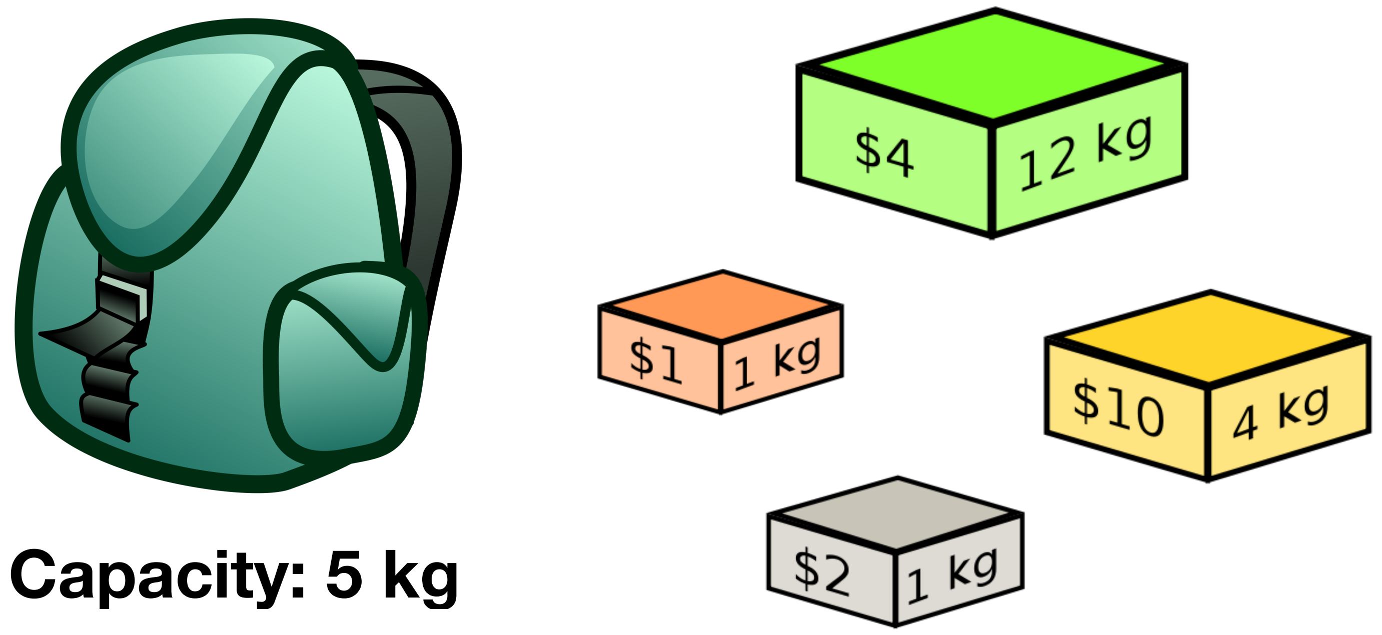
Dominance Breaking



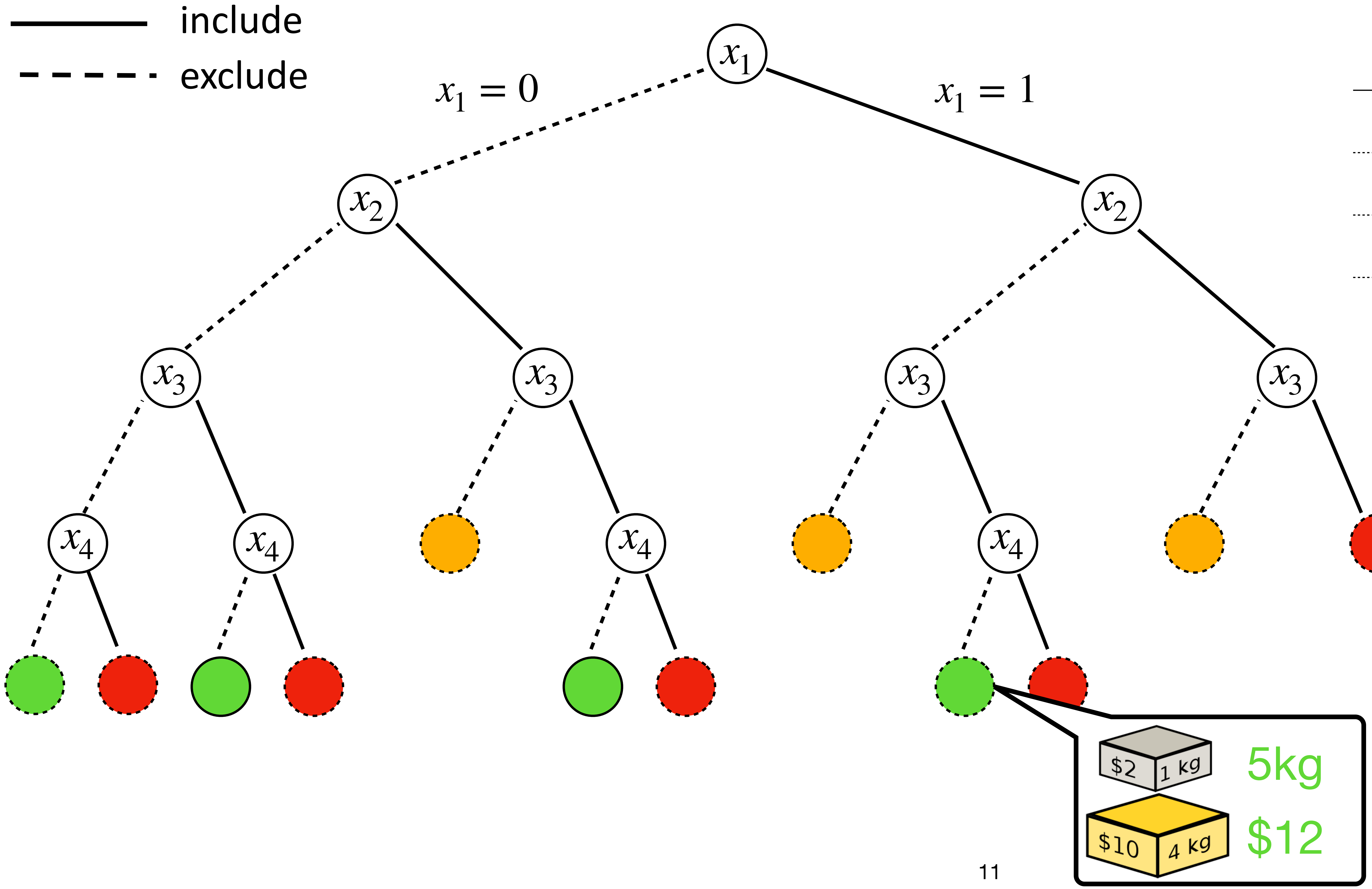
Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4



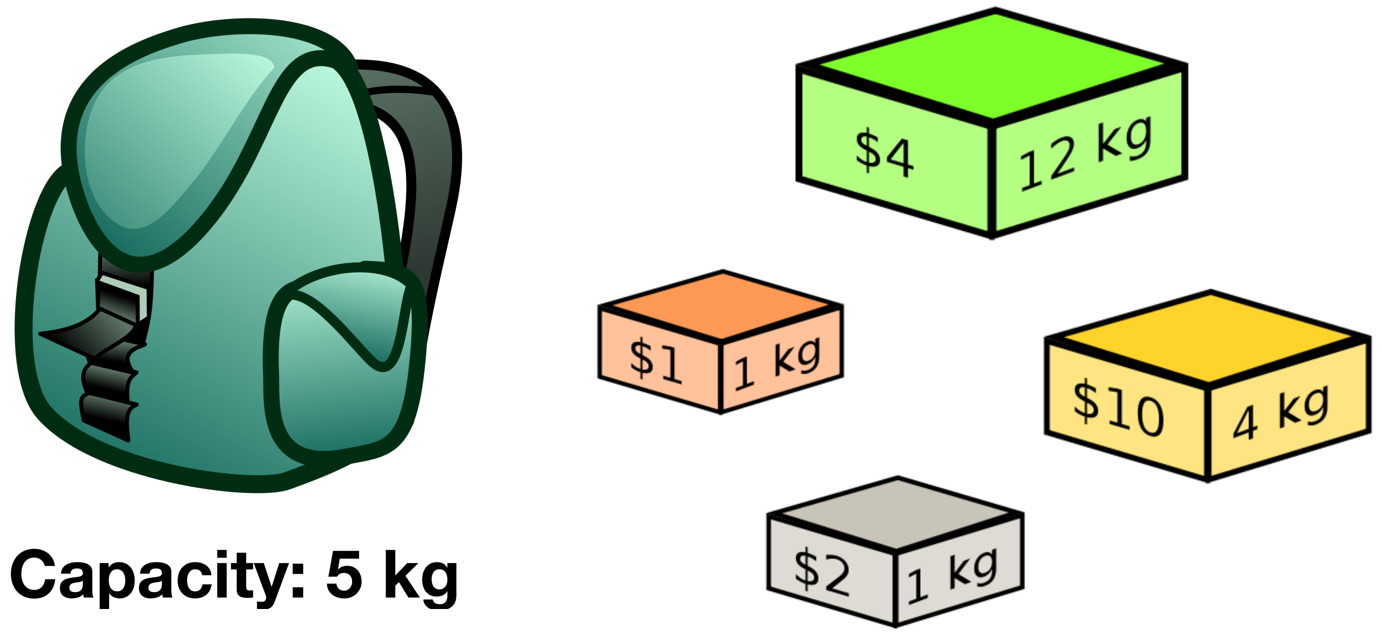
Dominance Breaking



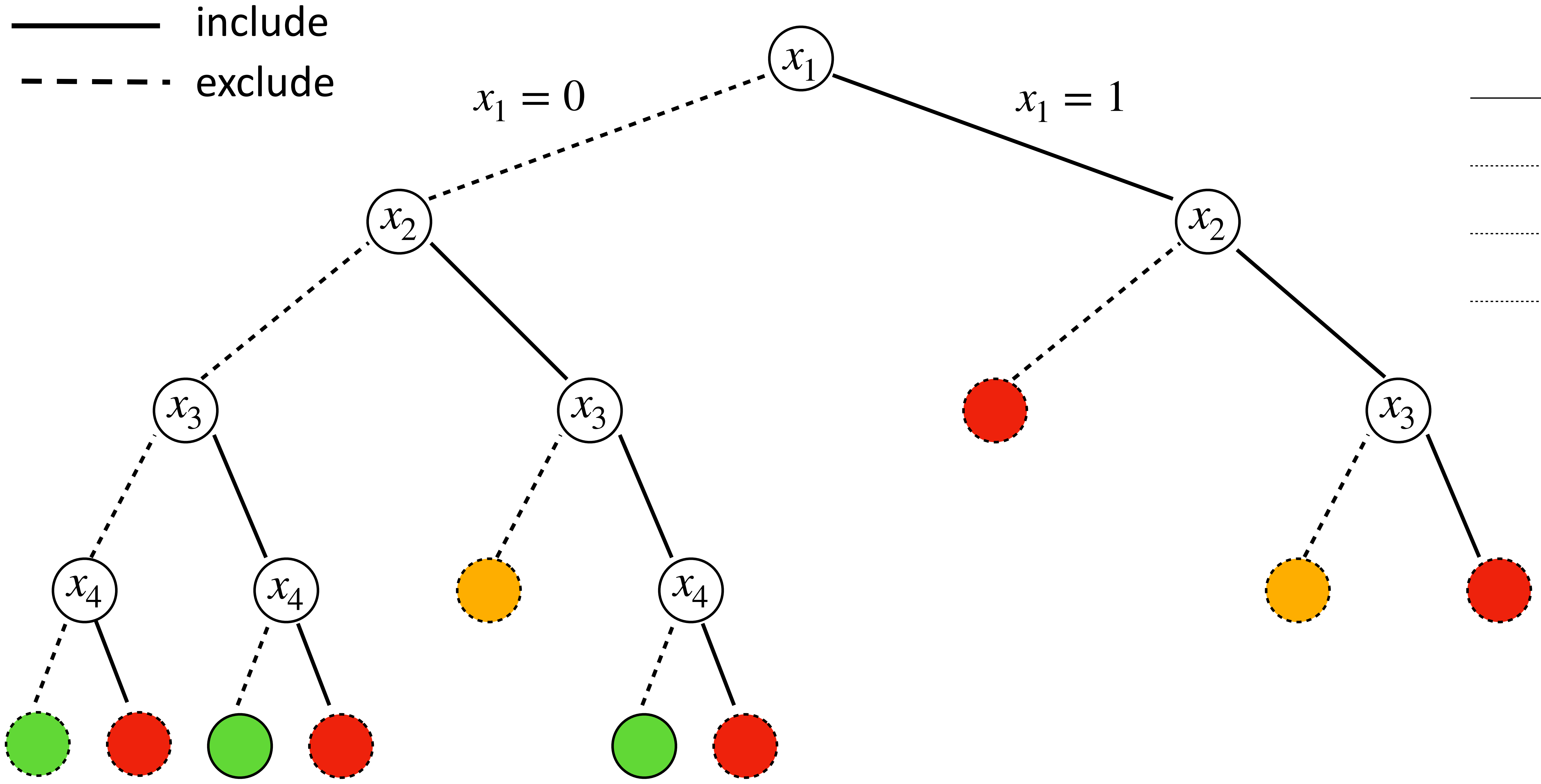
Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4



Dominance Breaking

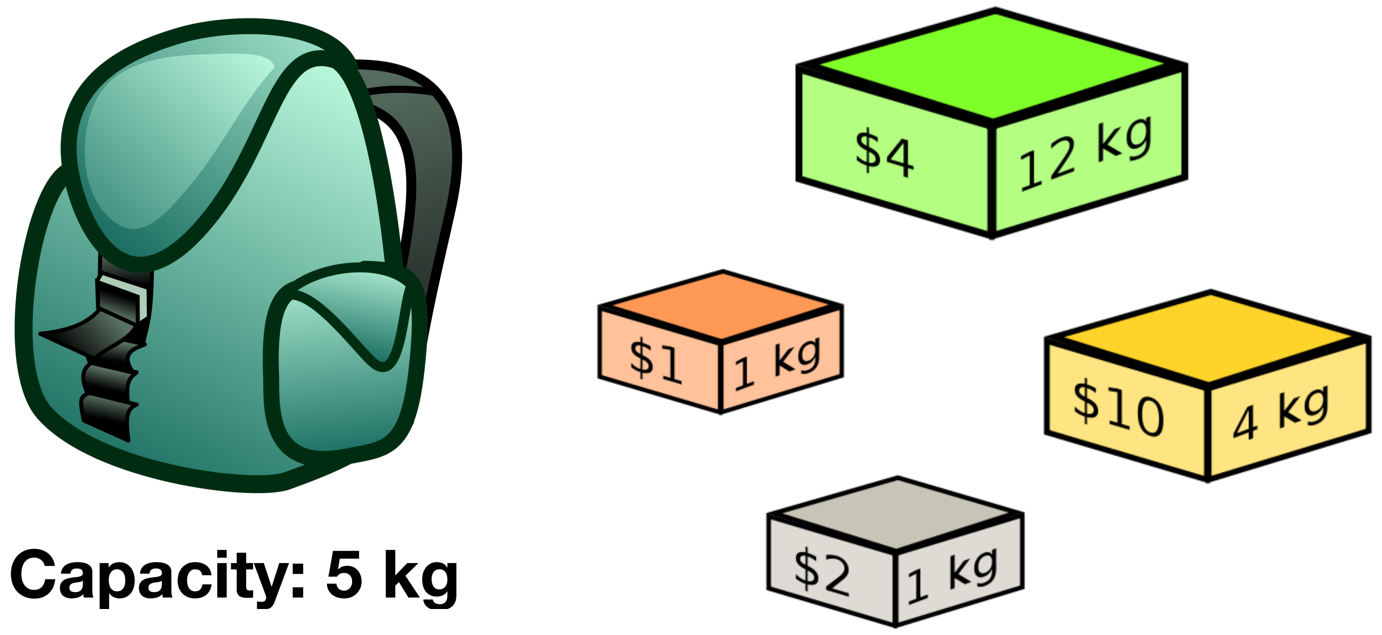


Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4



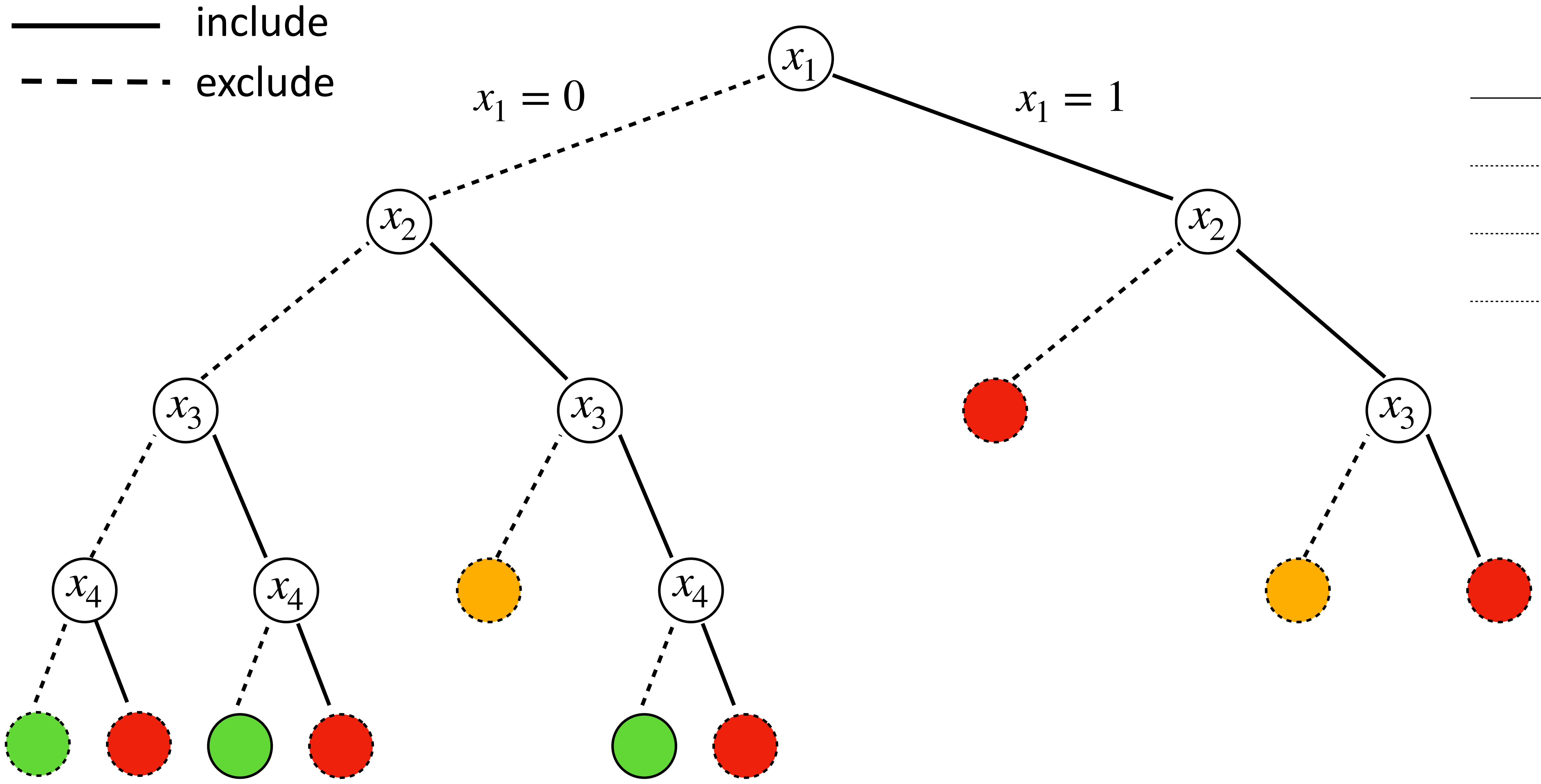
add constraint: $x_1 \leq x_2$

Dominance Breaking



Capacity: 5 kg

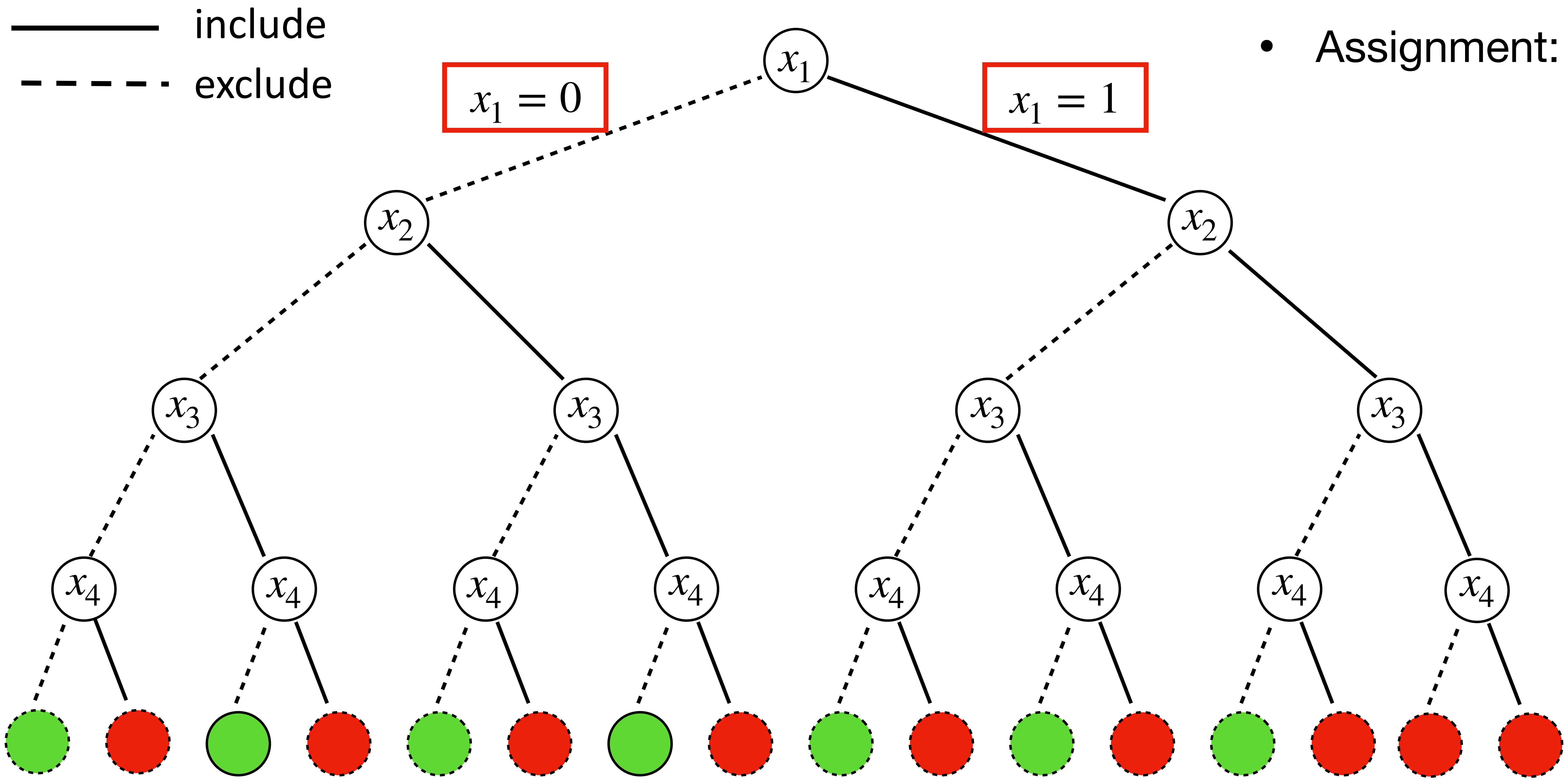
Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4



If $w_i \geq w_j \wedge v_i < v_j$,

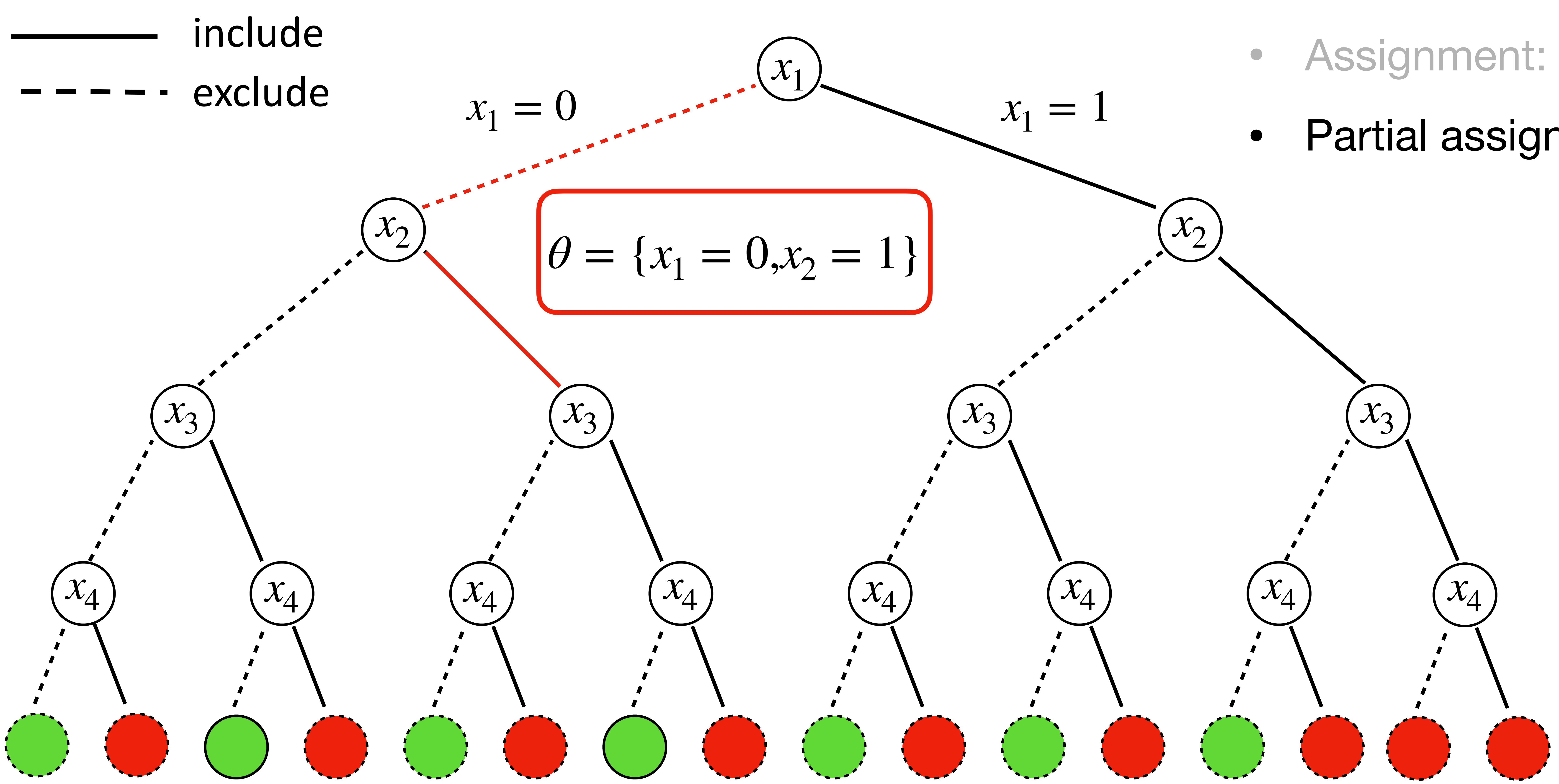
add constraint: $x_i \leq x_j$

Assignments



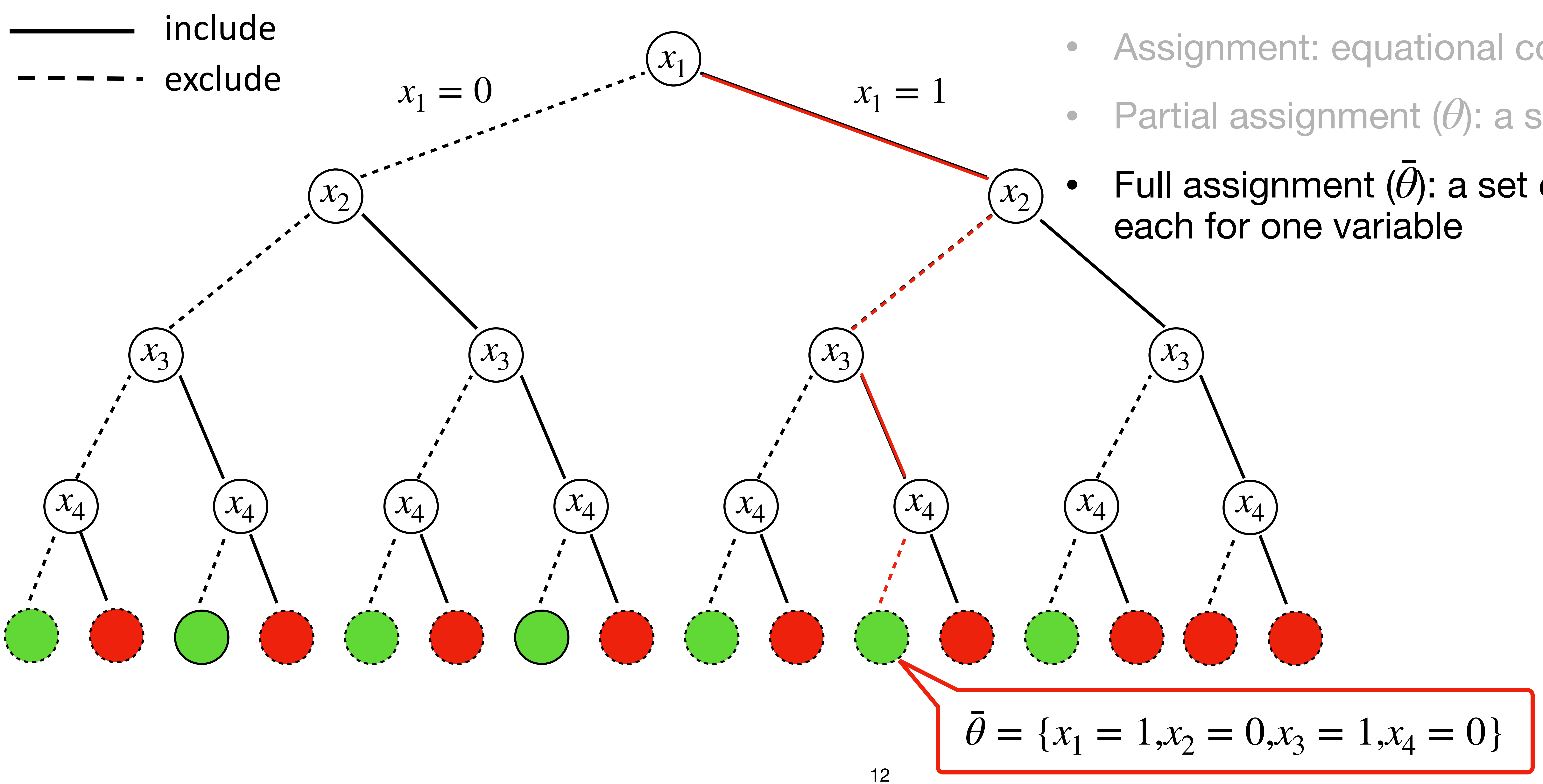
- Assignment: equational constraint

Assignments



- Assignment: equational constraint
- Partial assignment (θ): a set of assignments

Assignments



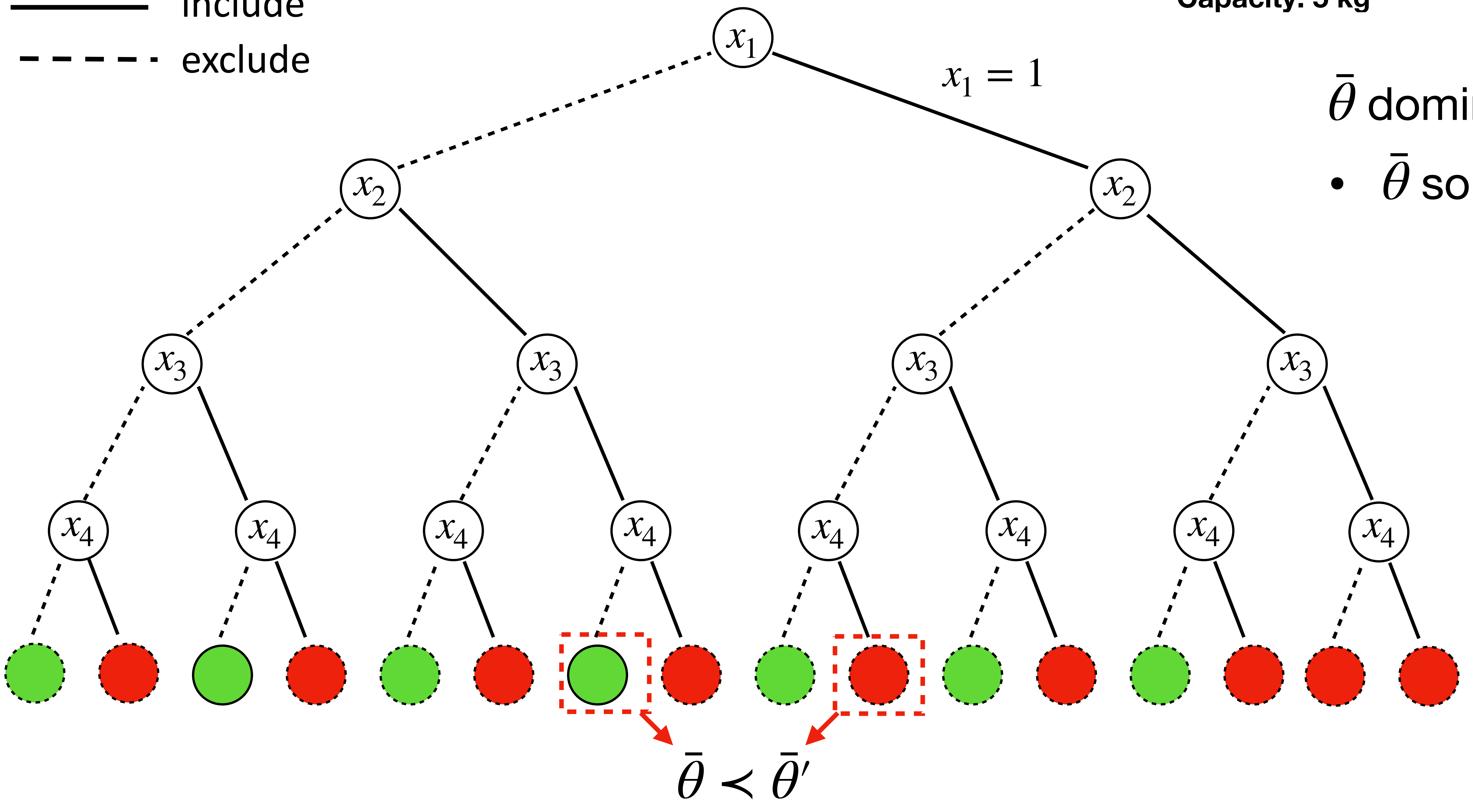
- Assignment: equational constraint
- Partial assignment (θ): a set of assignments
- Full assignment ($\bar{\theta}$): a set of assignments, each for one variable

Dominance Relations

(Chu and Stuckey 2012)

Capacity: 5 kg

Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4



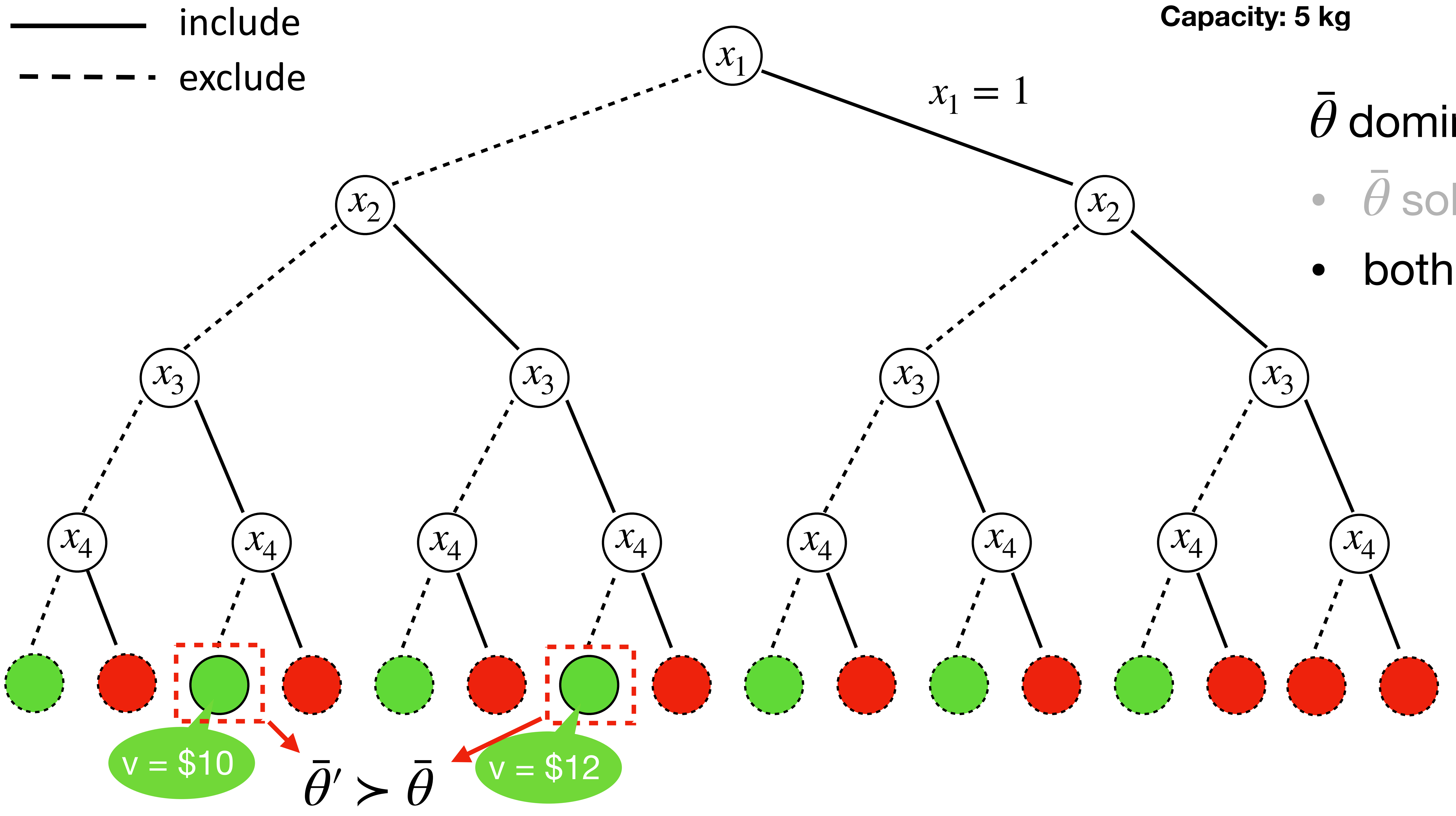
$\bar{\theta}$ dominates $\bar{\theta}'$ ($\bar{\theta} \prec \bar{\theta}'$):

- $\bar{\theta}$ solution, $\bar{\theta}'$ non-solution

Dominance Relations

(Chu and Stuckey 2012)

Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4



$\bar{\theta}$ dominates $\bar{\theta}'$ ($\bar{\theta} \prec \bar{\theta}'$):

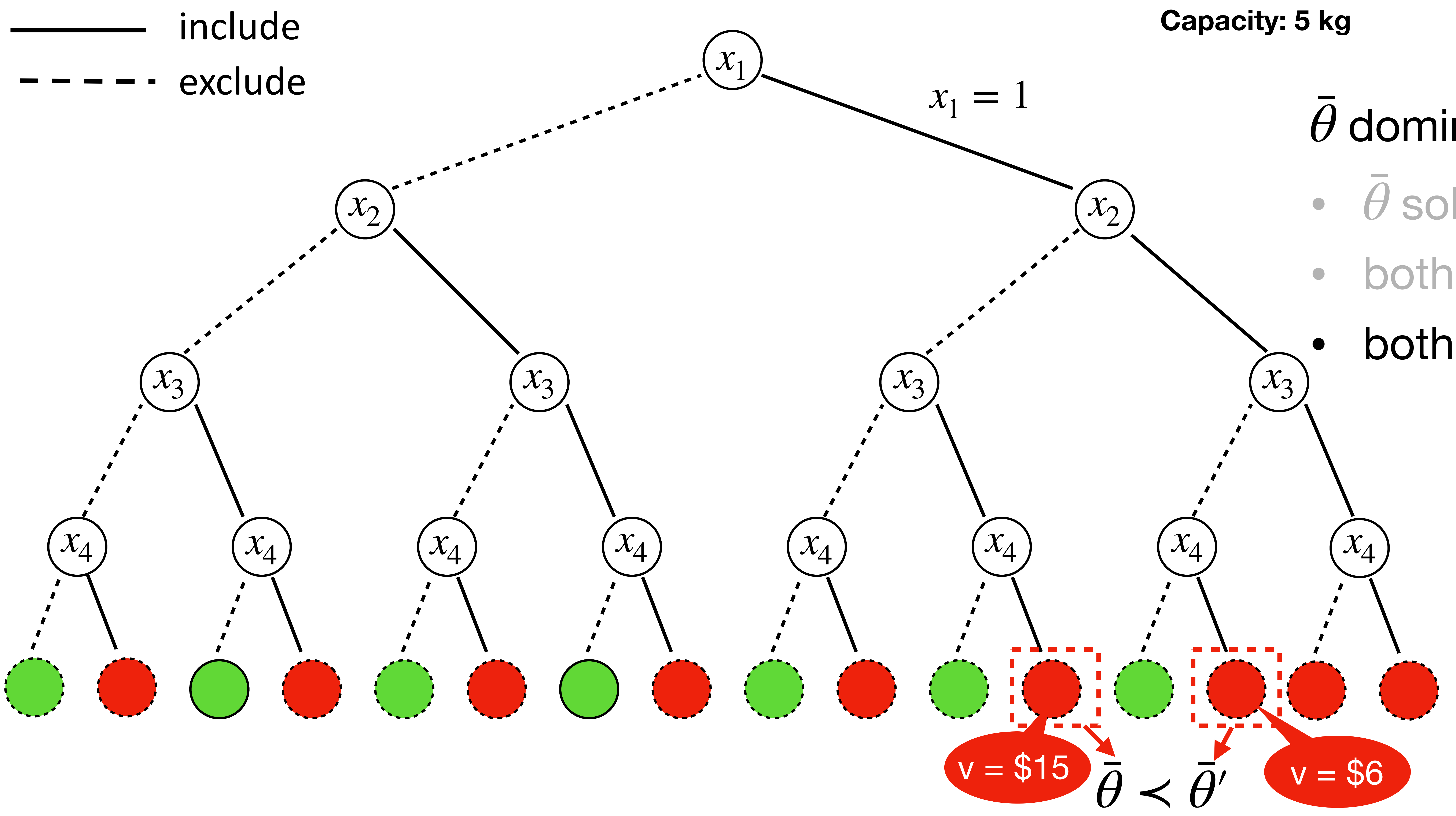
- $\bar{\theta}$ solution, $\bar{\theta}'$ non-solution
- both solutions, $f(\bar{\theta}) < f(\bar{\theta}')$

Dominance Relations

(Chu and Stuckey 2012)

Capacity: 5 kg

Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4



$\bar{\theta}$ dominates $\bar{\theta}'$ ($\bar{\theta} \prec \bar{\theta}'$):

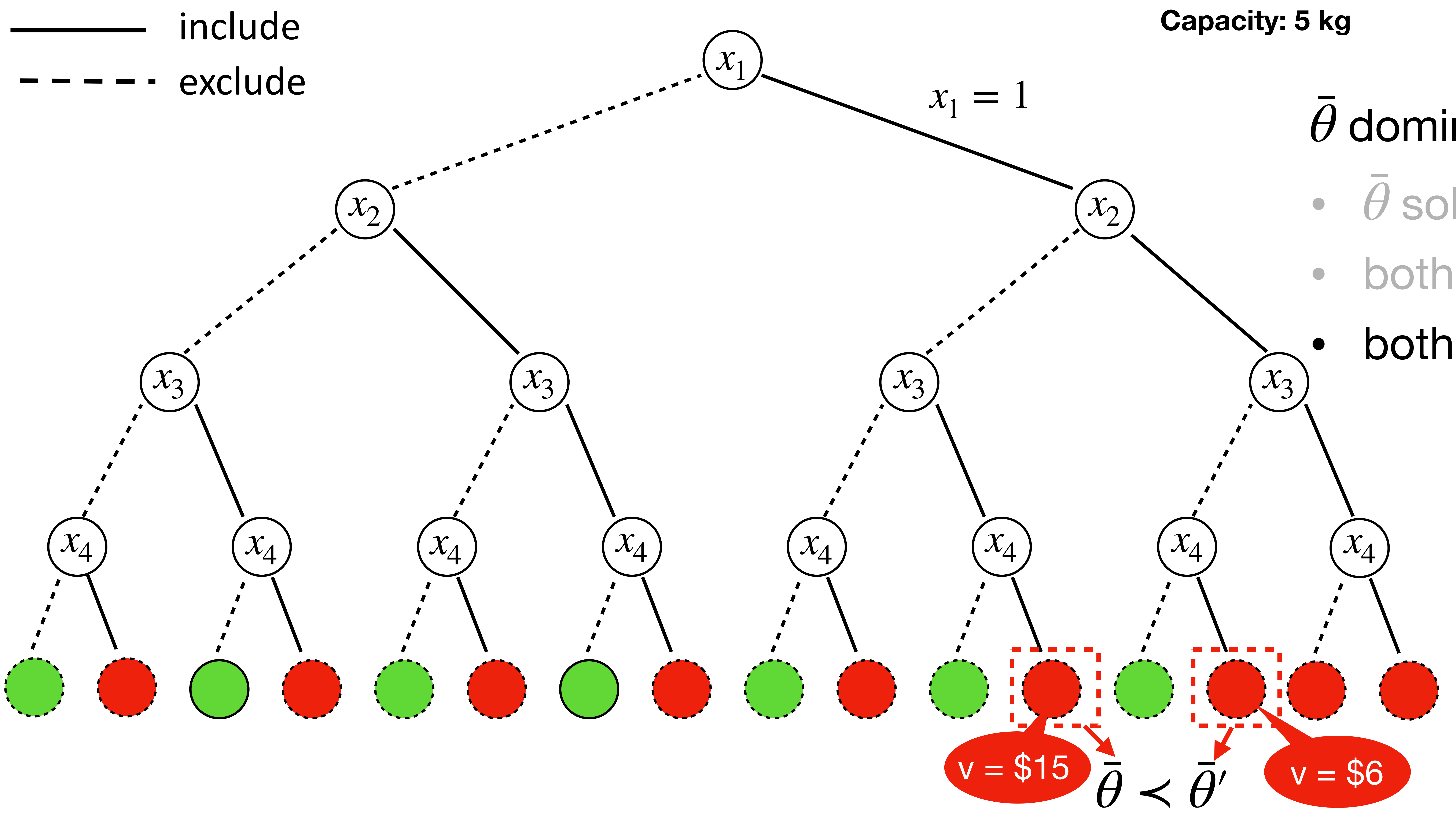
- $\bar{\theta}$ solution, $\bar{\theta}'$ non-solution
- both solutions, $f(\bar{\theta}) < f(\bar{\theta}')$
- both non-solutions, $f(\bar{\theta}) < f(\bar{\theta}')$

Dominance Relations

(Chu and Stuckey 2012)

Capacity: 5 kg

Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4



$\bar{\theta}$ dominates $\bar{\theta}'$ ($\bar{\theta} \prec \bar{\theta}'$):

- $\bar{\theta}$ solution, $\bar{\theta}'$ non-solution
- both solutions, $f(\bar{\theta}) < f(\bar{\theta}')$
- both non-solutions, $f(\bar{\theta}) < f(\bar{\theta}')$

Theorem:
All dominated full assignments can be pruned!

Dominance Relations

dominance relations
between constraints

$$c \prec c'$$

$$\bar{\theta} \prec \bar{\theta}'$$

dominance relations between
full assignments

$$\theta \prec \theta'$$

dominance relations between
partial assignments

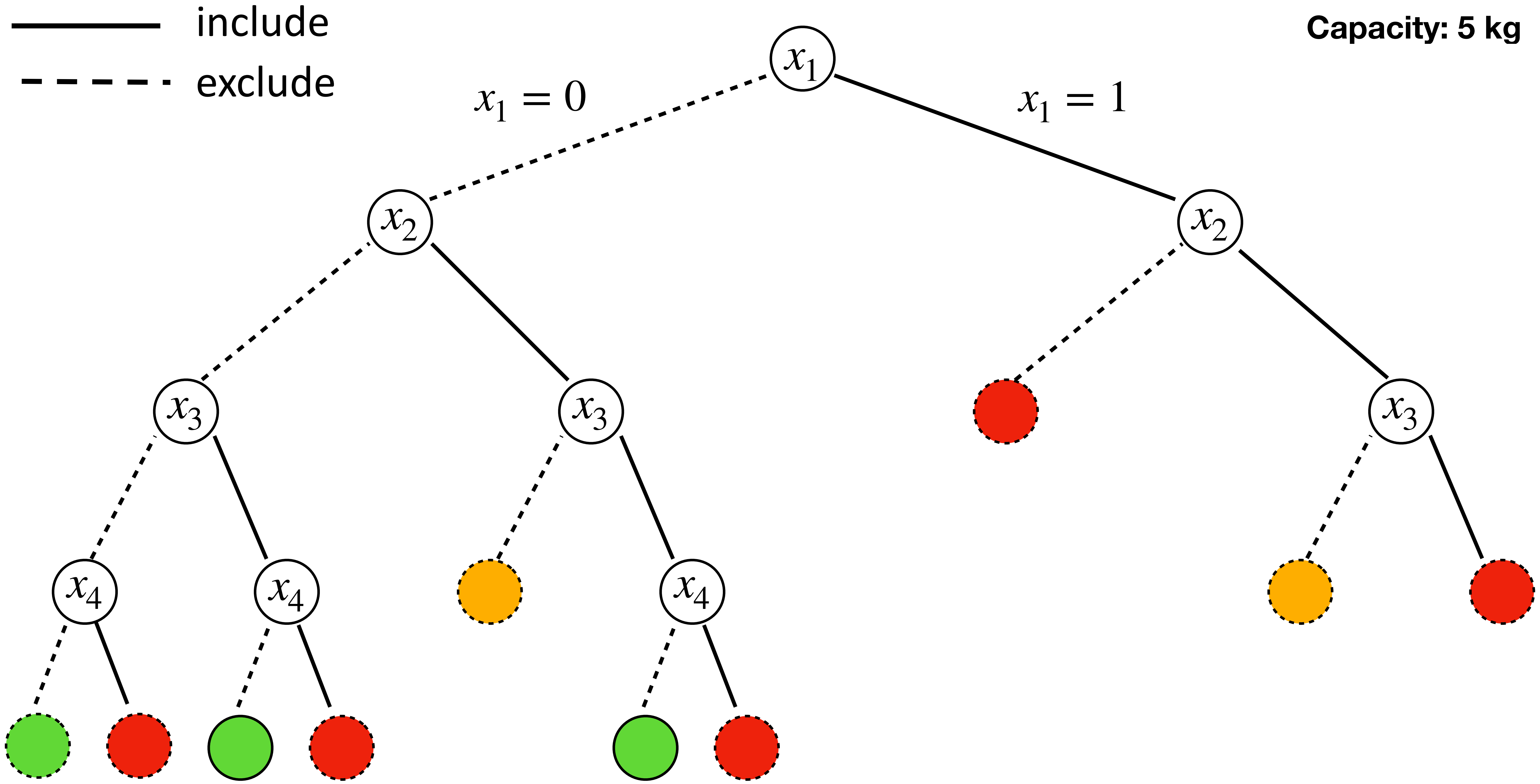
Agenda

- Background
- Automatic Dominance Breaking
- Experimental Results
- Q & A

Dominance Relations

Capacity: 5 kg

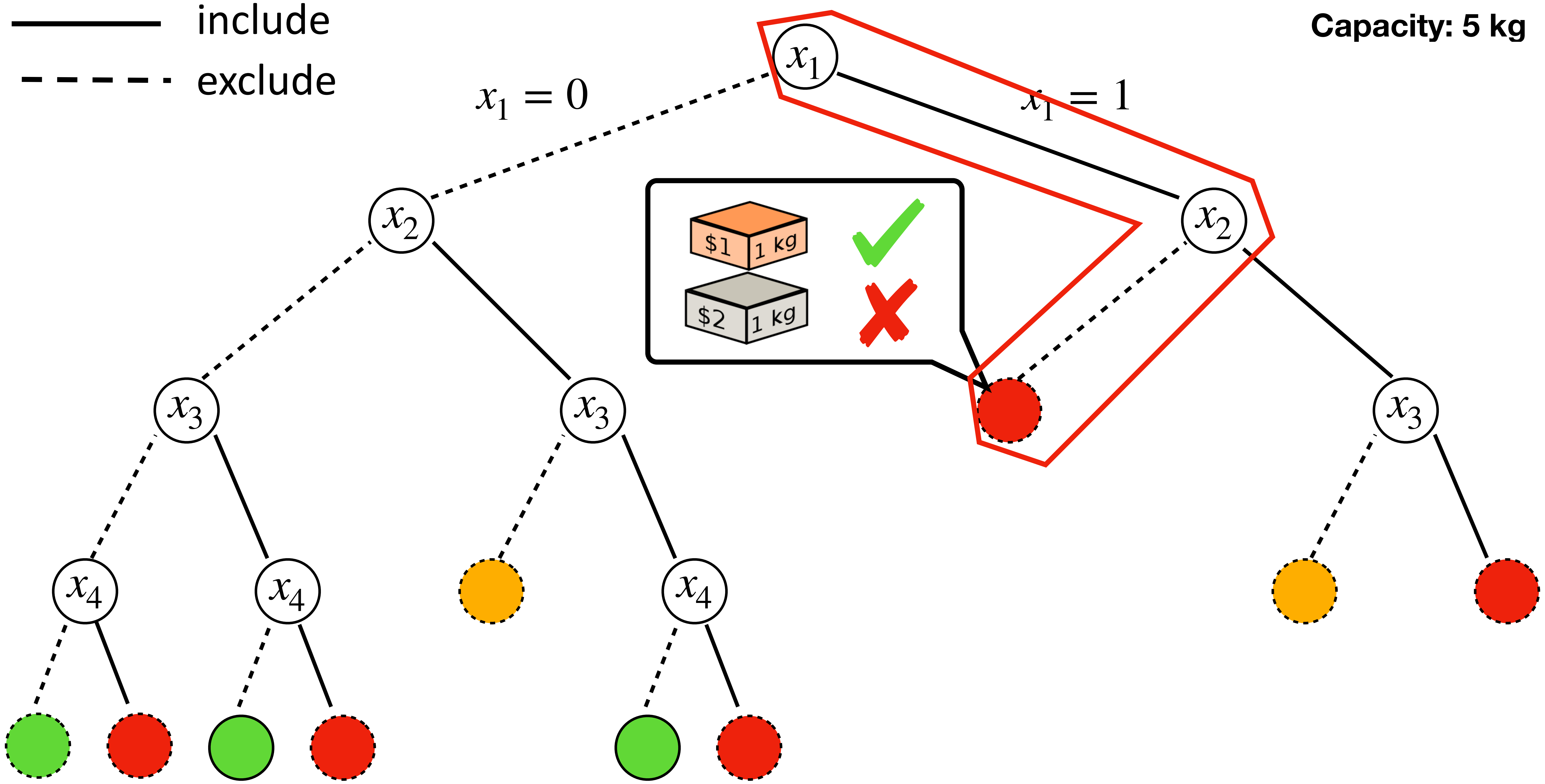
Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4



Dominance Relations

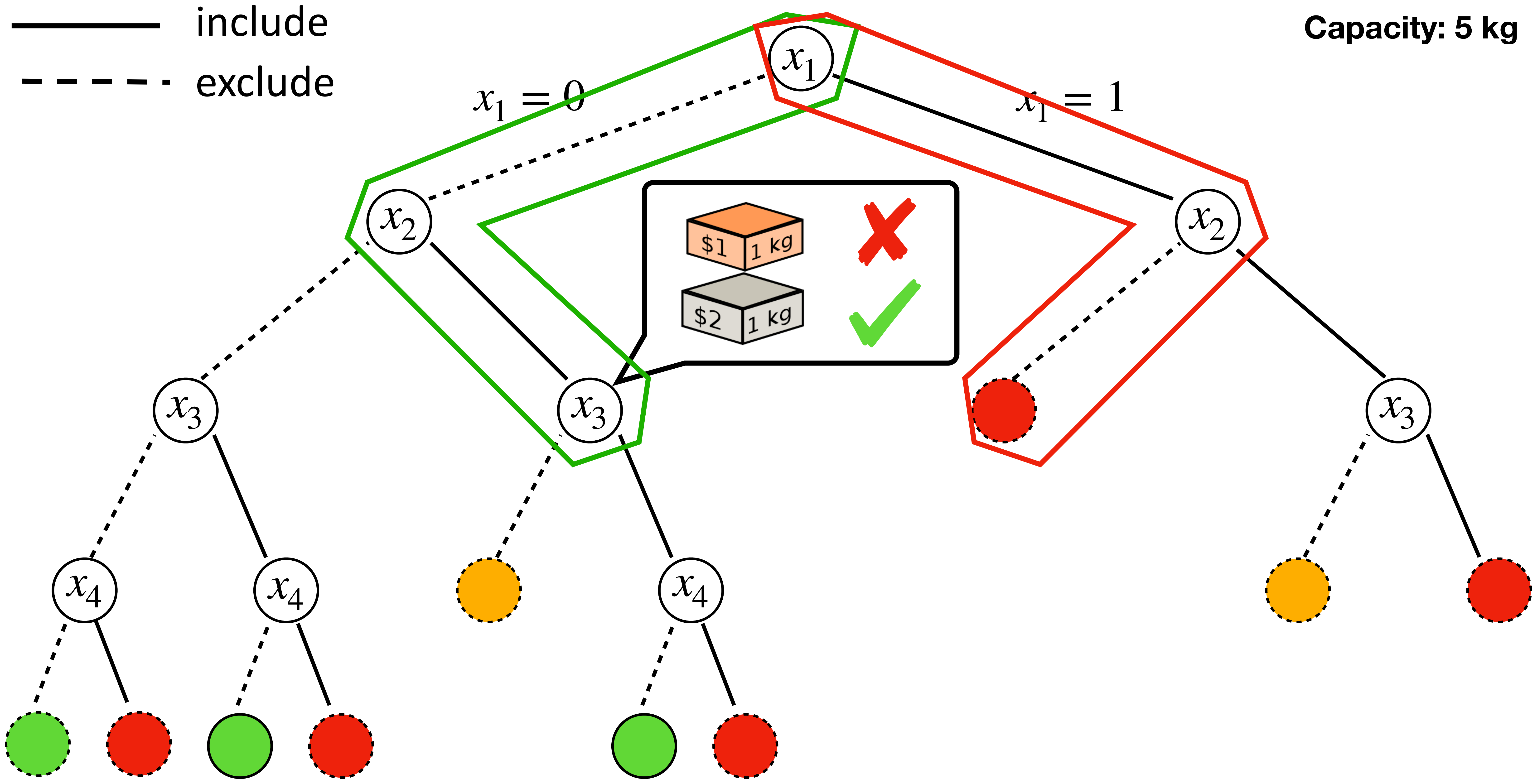
Capacity: 5 kg

Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4



Dominance Relations

Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4

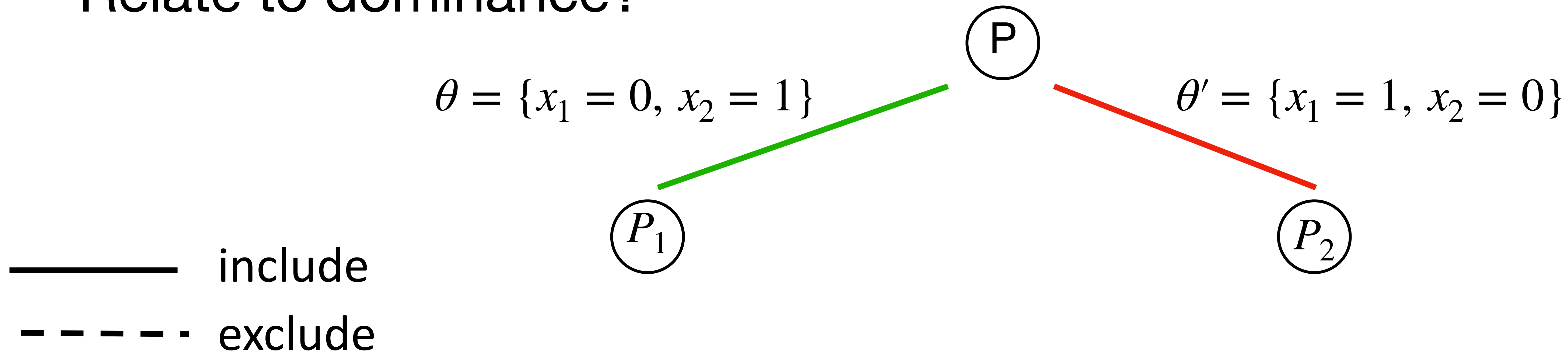


Dominance Relations

Capacity: 5 kg

Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4

Relate to dominance?

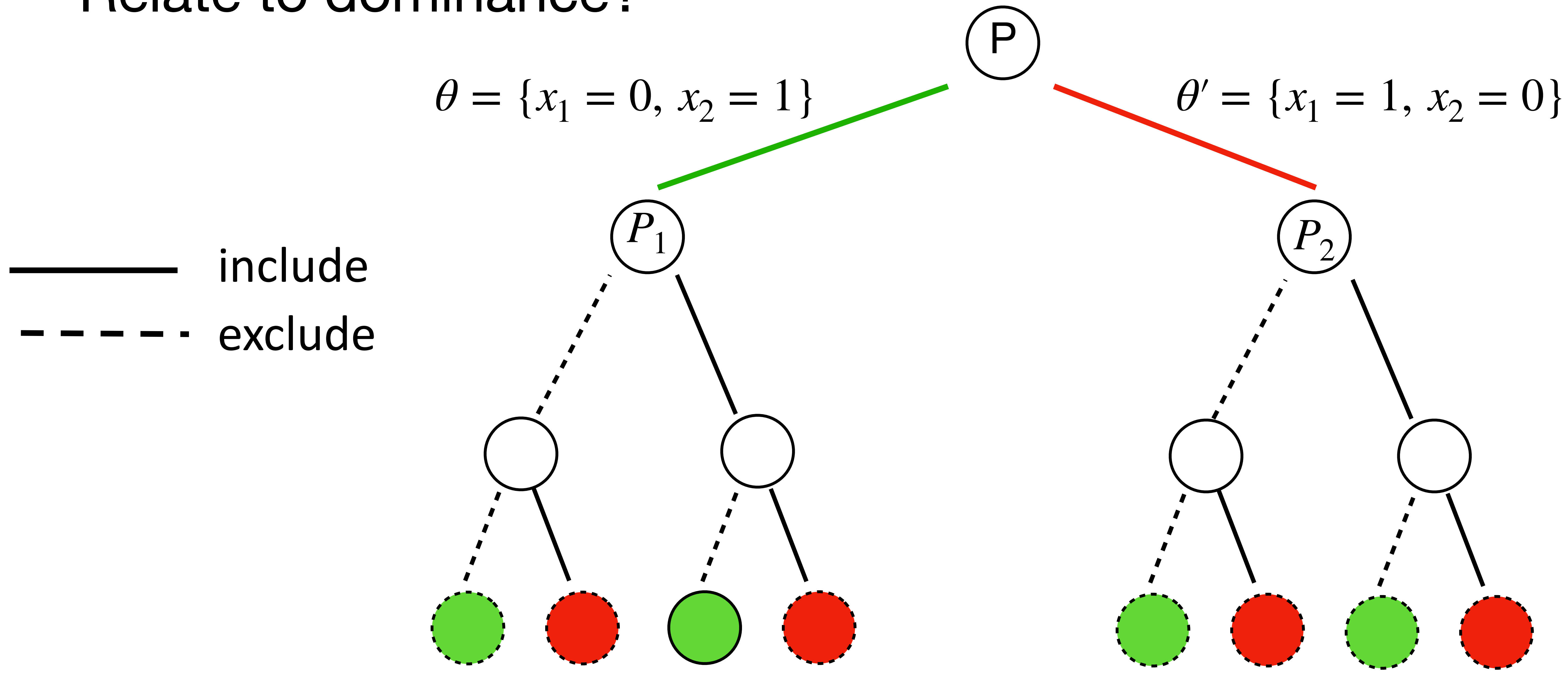


Dominance Relations

Capacity: 5 kg

Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4

Relate to dominance?

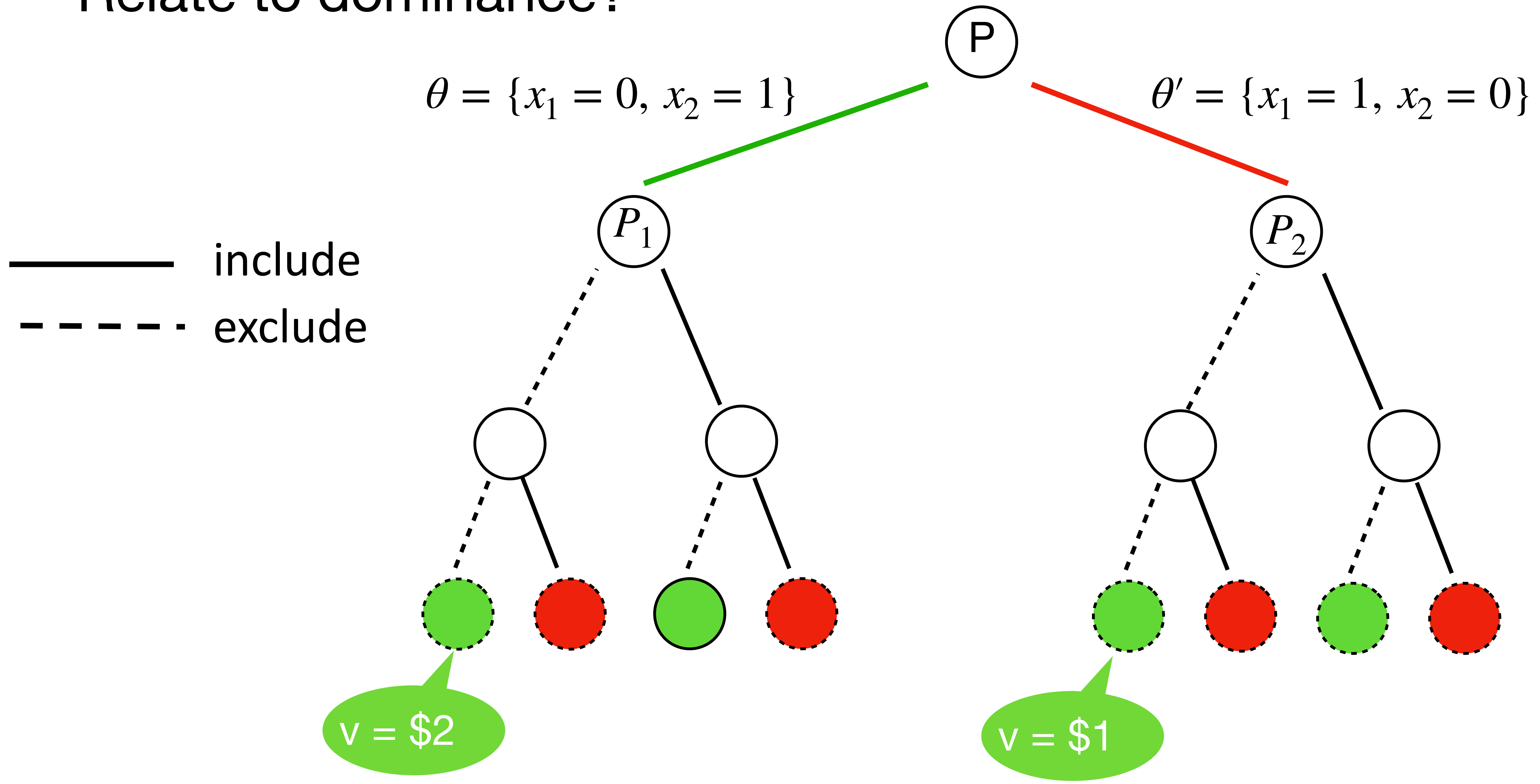


Dominance Relations

Capacity: 5 kg

Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4

Relate to dominance?



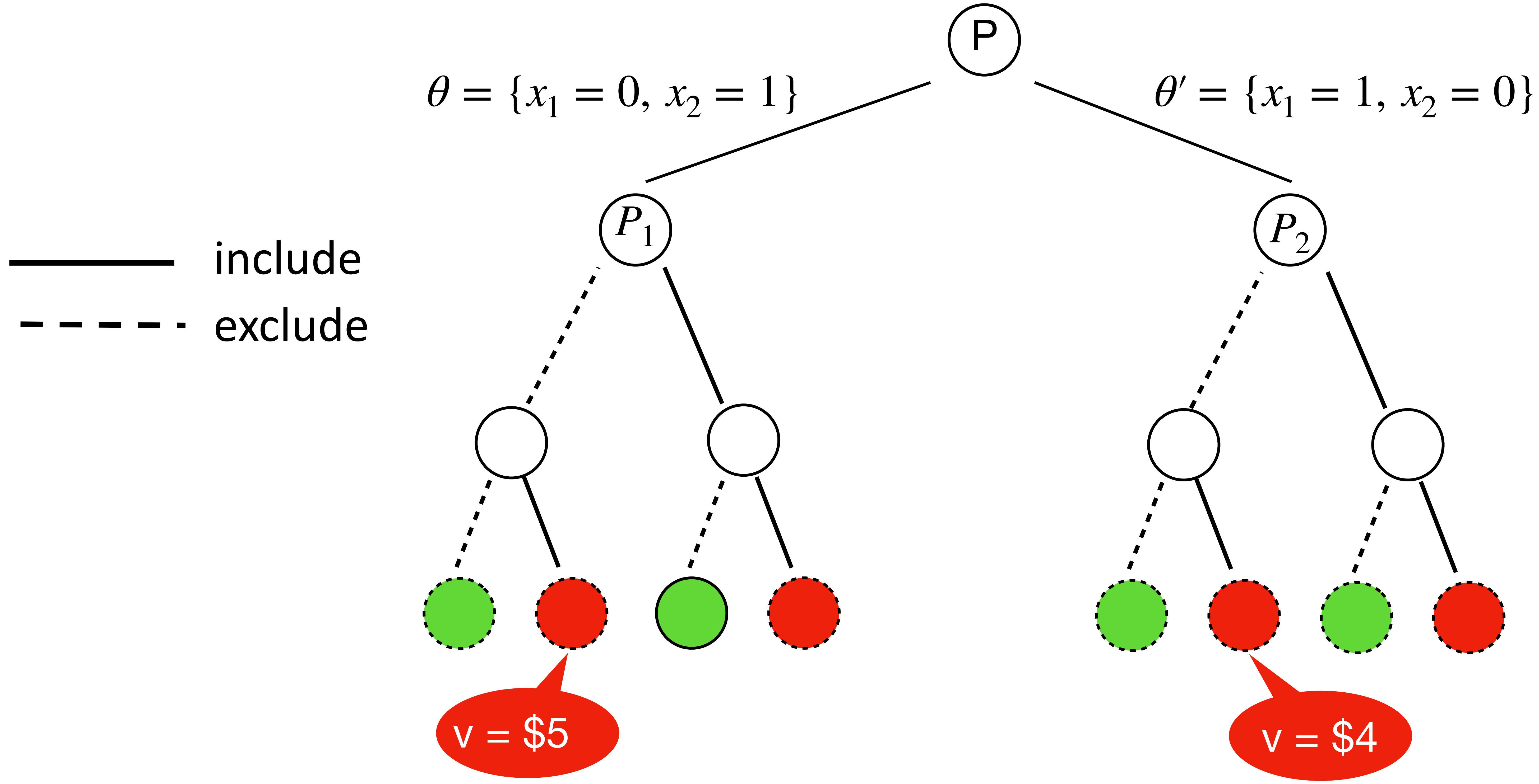
$\bar{\theta}$ dominates $\bar{\theta}'$ ($\bar{\theta} \prec \bar{\theta}'$):

- $\bar{\theta}$ solution, $\bar{\theta}'$ non-solution
- both solutions, $f(\bar{\theta}) < f(\bar{\theta}')$
- both non-solutions, $f(\bar{\theta}) < f(\bar{\theta}')$

Dominance Relations

Capacity: 5 kg

Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4

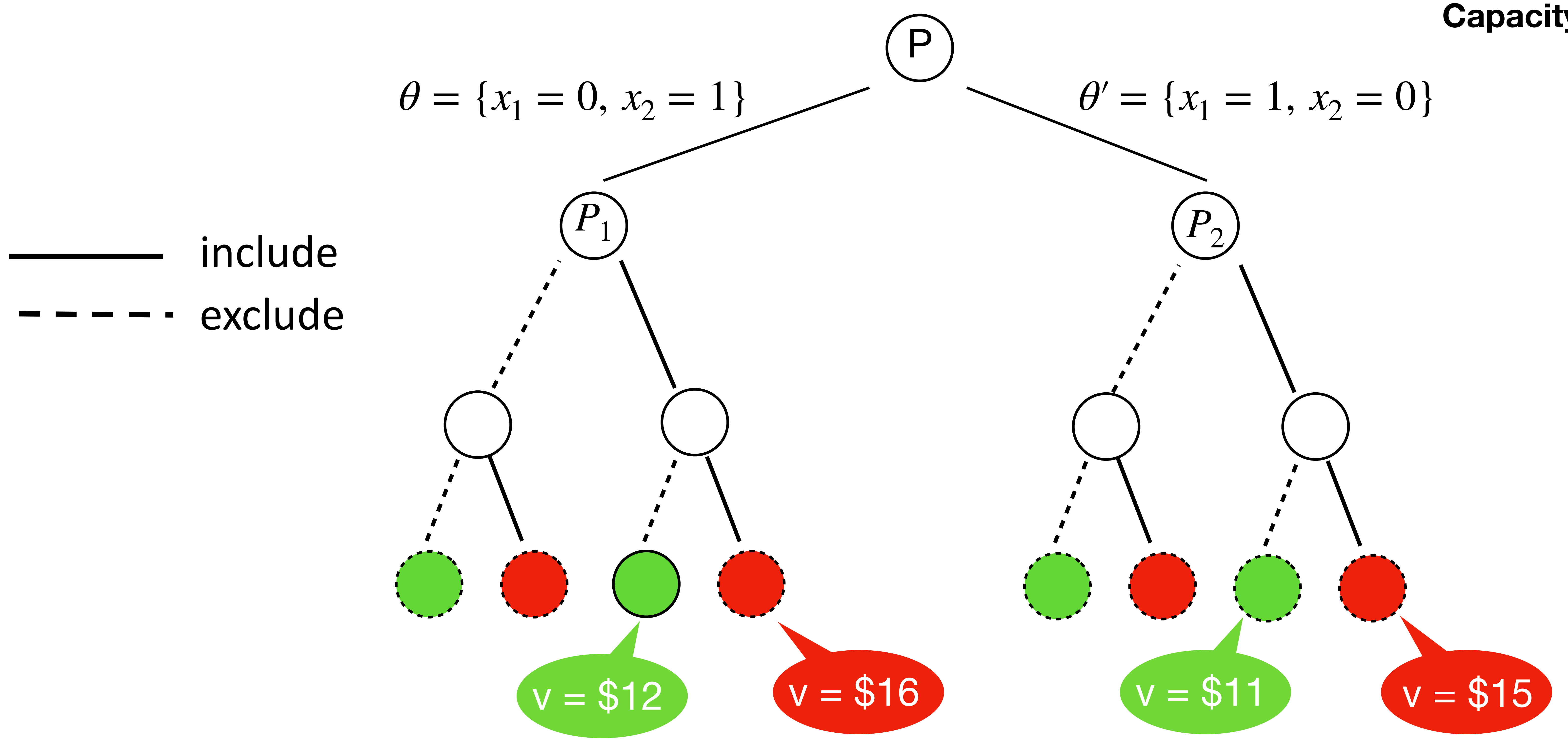


$\bar{\theta}$ dominates $\bar{\theta}'$ ($\bar{\theta} \prec \bar{\theta}'$):

- $\bar{\theta}$ solution, $\bar{\theta}'$ non-solution
- both solutions, $f(\bar{\theta}) < f(\bar{\theta}')$
- both non-solutions, $f(\bar{\theta}) < f(\bar{\theta}')$

Dominance Relations

Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4



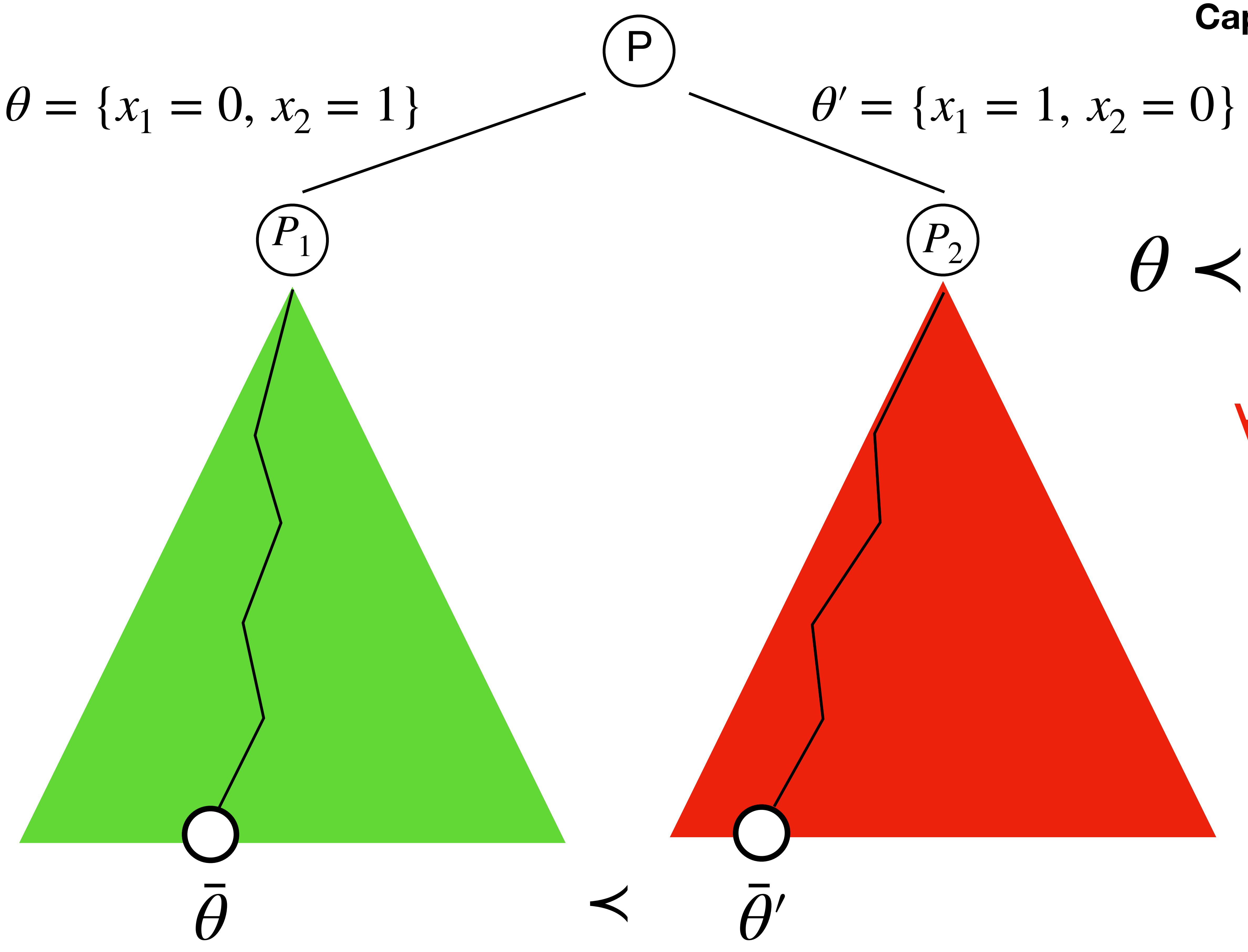
$\bar{\theta}$ dominates $\bar{\theta}'$ ($\bar{\theta} \prec \bar{\theta}'$):

- $\bar{\theta}$ solution, $\bar{\theta}'$ non-solution
- both solutions, $f(\bar{\theta}) < f(\bar{\theta}')$
- both non-solutions, $f(\bar{\theta}) < f(\bar{\theta}')$

Dominance Relations

Capacity: 5 kg

Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4



$\theta \prec \theta'$ w.r.t. P:

$$\forall \bar{\theta}' \in \Theta^{P \cup \theta'}, \exists \bar{\theta} \in \Theta^{P \cup \theta}, \bar{\theta} \prec \bar{\theta}'$$

the set of all full assignments

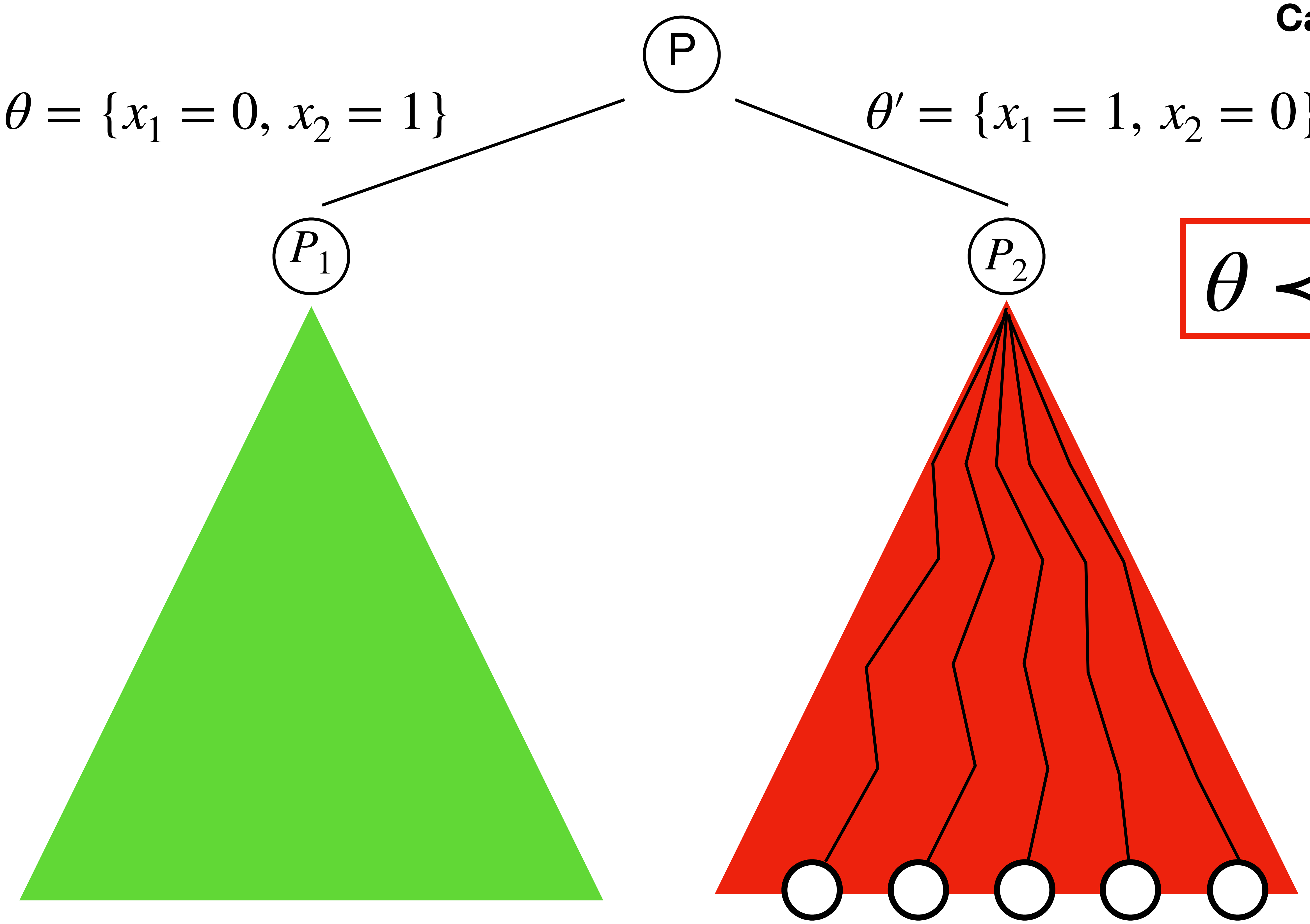
Dominance Relations

Capacity: 5 kg

Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4

$\theta = \{x_1 = 0,\ x_2 = 1\}$

$\theta' = \{x_1 = 1,\ x_2 = 0\}$



only defined when $var(\theta) = var(\theta')$

$\theta \prec \theta'$ w.r.t. P:

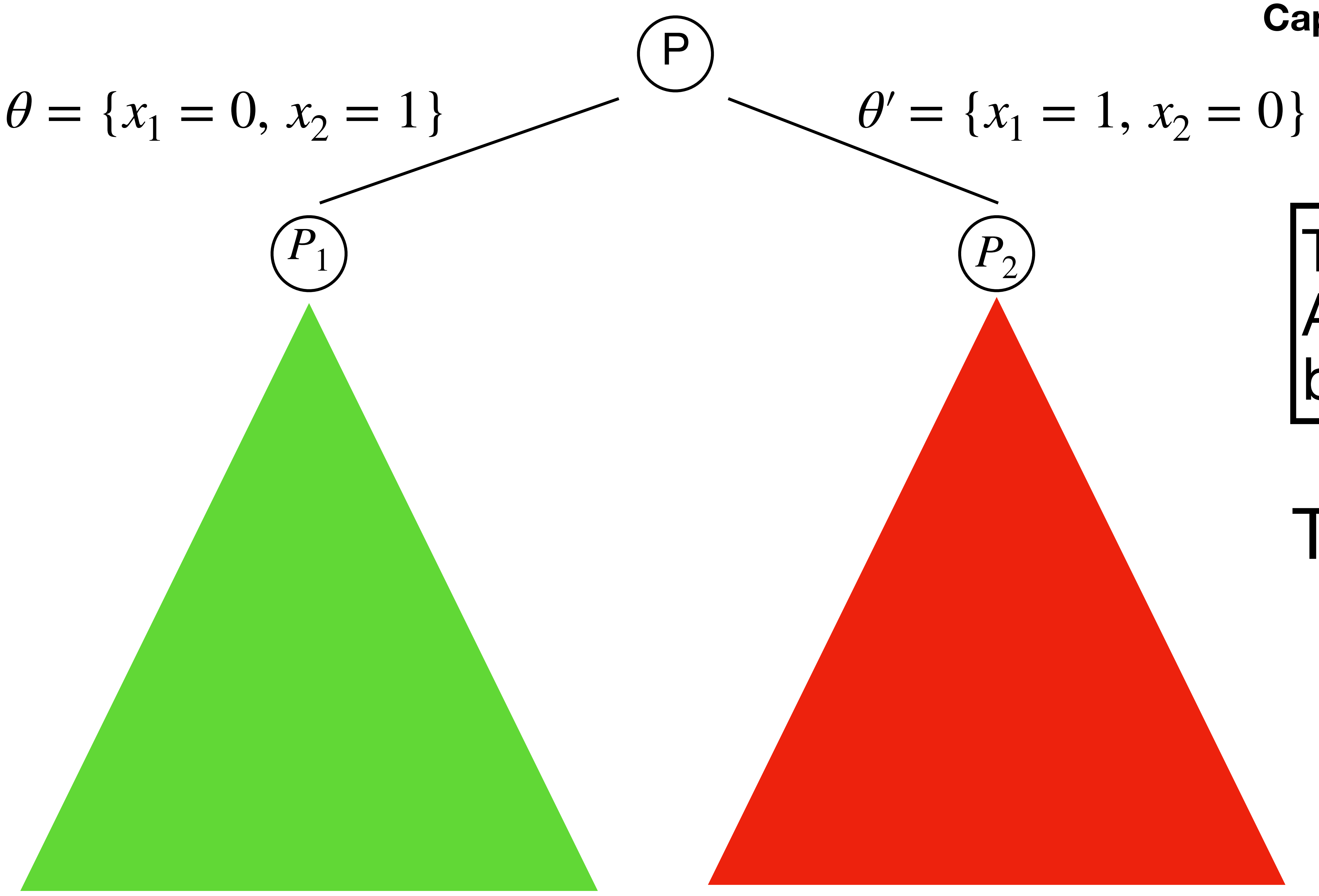
$\forall \bar{\theta}' \in \Theta^{P \cup \theta'}, \exists \bar{\theta} \in \Theta^{P \cup \theta}, \bar{\theta} \prec \bar{\theta}'$

the set of all full assignments

Dominance Breaking

Capacity: 5 kg

Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4



Theorem: (Chu and Stuckey 2012)
All dominated full assignments can be pruned!

Theorem:

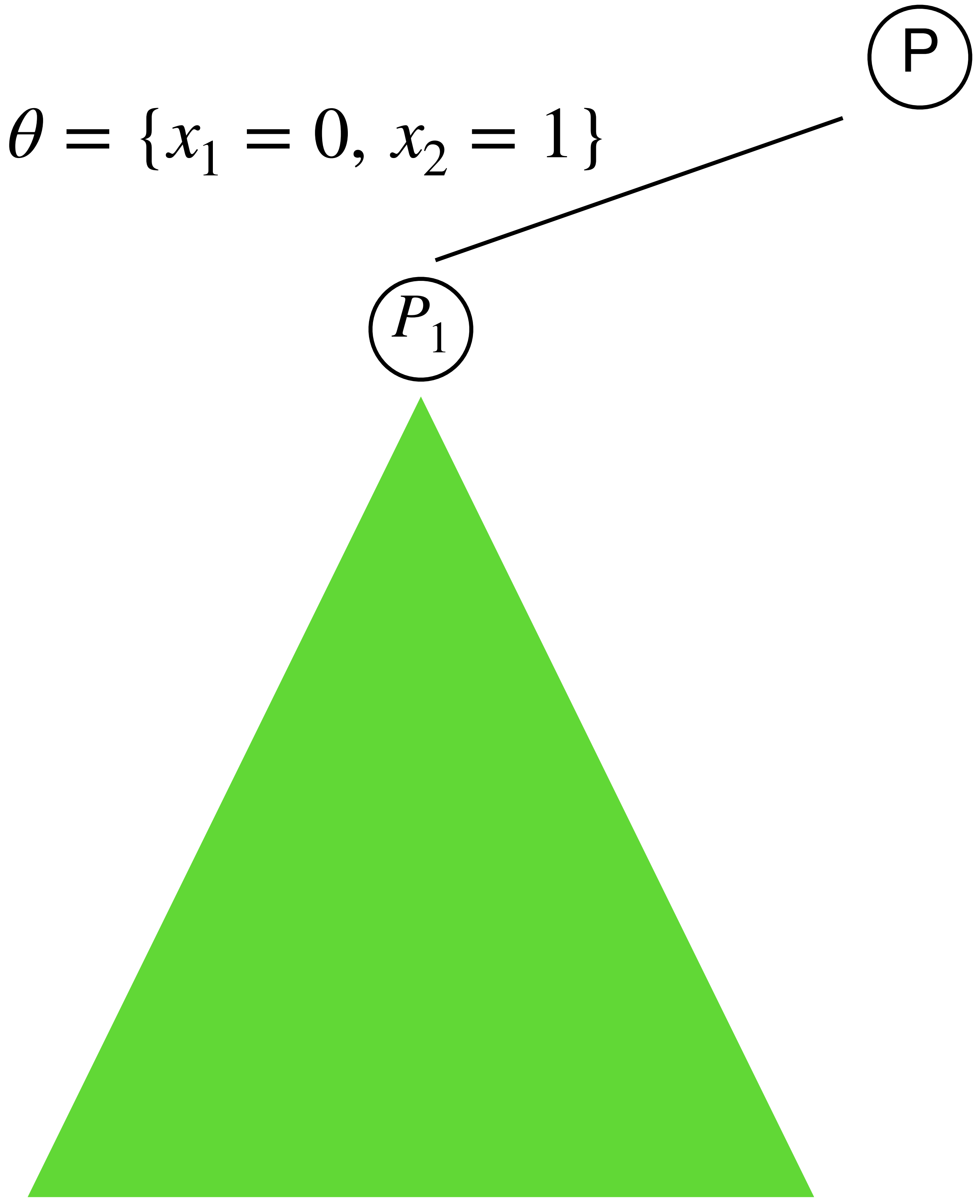
$$\theta \prec \theta' \text{ w.r.t. } P$$

$$\implies P \equiv (P \cup \neg\theta')$$

Dominance Breaking

Capacity: 5 kg

Item	Weight	Value
1	1kg	\$1
2	1kg	\$2
3	4kg	\$10
4	12kg	\$4



Theorem: (Chu and Stuckey 2012)
All dominated full assignments can be pruned!

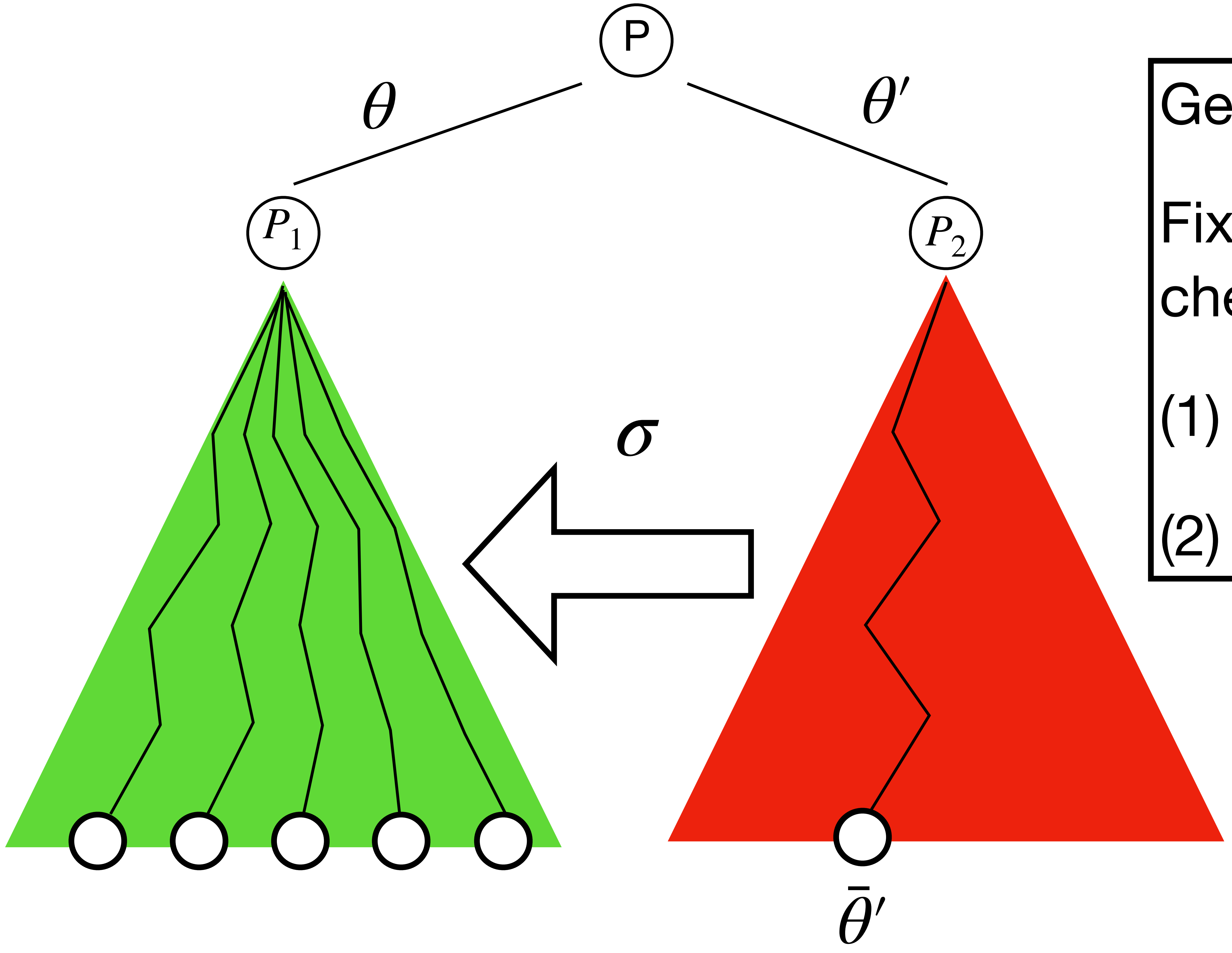
Theorem:

$$\theta \prec \theta' \text{ w.r.t. } P$$

$$\implies P \equiv (P \cup \neg\theta')$$

dominance breaking nogood

Dominance Breaking



Generation Problem:

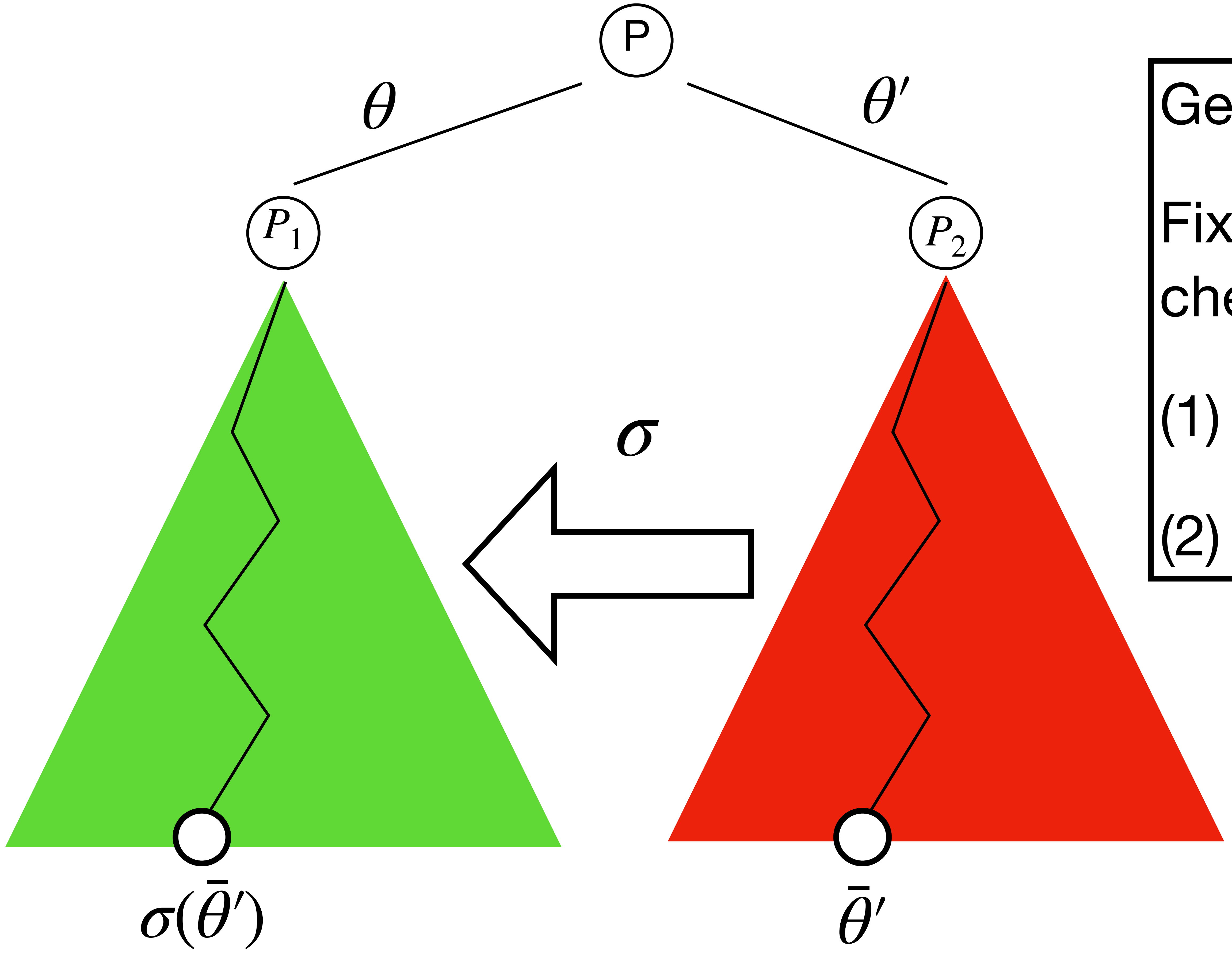
Fix length $|\theta| = |\theta'| = k$. For all θ', check whether there exists θ such that:

(1) $var(\theta) = var(\theta')$

(2) $\theta \prec \theta'$

how to check this condition?

Dominance Breaking



Generation Problem:

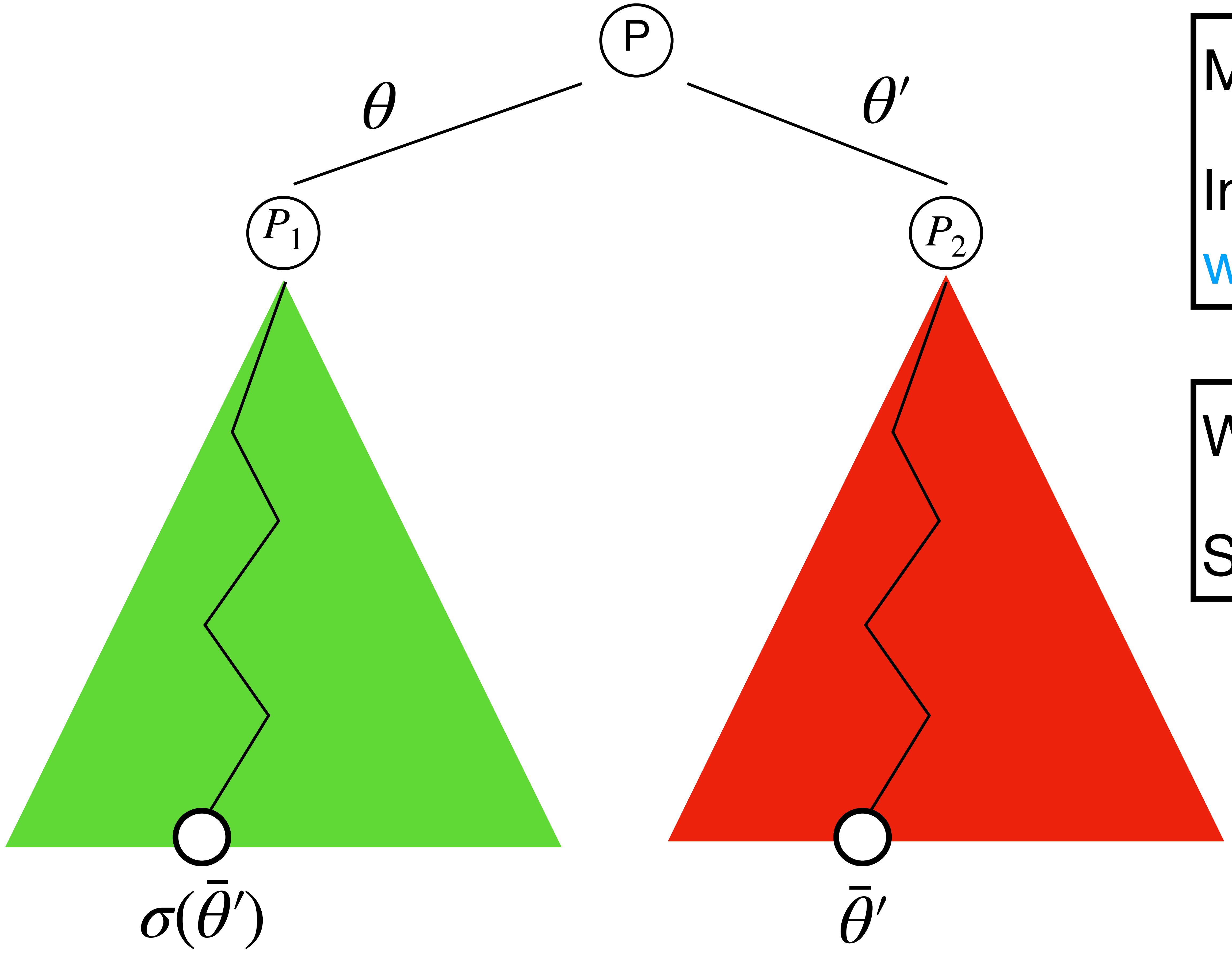
Fix length $|\theta| = |\theta'| = k$. For all θ', check whether there exists θ such that:

(1) $var(\theta) = var(\theta')$

(2) $\forall \bar{\theta}' \in \Theta^{P \cup \theta'}, \sigma(\bar{\theta}') \prec \bar{\theta}'$

how to check this condition?

Dominance Breaking



Mutation mapping: $\sigma(\bar{\theta}') = \bar{\theta}' \backslash \theta' \cup \theta$

In words, $\sigma(\bar{\theta}')$ extends θ in the same way $\bar{\theta}'$ extends θ'

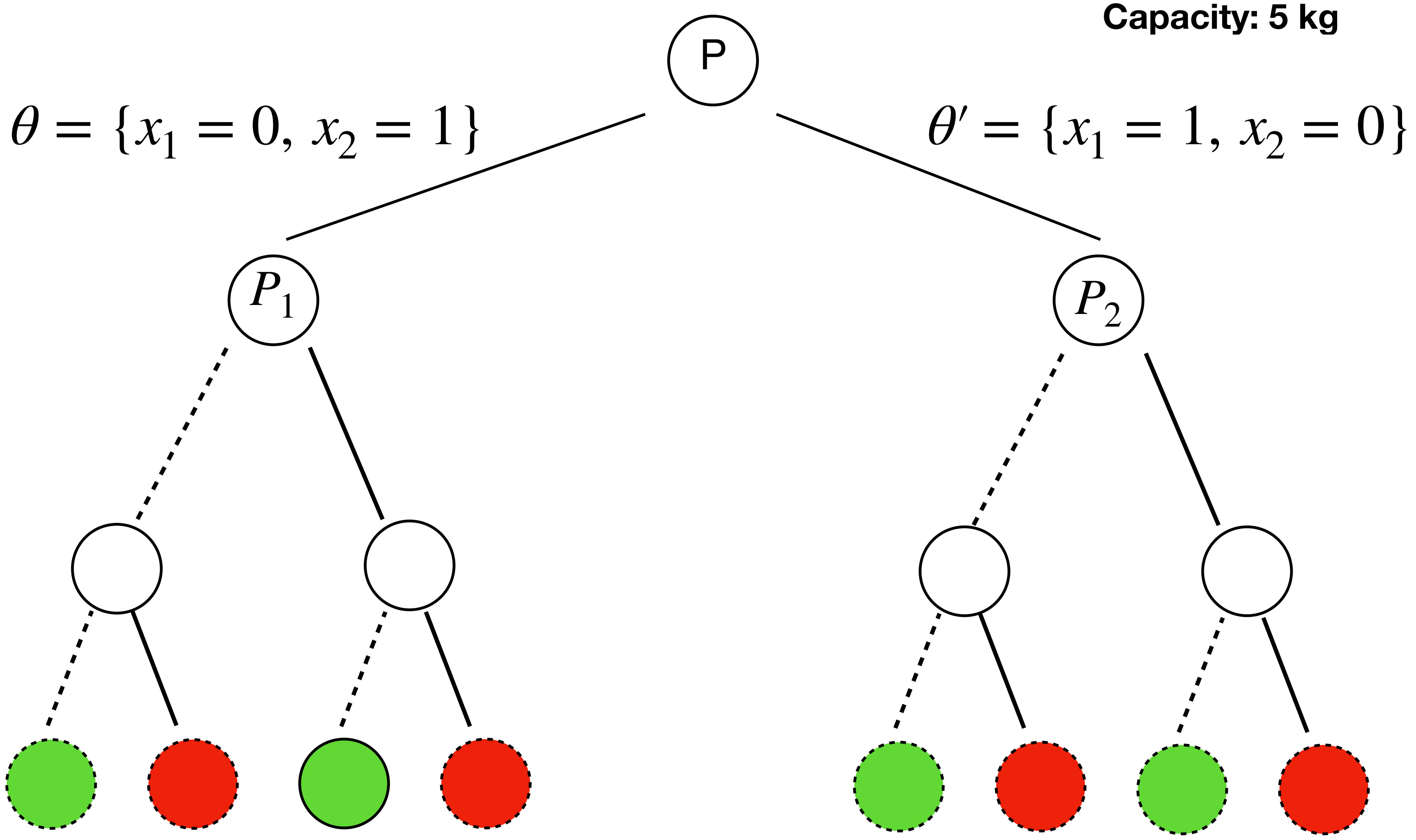
Want to show $\exists \bar{\theta}$ s.t. $\bar{\theta} \prec \bar{\theta}'$ for all $\bar{\theta}'$

Suffice to show $\sigma(\bar{\theta}') \prec \bar{\theta}'$ for all $\bar{\theta}'$

Dominance Relations

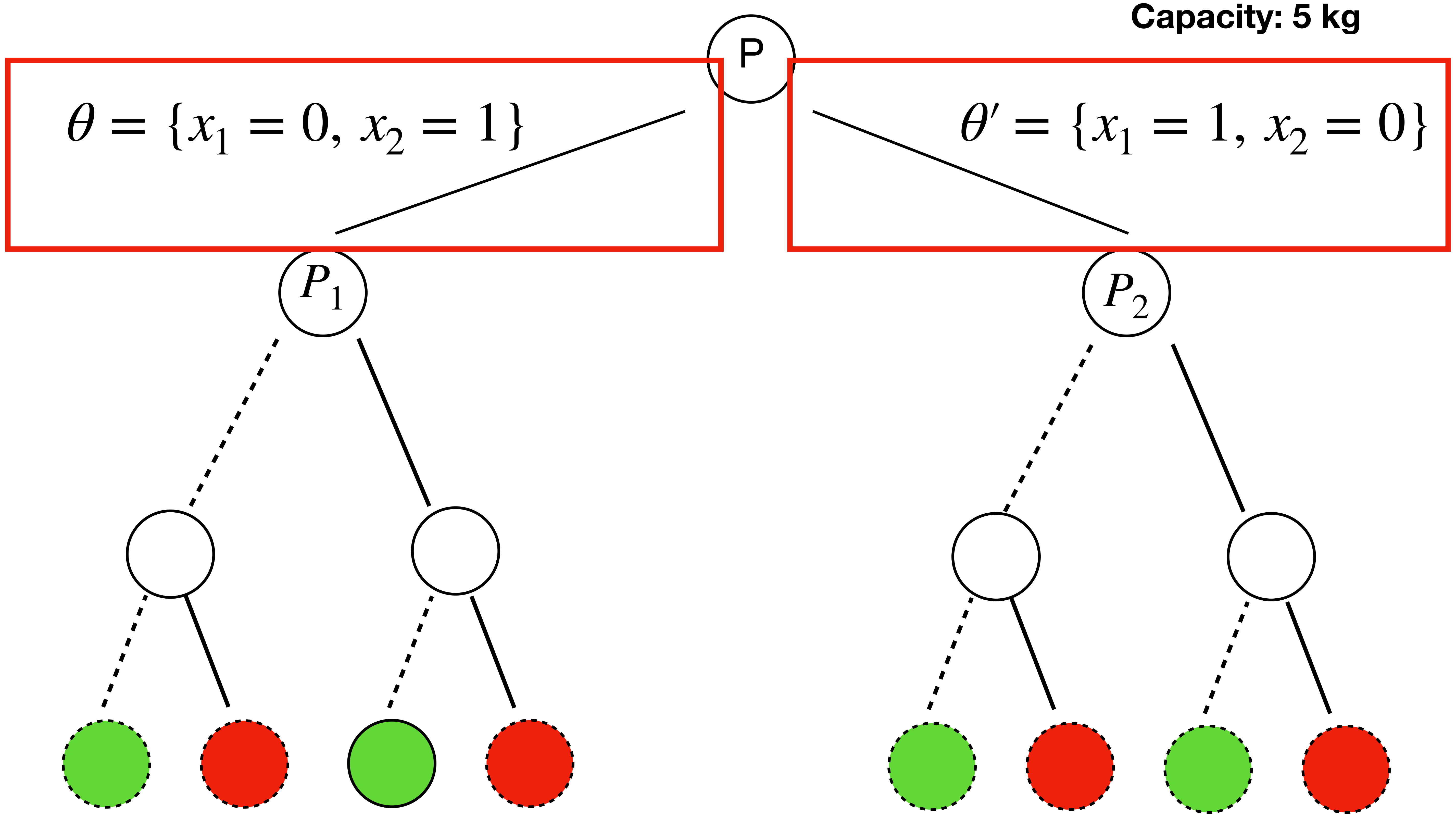
Capacity: 5 kg

Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4



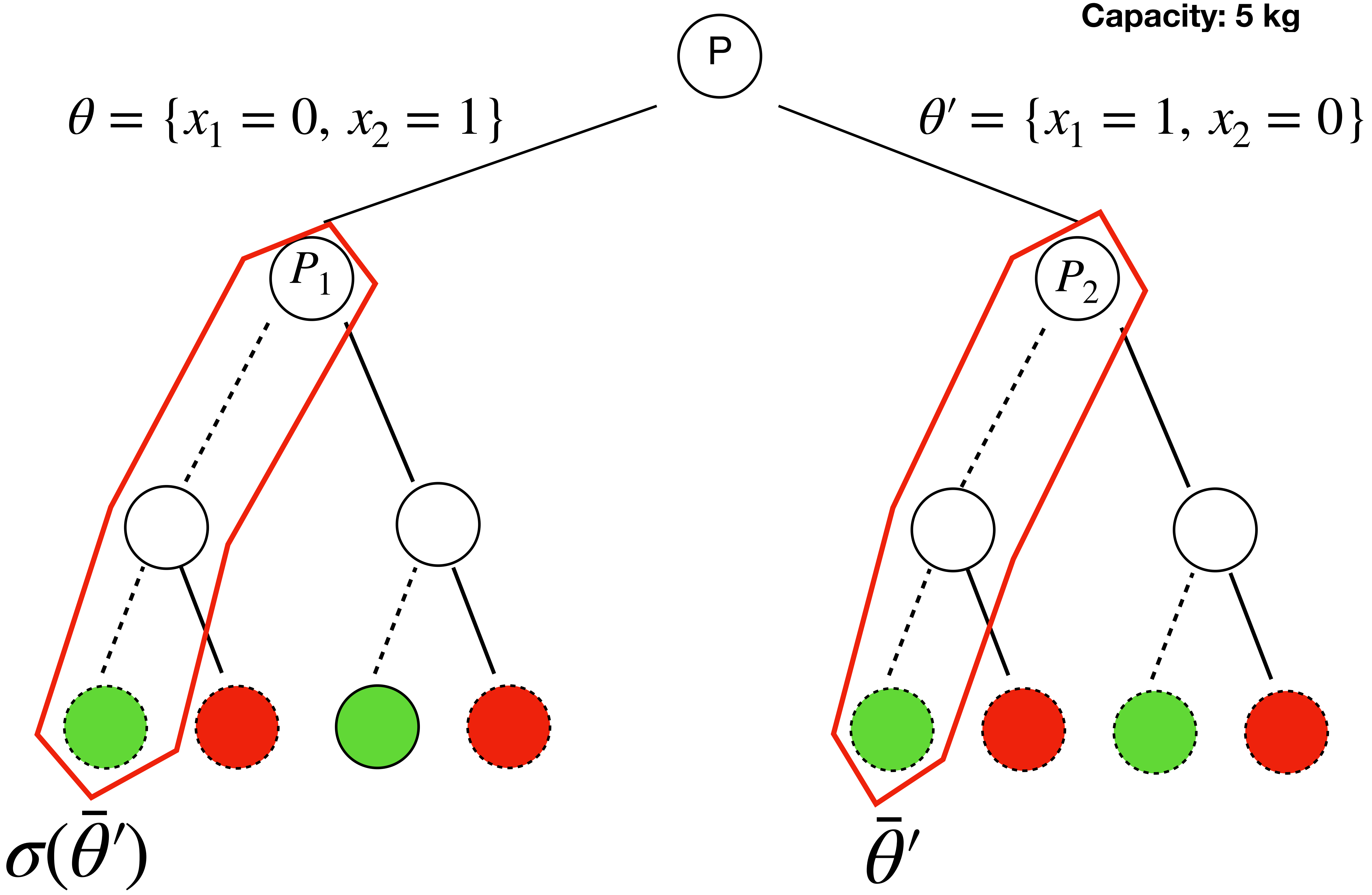
Dominance Relations

Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4



Dominance Relations

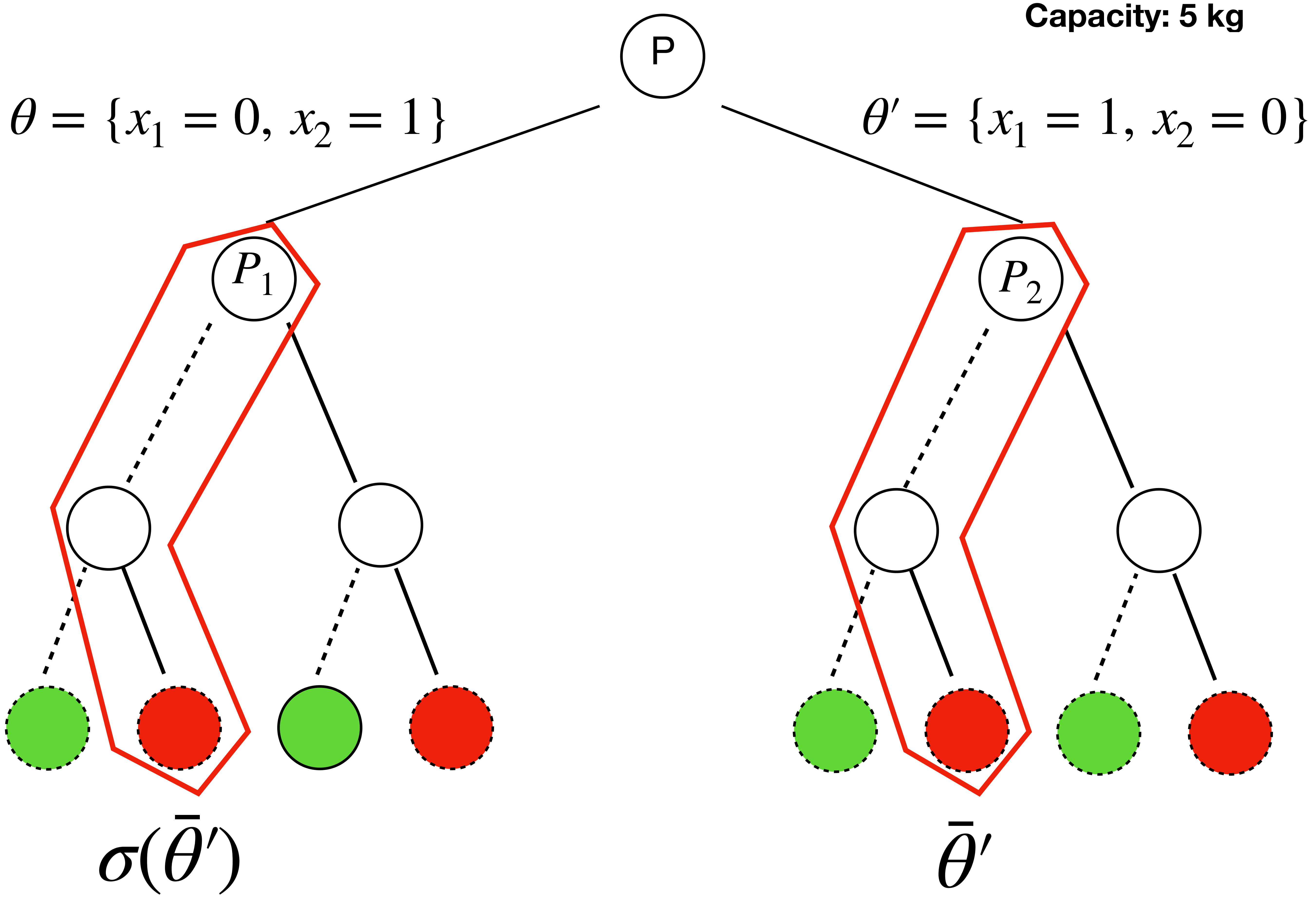
Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4



Dominance Relations


Capacity: 5 kg

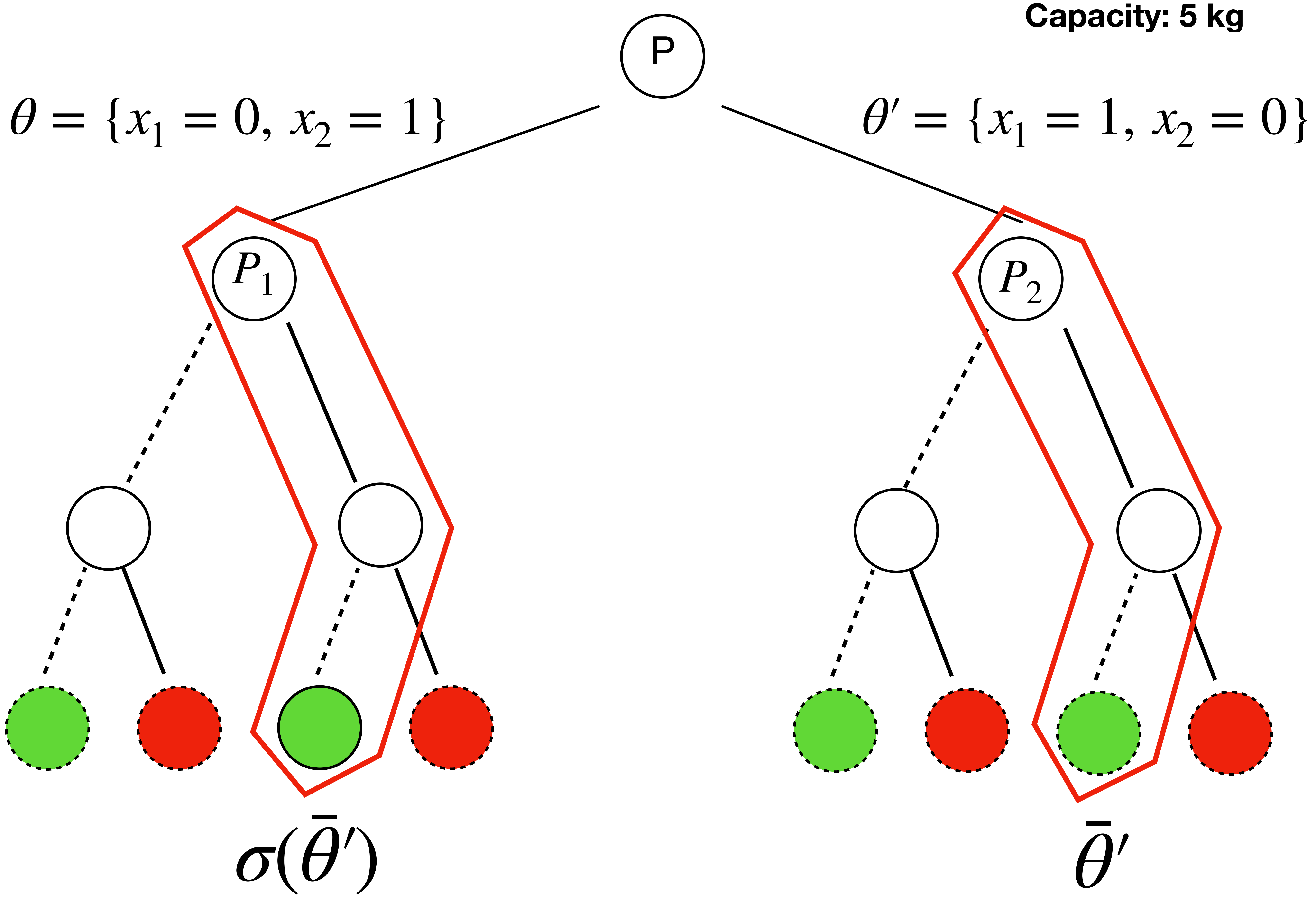
Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4



Dominance Relations

Capacity: 5 kg

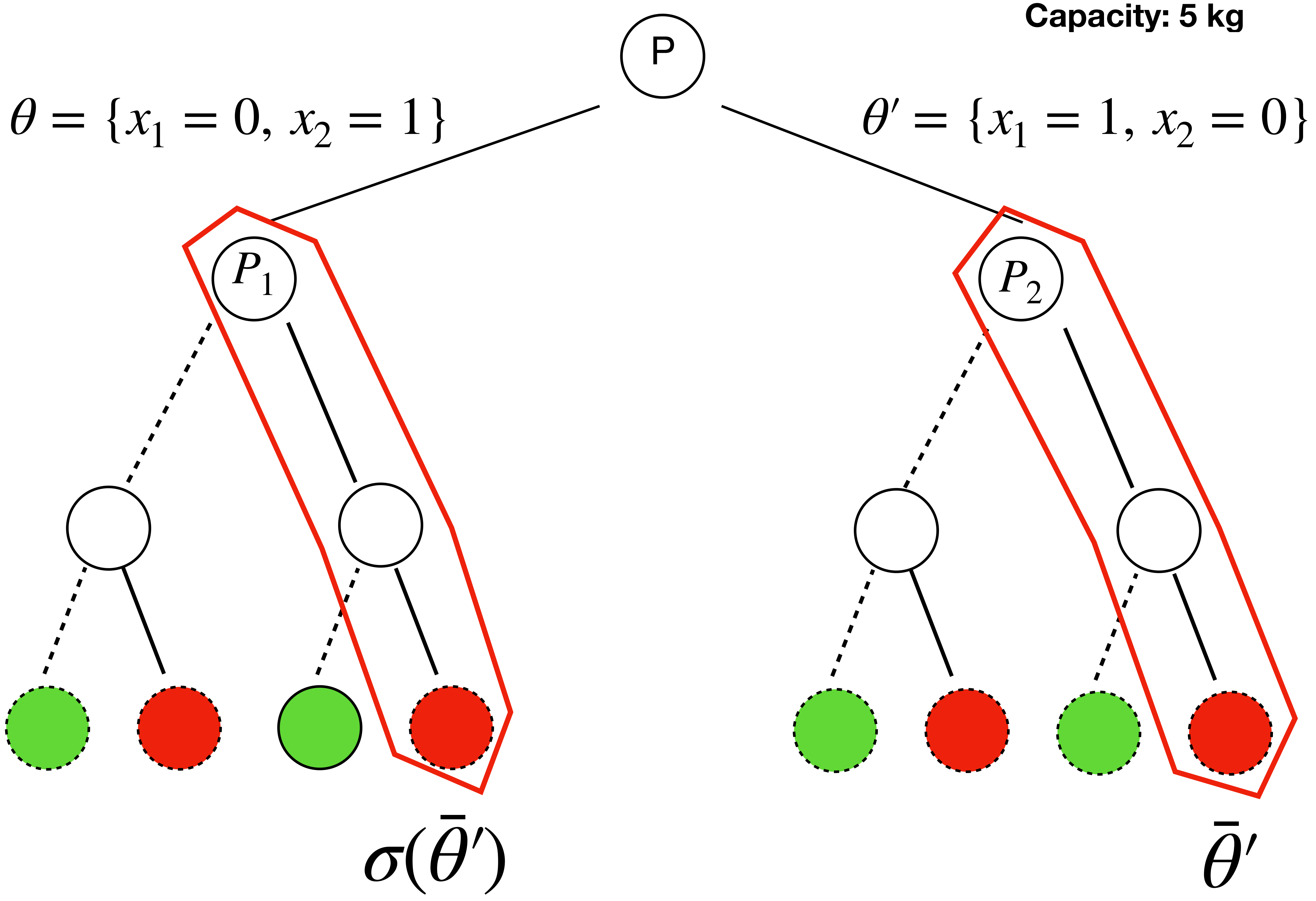
Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4



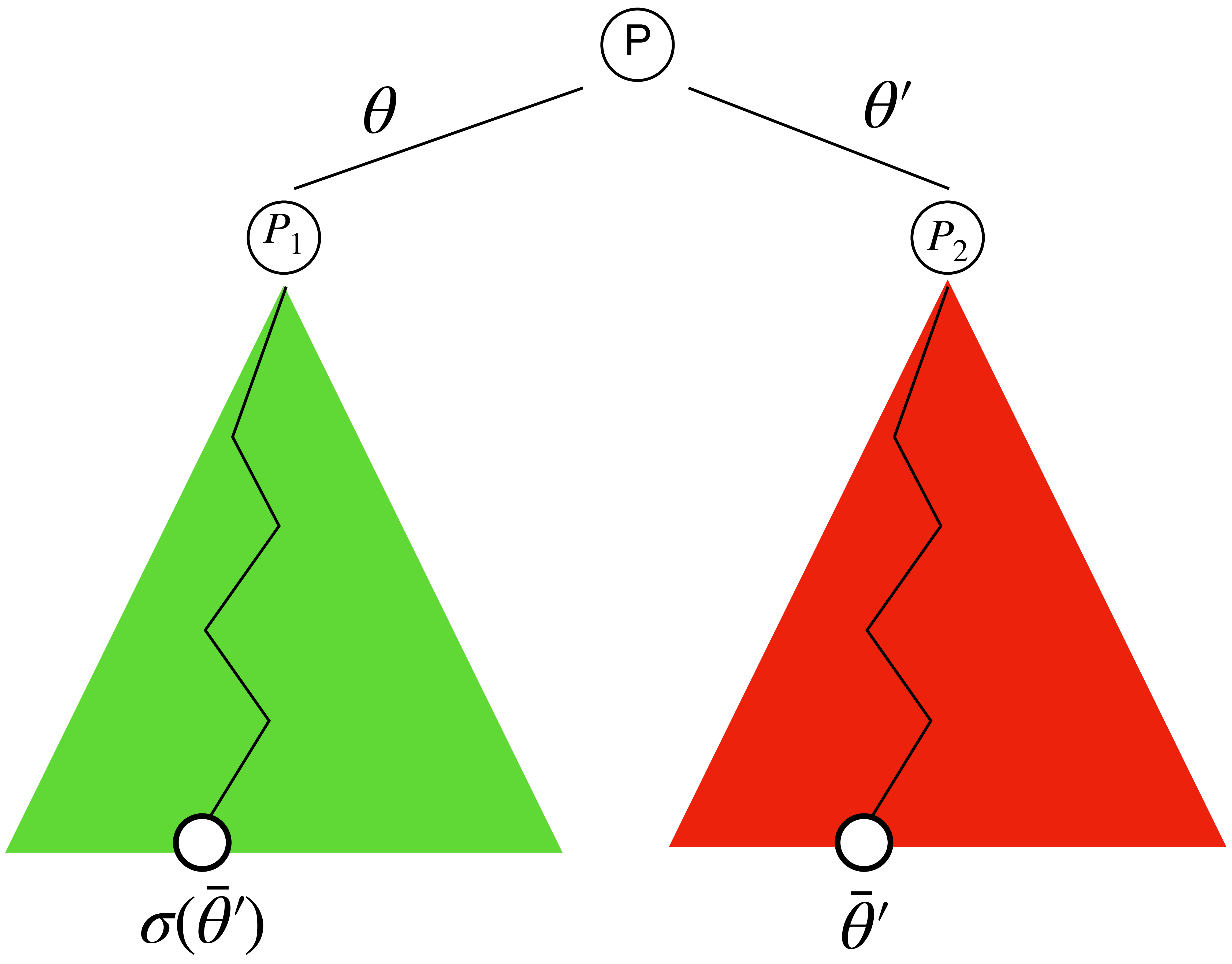
Dominance Relations

Capacity: 5 kg

Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4



Dominance Breaking



Need $\sigma(\bar{\theta}') \prec \bar{\theta}'$:

- $\sigma(\bar{\theta}')$ solution, $\bar{\theta}'$ non-solution
- both solutions, $f(\sigma(\bar{\theta}')) < f(\bar{\theta}')$
- both non-solutions, $f(\sigma(\bar{\theta}')) < f(\bar{\theta}')$

Theorem: $\sigma(\bar{\theta}') \prec \bar{\theta}'$ if

(1) implied satisfaction: if $\bar{\theta}'$ is a solution, $\sigma(\bar{\theta}')$ is also a solution

(2) betterment: $f(\sigma(\bar{\theta}')) < f(\bar{\theta}')$

Constraints on two partial assignments!

Modelling

```
int: n;       % number of items
int: W;       % knapsack capacity
array [1..n] of int: w;  % weight of each item
array [1..n] of int: v;  % value of each item

array [1..n] of var 0..1: x;

% constraint
constraint sum (i in 1..n) (w[i]*x[i]) <= W;

% objective
solve maximize sum (i in 1..n) (v[i]*x[i]);
```

Problem Model

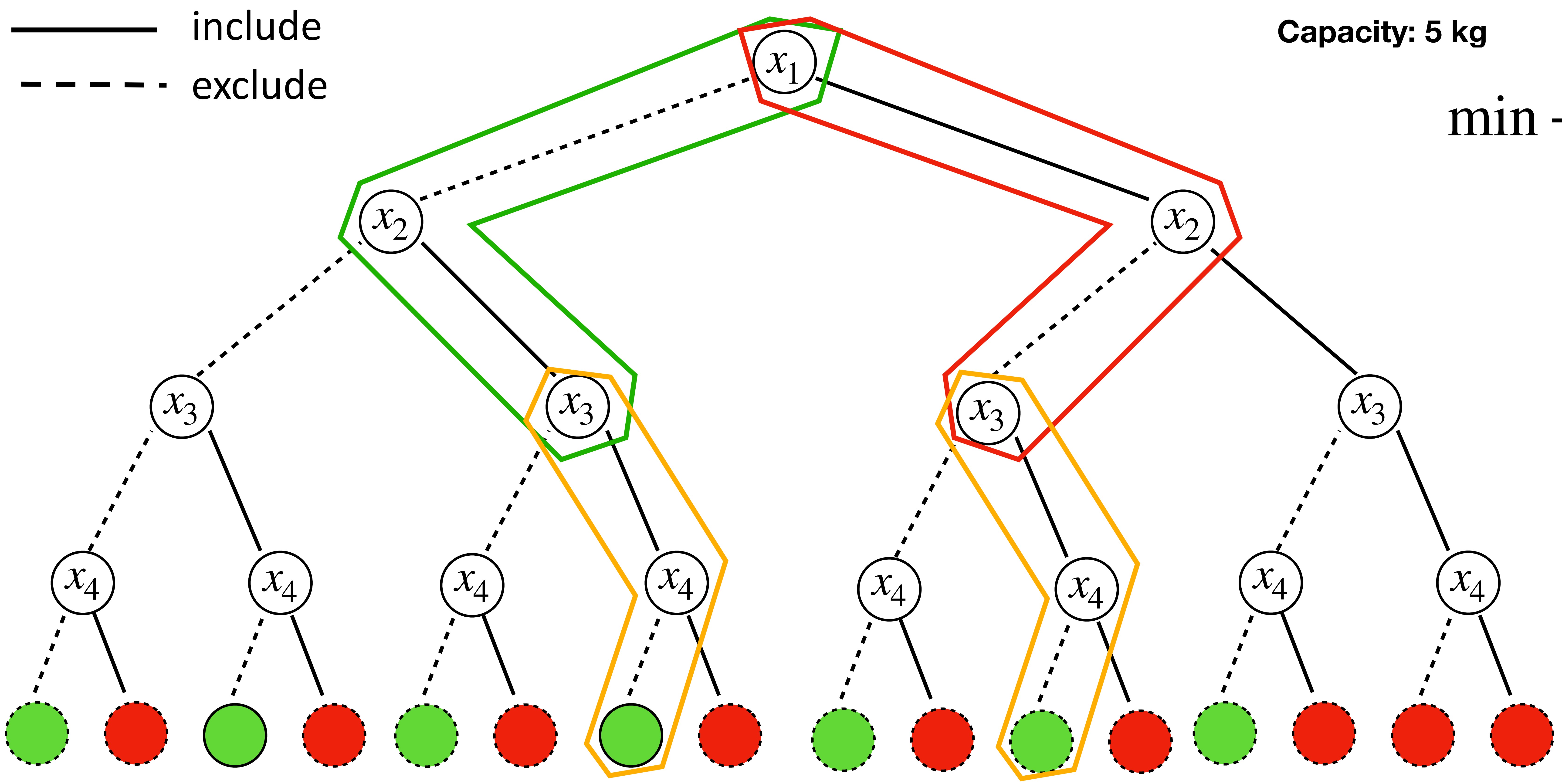
```
int: n;       % number of items
int: W;       % knapsack capacity
array [1..n] of int: w;  % weight of each item
array [1..n] of int: v;  % value of each item
int: k; % length of partial assignments

array [1..k] of var 1..n: F; % indices for fixed variable
array [1..k] of var 0..1: v1; % fixed value for \theta
array [1..k] of var 0..1: v2; % fixed value for \theta'
constraint increasing(F); % symmetry breaking

% constraint for implied satisfaction
…

% constraint for betterment
...
```

Generation Model

Betterment

- Separable function: sum of functions each depending on one variable
  - e.g. linear objective function
  - e.g. assignment problem
- Submodular set function: diminishing return property
  - e.g. cut function in weighted undirected graph

Betterment



Capacity: 5 kg

Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4

$$\min -(x_3 + 2x_2 + 10x_3 + 4x_4)$$

V_1 =-($2 + $10)

V_2 =-($1 + $10)

Because $10 = $10,

$2 > $1 $\Rightarrow V_1 < V_2$

Modelling

```
…

% objective
solve maximize sum (i in 1..n) (v[i]*x[i]);

…
```

Problem Model

```
…

array [1..k] of var 1..n: F; % indices for fixed variables
array [1..k] of var 0..1: v1; % fixed value for \theta
array [1..k] of var 0..1: v2; % fixed value for \theta'
constraint increasing(F);
…

% constraint for betterment
constraint sum(t in 1..k)( v[F[t]] * v1[t] )
            > sum(t in 1..k)( v[F[t]] * v2[t] );
…
```

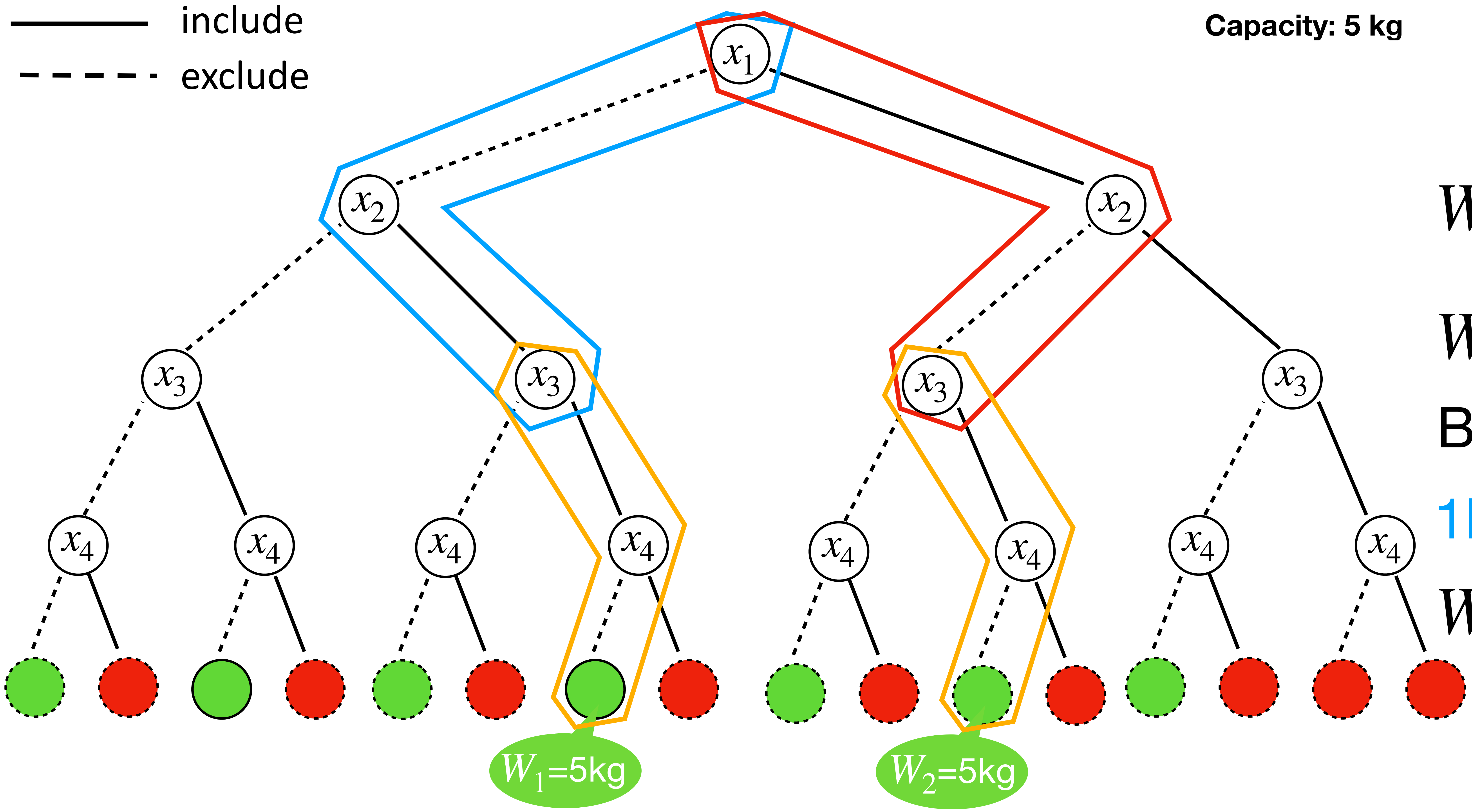
Generation Model

Implied Satisfaction

- Domain constraint
- Boolean disjunction constraint
- Linear Inequality constraint
- Alldifferent constraint

Implied Satisfaction

Item	Weight	Value
1	1kg	$1
2	1kg	$2
3	4kg	$10
4	12kg	$4



W_1= 1kg+ 4kg ≤ 5kg

W_2= 1kg+ 4kg ≤ 5kg

Because 4kg = 4kg,

1kg ≤ 1kg implies

$$W_1 \leq 5kg \Rightarrow W_2 \leq 5kg$$

Modelling

```
…

% linear inequality constraint
constraint sum (i in 1..n) (w[i]*x[i]) <= W;

…
```

Problem Model

```
…

array [1..k] of var 1..n: F; % indices for fixed variable
array [1..k] of var 0..1: v1; % fixed value for \theta
array [1..k] of var 0..1: v2; % fixed value for \theta'
constraint increasing(F);

% constraint for implied satisfaction
constraint sum(t in 1..k)( w[F[t]] * v1[t] )
           <=  sum(t in 1..k)( w[F[t]] * v2[t] );

…
```

Generation Model

Agenda

- Background
- Automatic Dominance Breaking
- Experimental Results
- Q & A

Experimental Results

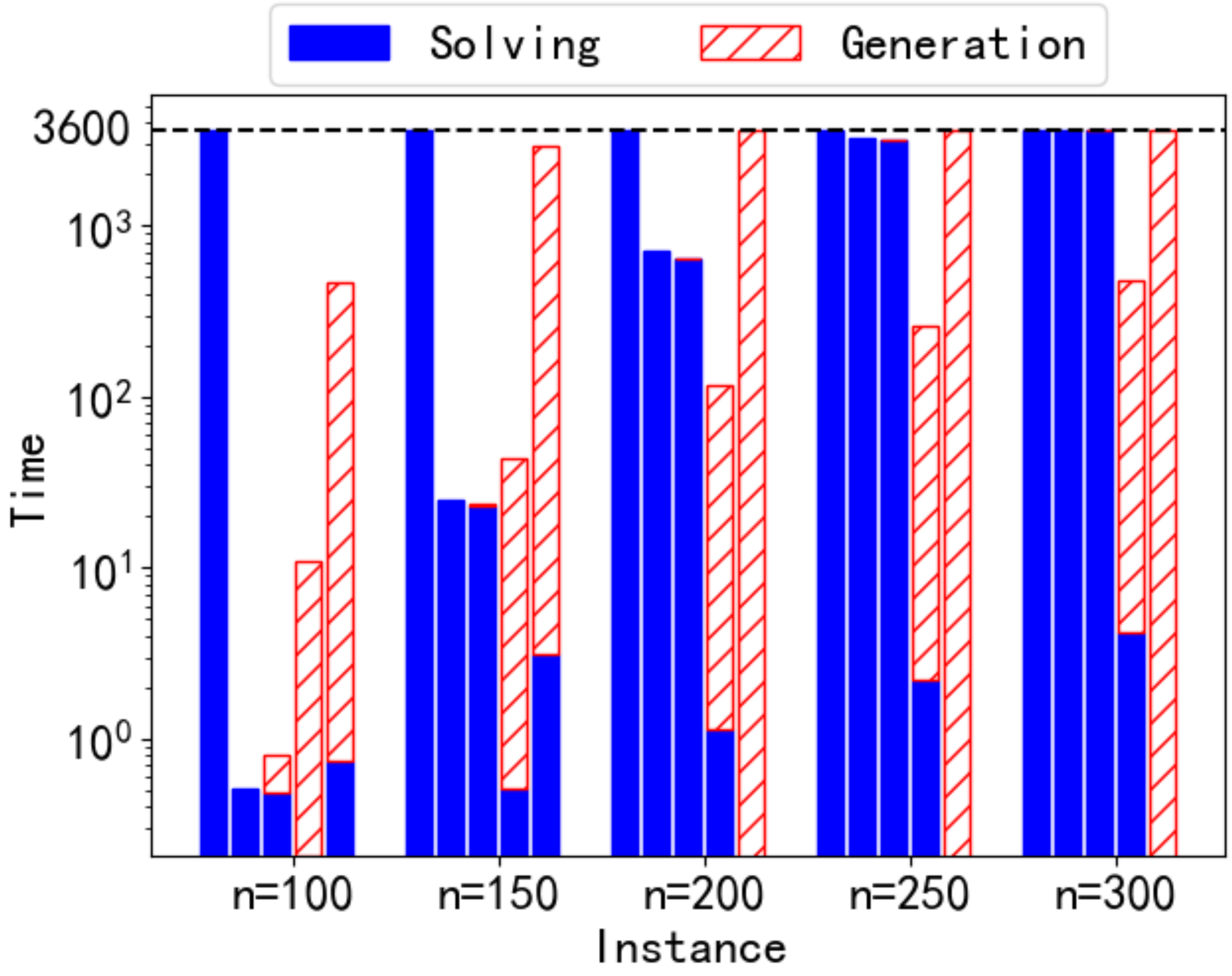
- Four benchmarks, 10 random instances for each configuration
    - **KP**: 0-1 Knapsack Problem
    - **DCKP**: Disjunctively Constrained Knapsack Problem
    - **CHSP**: Capacitated Concert Hall Scheduling Problem
    - **WMCP**: Weighted Maximum Cut Problem
- Use MiniZinc to model problems and nogood generation, and the backend solver is Chuffed
- One hour timeout for nogood generation + problem solving

Experimental Results

- Compare five different methods:
  - **no-dom**: basic BnB
  - **manual**: BnB with manual DB constraints
  - **2-dom**: BnB with generated DB nogoods of $k \leq 2$
  - **3-dom**: BnB with generated DB nogoods of $k \leq 3$
  - **4-dom**: BnB with generated DB nogoods of $k \leq 4$

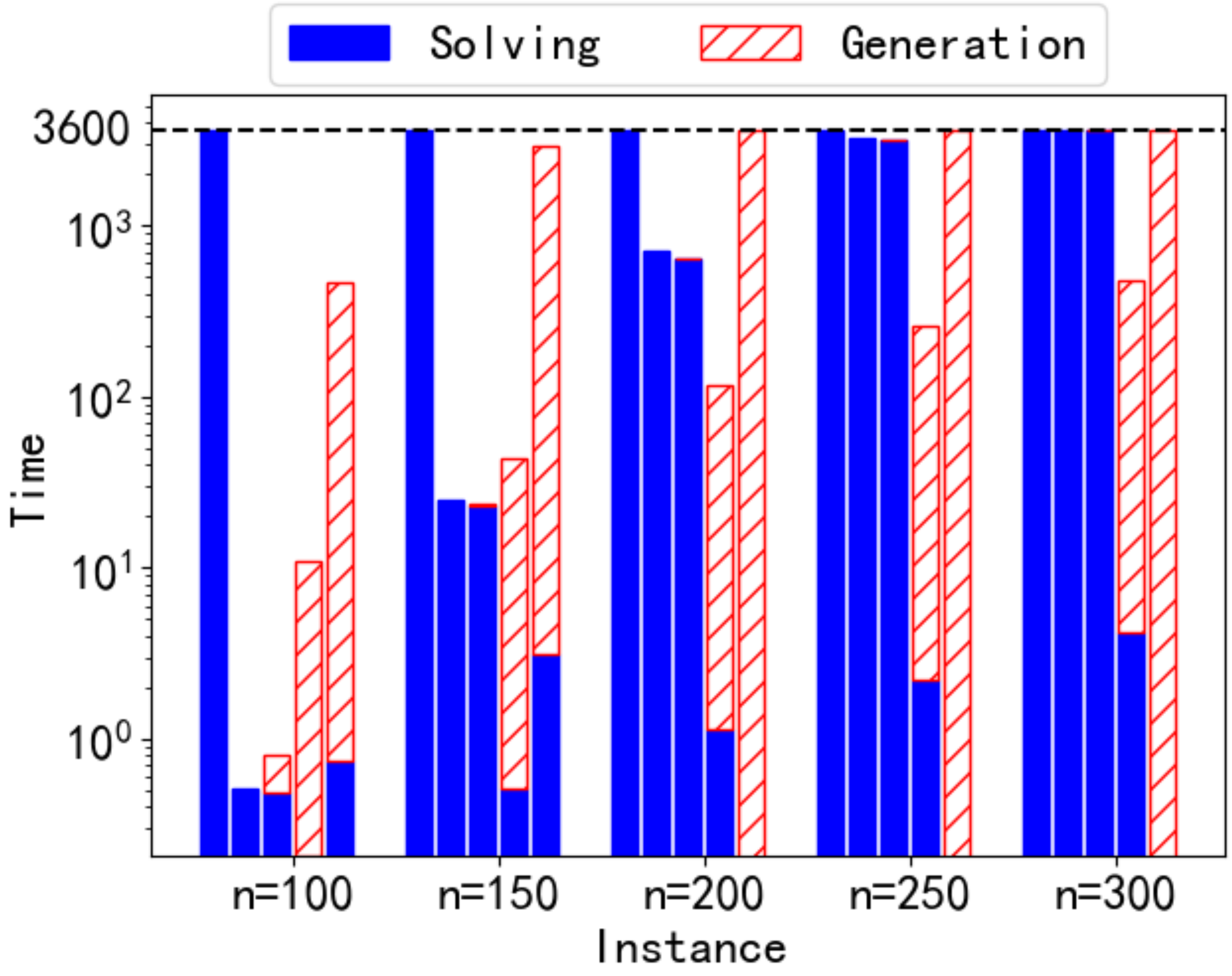
Experimental Results

KP Time

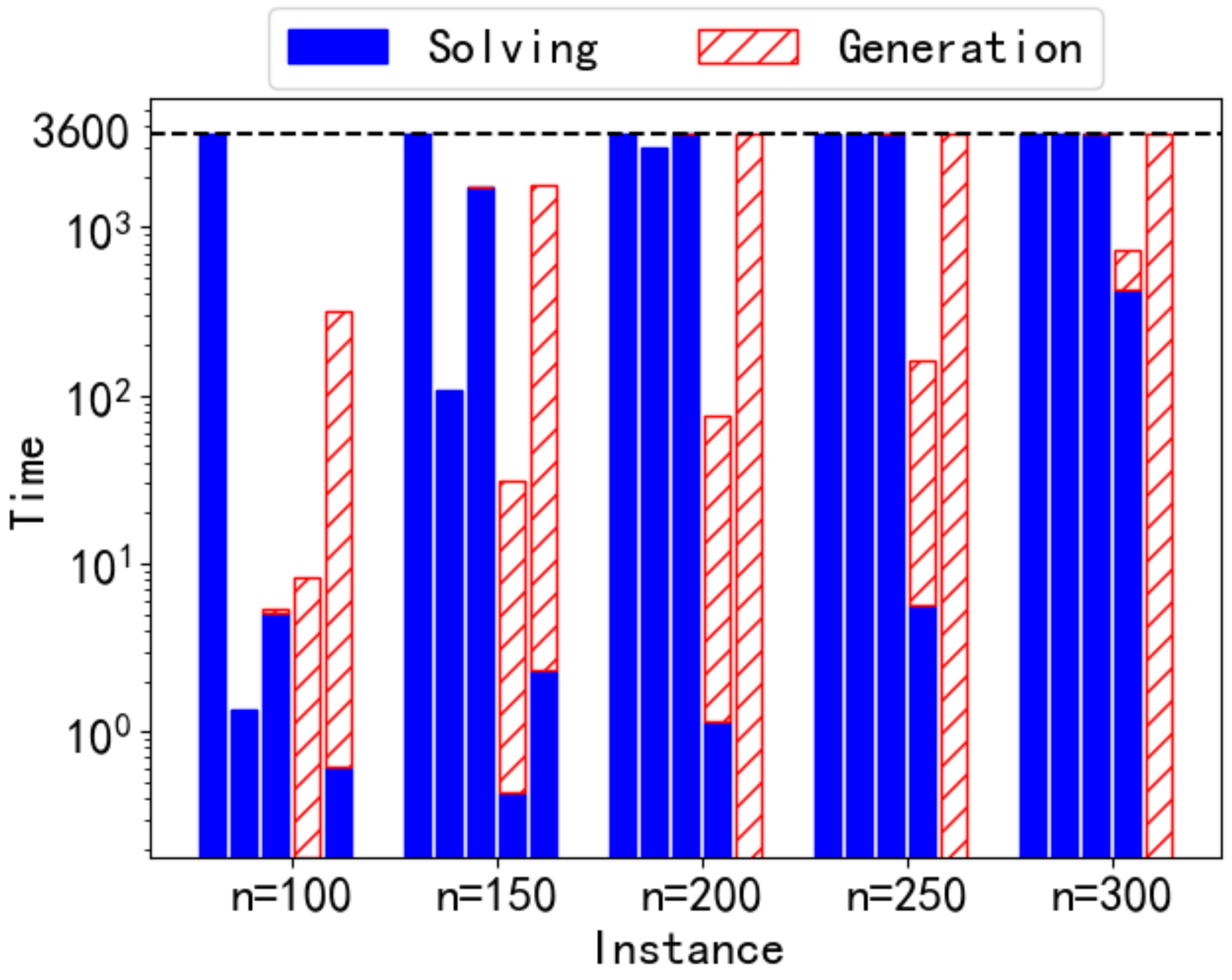


Experimental Results

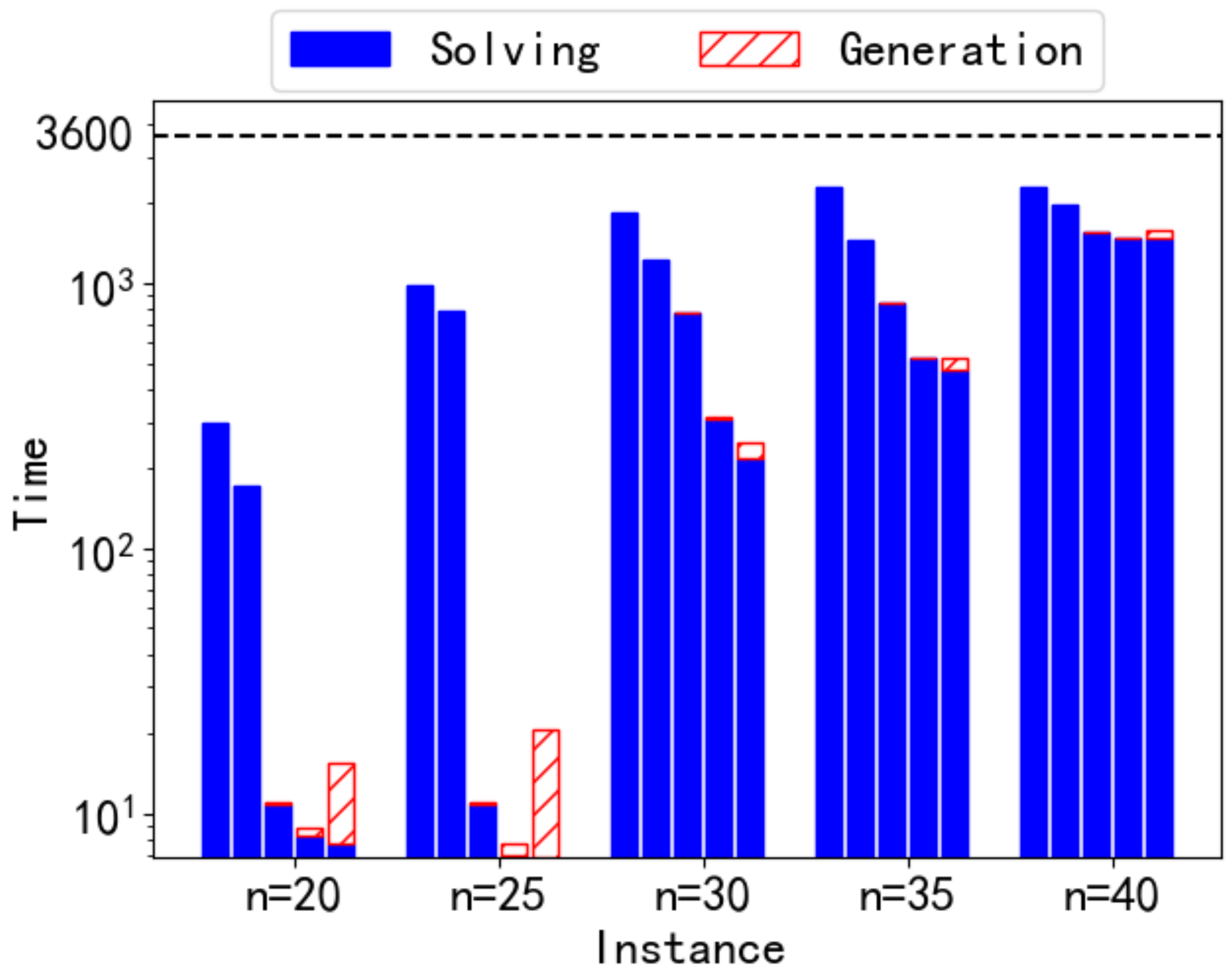
KP Time



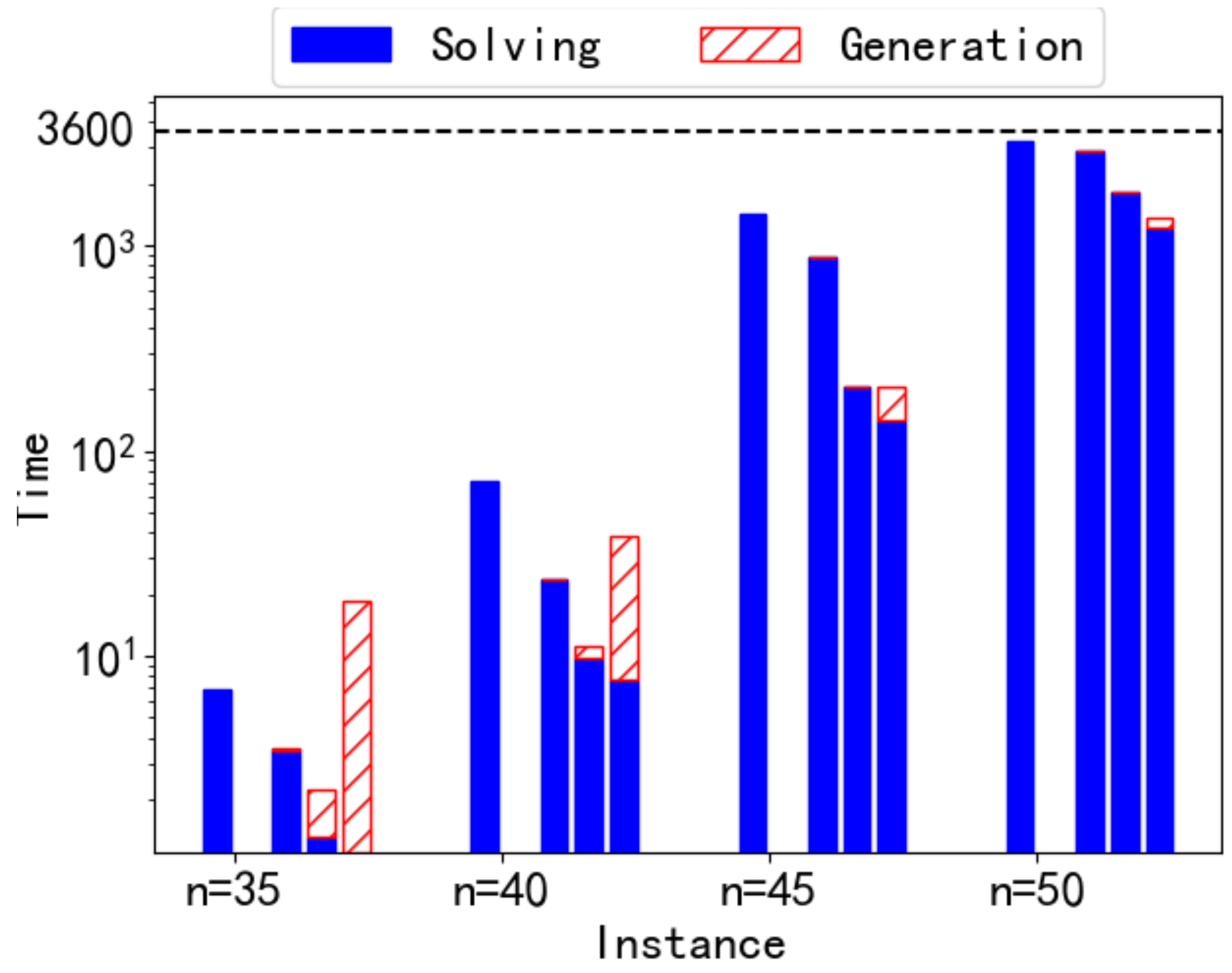
Experimental Results



DCKP Time



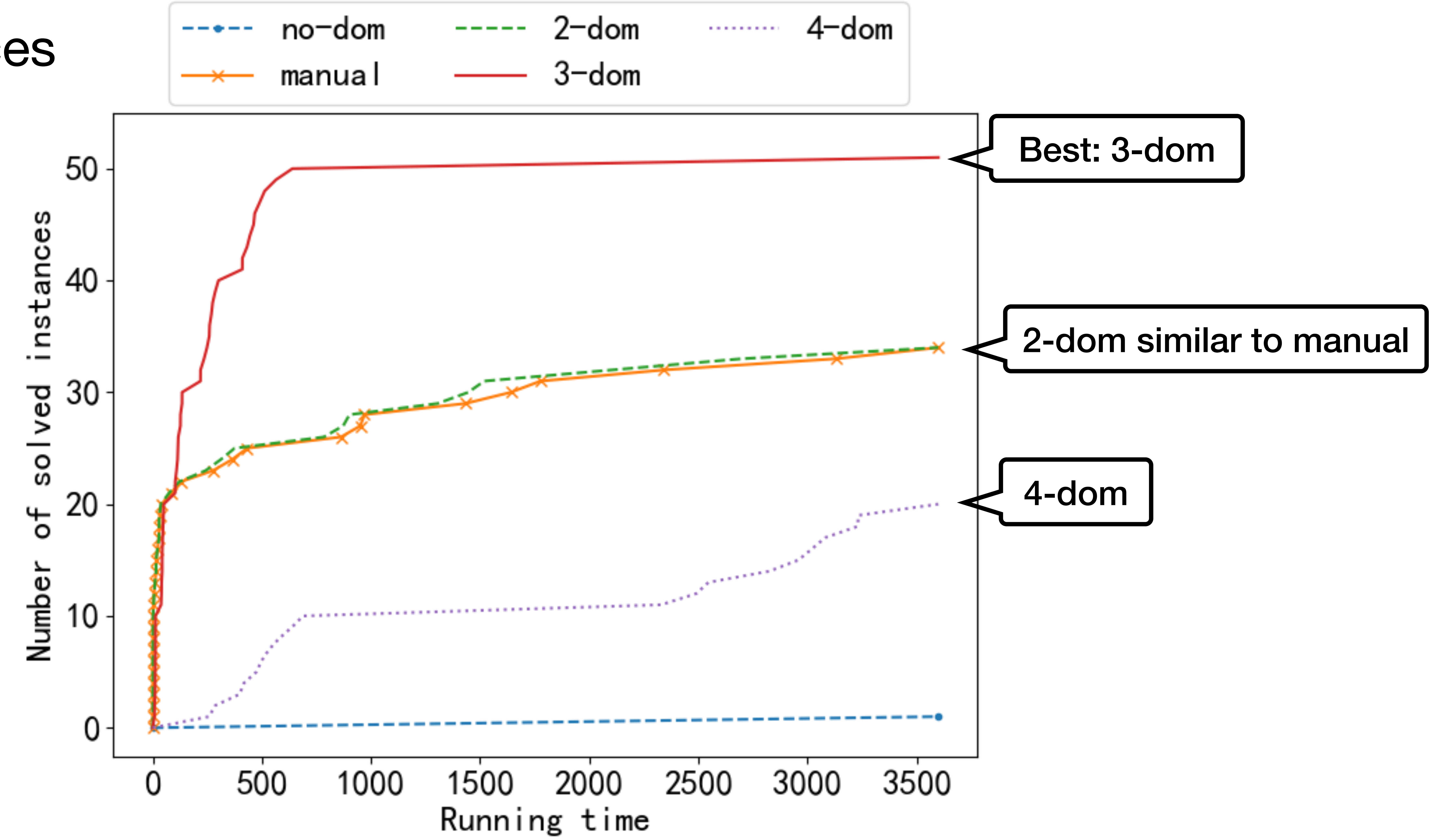
CHSP Time



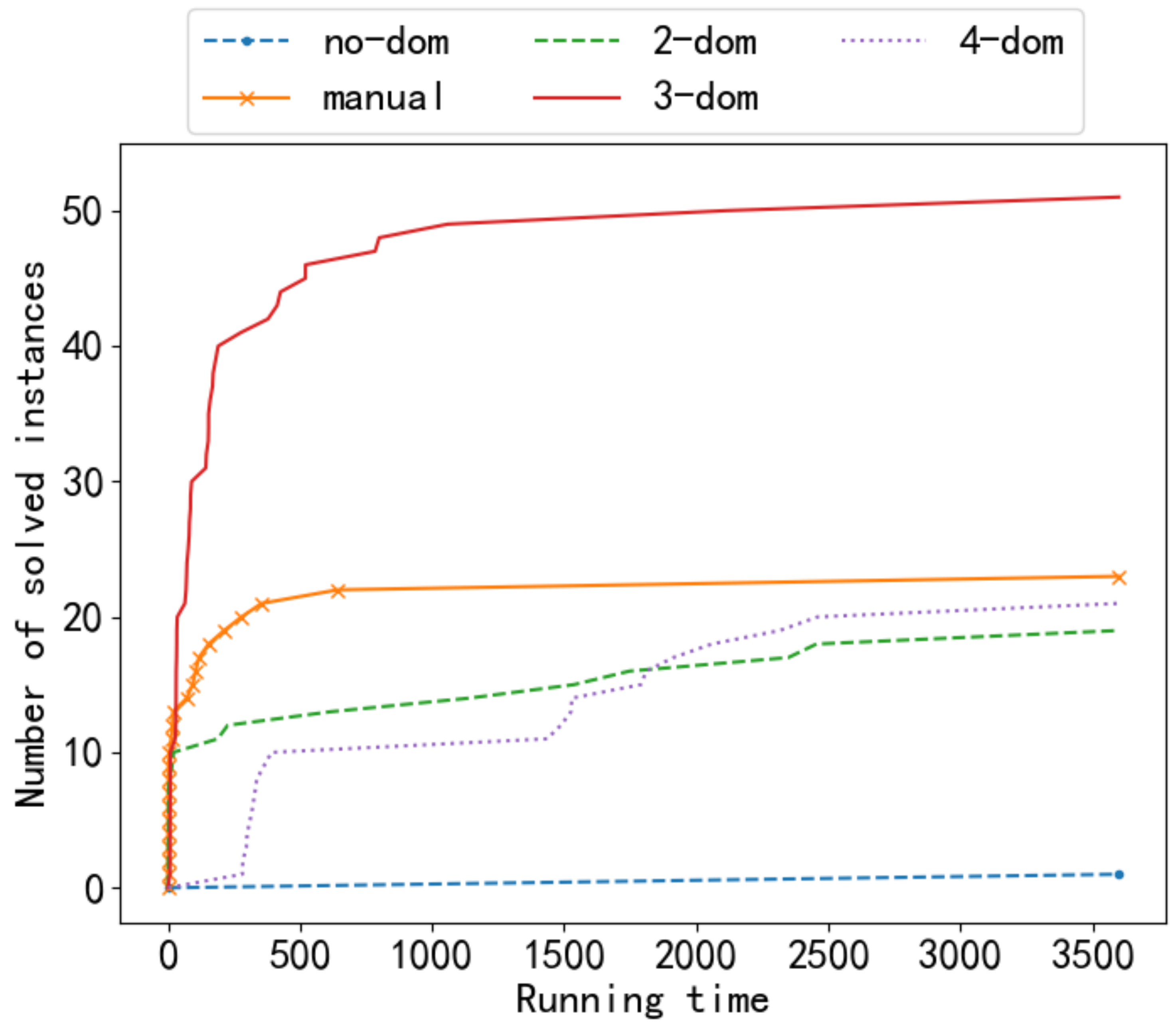
WMCP Time

Experimental Results

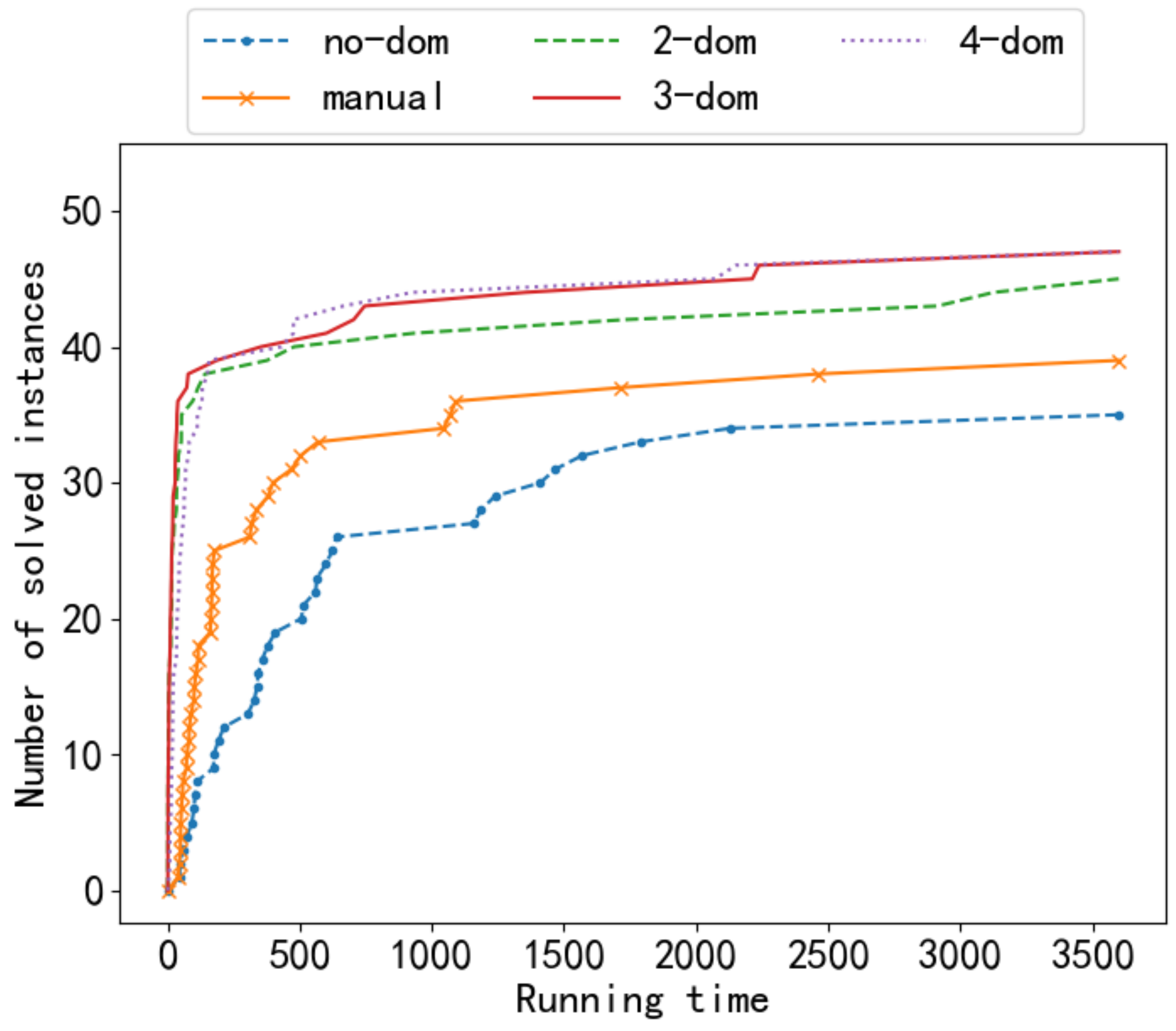
KP Solved Instances



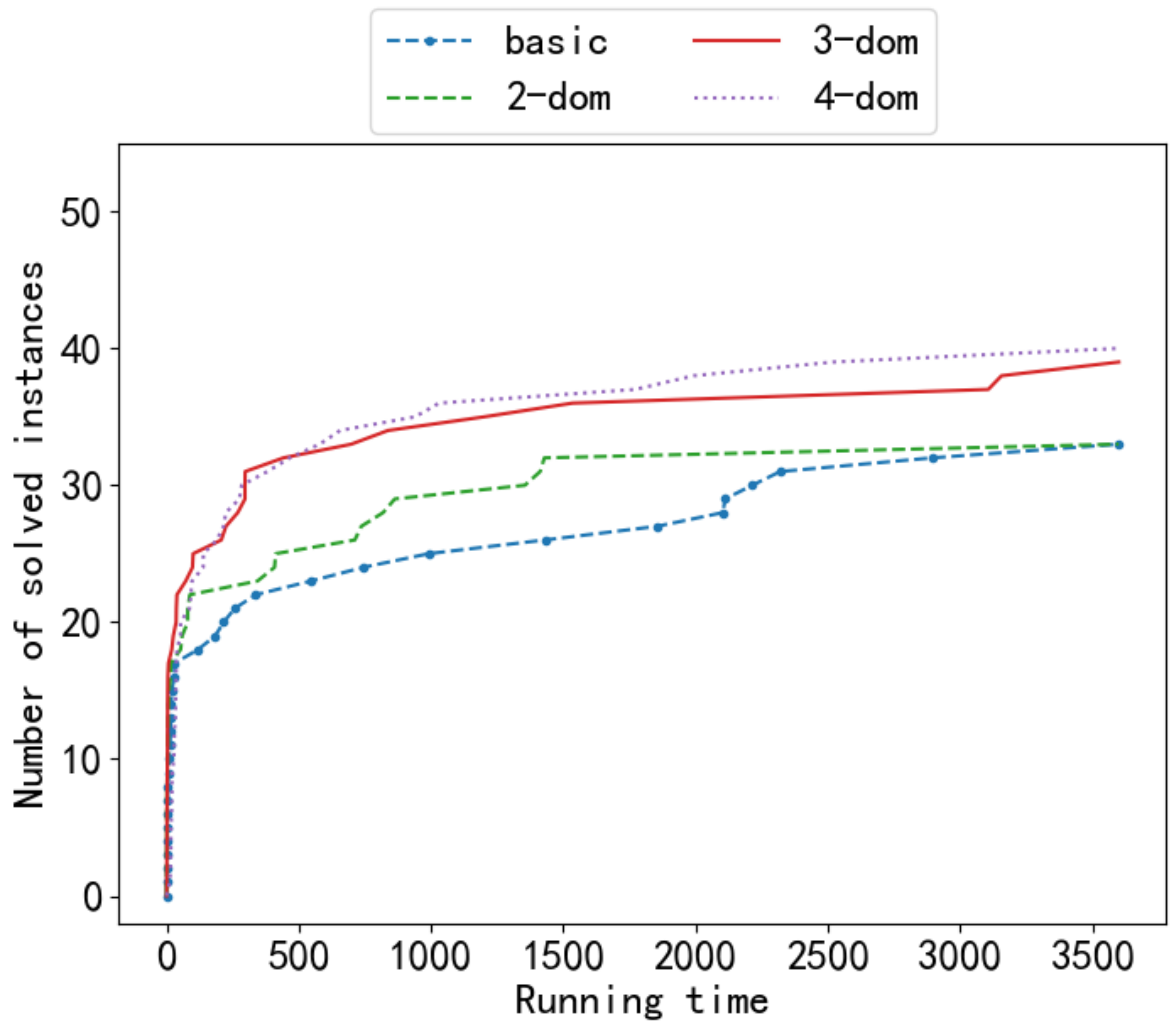
Experimental Results



DCKP



CHSP



WMCP

Conclusion

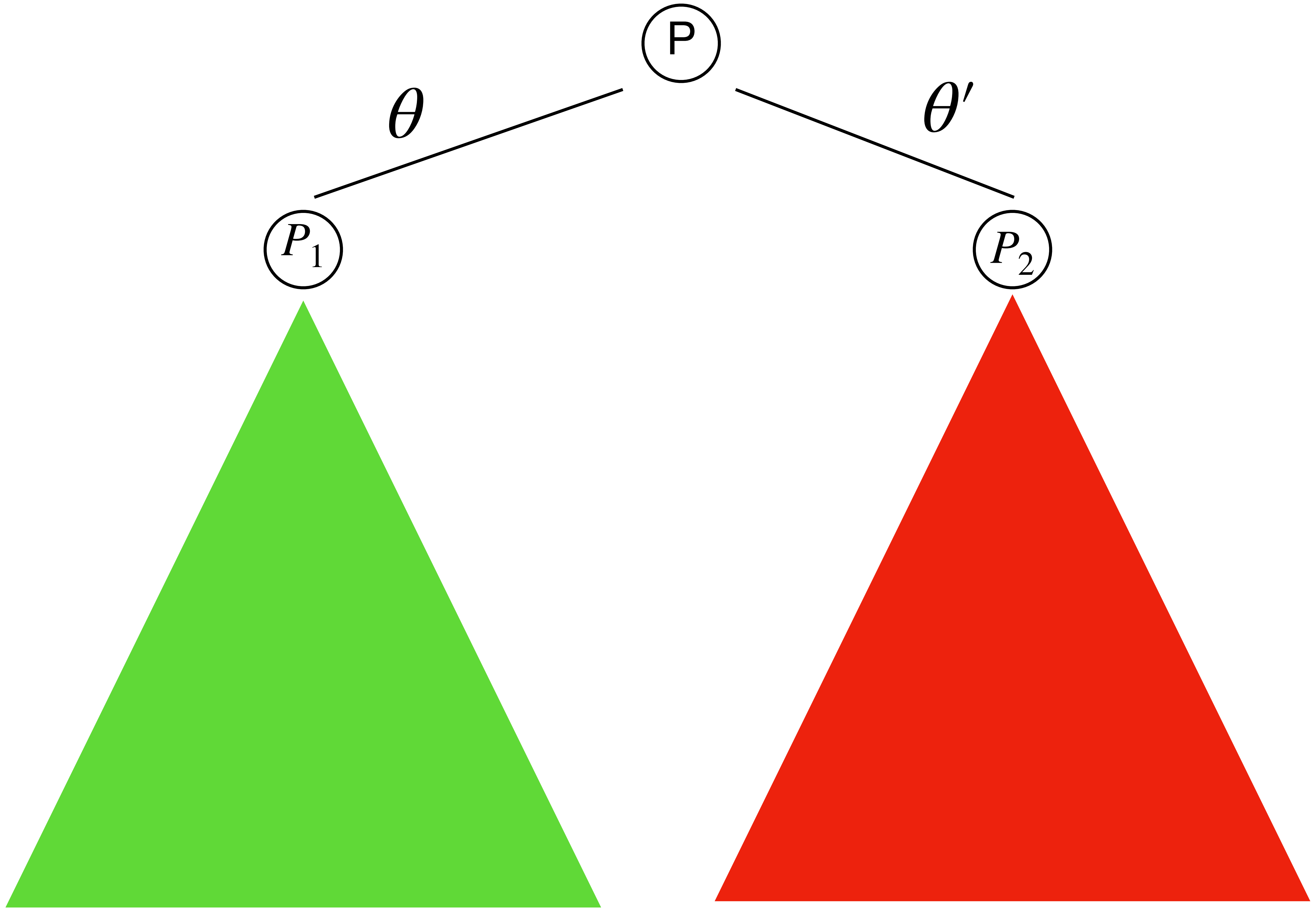
- We propose a fully automatic method for dominance breaking for a class of constraint optimization problems
- Our method can generate dominance breaking nogoods that are stronger than that of manual dominance breaking
- Future works:
    - Improve the efficiency of nogood generation and reduce overhead
    - Analyse more objective and constraint types for betterment and implied satisfaction

Q & A

Dominance Breaking

Theorem:

There are totally $O(\binom{n}{k}\binom{d^k}{2})$ pairs of (θ, θ')



Generation Problem:

Fix length $|\theta| = |\theta'| = k$. For all θ', check whether there exists θ such that:

(1) $var(\theta) = var(\theta')$

(2) $\theta \prec \theta'$

Modelling

```
int: n;       % number of items
int: W;       % knapsack capacity
array [1..n] of int: w;   % weight of each item
array [1..n] of int: v;   % value of each item
int: k; % length of partial assignments

array [1..k] of var 1..n: F; % indices for fixed variable
array [1..k] of var 0..1: v1; % fixed value for \theta
array [1..k] of var 0..1: v2; % fixed value for \theta'
constraint increasing(F);

% constraint for implied satisfaction
…
% constraint for betterment
...
```

Generation Model

Example:

F = [3, 10], v1 = [0, 1], v2 = [1, 0] represents:

$\theta = \{x_3 = 0, x_{10} = 1\}$ and

$\theta' = \{x_3 = 1, x_{10} = 0\}$

Symmetry!

Example:

F = [10, 3], v1 = [1, 0], v2 = [0, 1] represents:

$\theta = \{x_3 = 0, x_{10} = 1\}$ and

$\theta' = \{x_3 = 1, x_{10} = 0\}$

Modelling

```
int: n;        % number of items
int: W;        % knapsack capacity
array [1..n] of int: w;   % weight of each item
array [1..n] of int: v;   % value of each item
int: k; % length of partial assignments

array [1..k] of var 1..n: F; % indices for fixed variable
array [1..k] of var 0..1: v1; % fixed value for \theta
array [1..k] of var 0..1: v2; % fixed value for \theta'
constraint increasing(F);

% constraint for implied satisfaction
…
% constraint for betterment
...
```

Generation Model

Example:

F = [3, 10], v1 = [0, 1], v2 = [1, 0] represents:

$\theta = \{x_3 = 0, x_{10} = 1\}$ and

$\theta' = \{x_3 = 1, x_{10} = 0\}$

Symmetry!

Submodular Set Function

```
int: n; % number of nodes
int: m; % number of edges
array [1..m] of int: w; % weights of edges
array [1..m, 1..2] of 1..n: edges;

array [1..n] of var 0..1: x;

var bool: crossing =
     [x[edges[i, 1]] != x[edges[i, 2]] | i in 1..m];

% objective
solve maximize
  sum(i in 1..m) (bool2int(crossing[i]) * w[i] );
```

Problem Model

```
…
array [1..k] of var 1..n: F; % indices for fixed variables
array [1..k] of var 0..1: v1; % fixed value for \theta
array [1..k] of var 0..1: v2; % fixed value for \theta'
constraint increasing(F);
…
% submodular set function
function var int: obj(array[1..k] of var 1..n: F,
array[1..k] of var 0..1: v) = … ;

% constraint for betterment
constraint forall(i in 1..k)( v1[i] = 1 -> v2[i] = 1 );
constraint obj(F, v1) > obj(F, v2);
```

Generation Model

Domain constraint

```
…
array [1..10] of var 0..10: x;

% domain constraint
constraint x[5] in {0, 1, 2};
…
```

Problem Model

```
…
array [1..k] of var 1..n: F; % indices for fixed variables
array [1..k] of var 0..1: v1; % fixed value for \theta
array [1..k] of var 0..1: v2; % fixed value for \theta'
constraint increasing(F);
…
% implied satisfaction
constraint forall(i in 1..k)( (F[i] = 5)  ->
            (v1[F[i]] in {0, 1, 2} /\ v2[F[i]] in {0, 1, 2})
);
…
```

Generation Model

Alldifferent constraint

```
…
array [1..10] of var 0..10: x;
set of int: U;

% alldifferent constraint
constraint alldifferent( {x[i] | i in U} );
…
```

Problem Model

```
…
array [1..k] of var 1..n: F; % indices for fixed variables
array [1..k] of var 0..1: v1; % fixed value for \theta
array [1..k] of var 0..1: v2; % fixed value for \theta'
constraint increasing(F);
…
% implied satisfaction
forall(i, j in 1..k where i < j)(
        (F[i] in U ∧ F[j] in U) ->
        (v1[F[i]] != v1[F[j]] ∧ v2[F[i]] != v2[F[j]])
);
…
```

Generation Model

Boolean Disjunction constraint

```
…

array [1..10] of var bool: x;

% domain constraint
constraint x[1] ∨ x[2] ∨ x[5];

…
```

Problem Model

```
…
array [1..k] of var 1..n: F; % indices for fixed variables
array [1..k] of var bool: v1; % fixed value for \theta
array [1..k] of var bool: v2; % fixed value for \theta'
constraint increasing(F);
…
% implied satisfaction
constraint exists(i in 1..k)( F[i] in {1, 2, 5} ∧ v1[i])
        <=  exists(i in 1..k)( F[i] in {1, 2, 5} ∧ v2[i]);

…
```

Generation Model