

## Hakupuu rajoiteongelmalle P

- Hakupuu
  - solmut rajoiteongelmia
  - juuri P
  - parillisen syvyyden solmuilla tasan yksi perillinen
  - isäsolmu on  $X:n$  suhteen ekvivalentti lasten unionin kanssa
- nimiöintipuut, jakaminen singleton joukkoihin (labeling trees) vs. enumeraatio (bisectio), arvoalueen rajaaminen rajoitteita vyöryttämällä

## Nimiöintipuiden tyypit

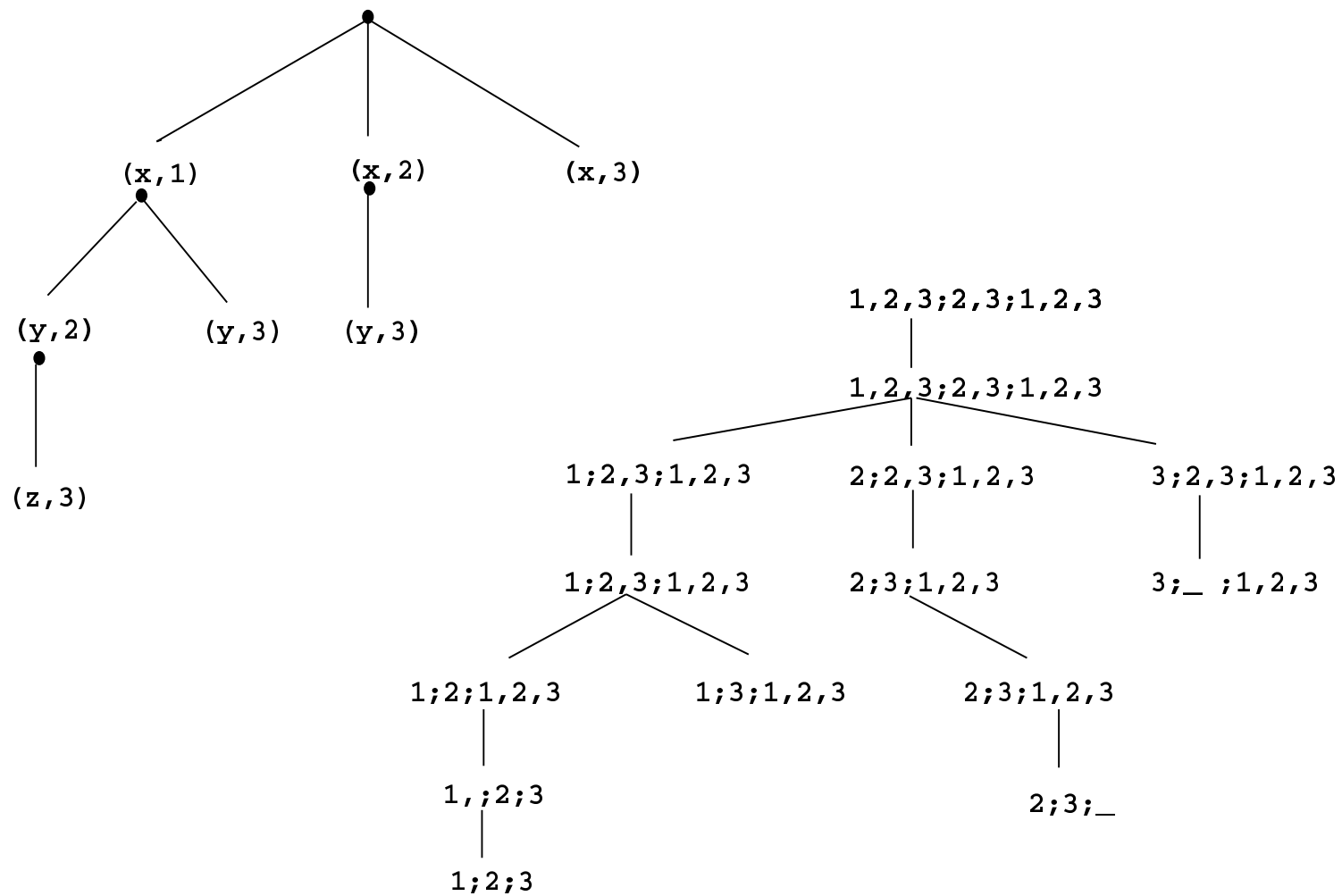
- täydellinen nimiöinti :
  - juuren lapset  $(x_1, d)$
  - $(x_j, d)$  ,  $j \in [1 \dots n - 1]$  lapset ovat muotoa  $(x_{j+1}, e)$
  - haarat täydellisiä instantiaatioita
- **karsitut** nimiöintipuut: haarat kuvaavat kaikki konsistentit instantiaatiot järjestyksen  $x_1 \dots x_n$  mukaan (**reduced labeling trees** )
- **vyörytettävät** nimiöintipuut : kiinnitetty rajoitteiden vyörytysmenetelmä, solmut arvoaluelausekkeita (**prop labelin trees** )

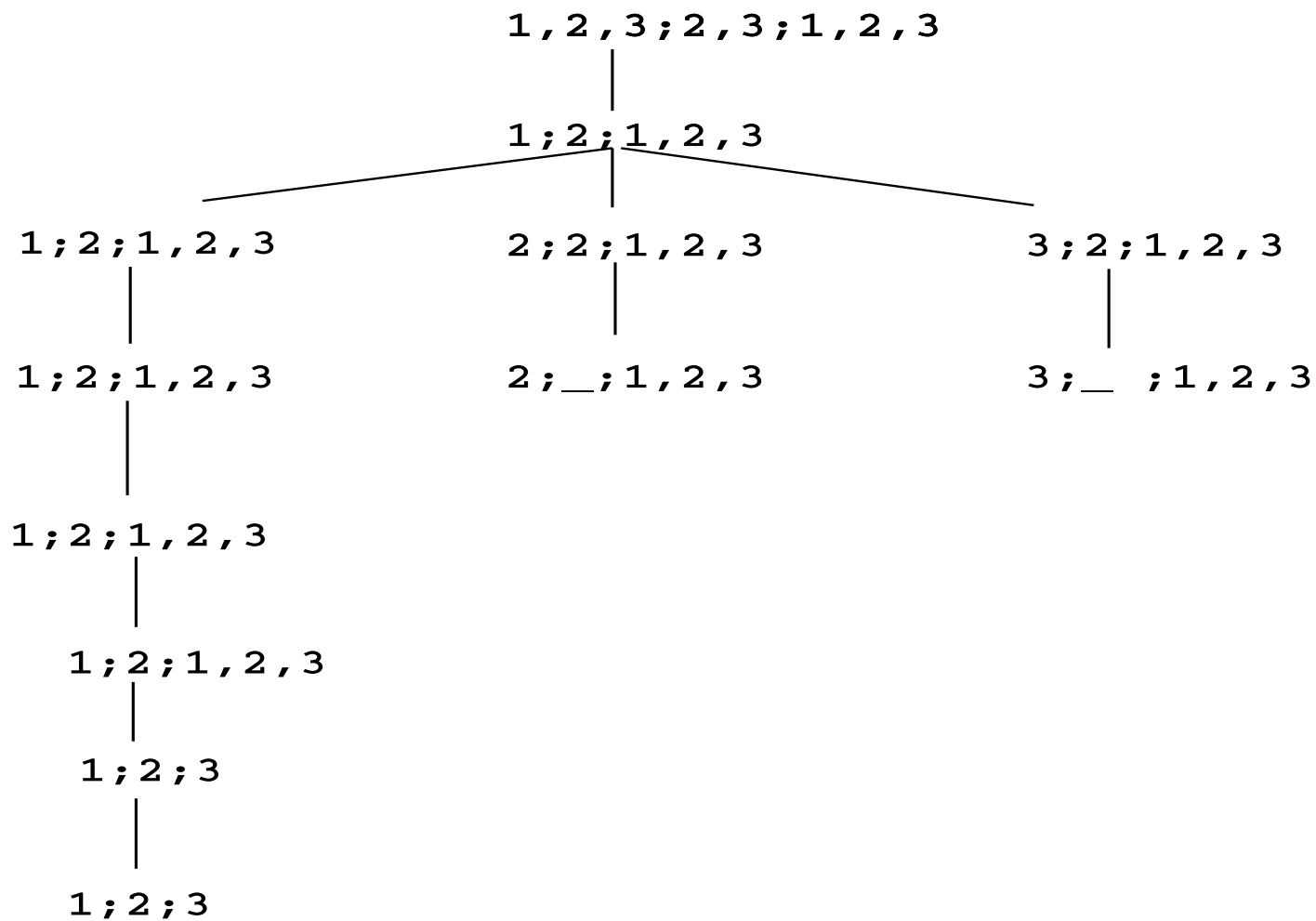
## Vyörytettävät (*prop*) nimimiöintipuut

- kiinnitetty rajoitteiden vyörytysmenetelmä , (eteenpäin karsiva, suunnattukaarikonsistenssi, kaarikonsistenssi)
- solmut arvoaluelausekkeita  $x_1 \in E_1, \dots, x_n \in E_n$ ,
- juuri  $x_1 \in D_1, \dots, x_n \in D_n$
- solmut tasolla  $2i$  muotoa  
 $x_1 \in \{d_1\}, \dots, x_i \in \{d_n\}, x_{i+1} \in E_{i+1}, \dots, x_n \in E_n$
- solmut tasolla  $2i + 1$  muotoa konsistenssin vyörytyksellä kustakin  $2i$  tason solmusta, jos jokin  $E_j = \emptyset, j \in [i + 1, \dots, n]$  solmu on lehtisolmu, muutoin nimiöidään seuraava taso  $x_{i+1}$ :n pohjalta.

## Esimerkki

- $\langle x < y, y < z; x \in \{1, 2, 3\}, y \in \{2, 3\}, z \in \{1, 23\} \rangle$
- karsiva ja vyöryttävä (eteenpäin tarkistava)
- vyöryttävä ( suunnattu konsistenssi)





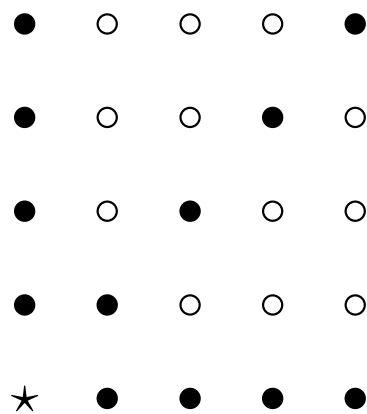
## 5 kuningattaren ongelma

$\langle \mathbb{C}; x_1 \in [1..5], x_2 \in [1..5], x_3 \in [1..5], x_4 \in [1..5], x_5 \in [1..5] \rangle$

○	○	○	○	○
○	○	○	○	○
○	○	○	○	○
○	○	○	○	○
★	○	○	○	○

## eteenpäin karsinta

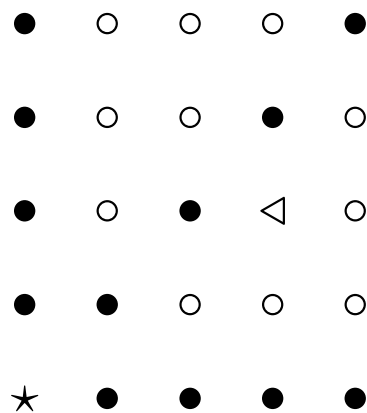
$$x_1 \in 1, x_2 \in \{3, 4, 5\}, x_3 \in \{2, 4, 5\}, x_4 \in \{2, 3, 5\}, x_5 \in \{2, 3, 4\}$$





## suunnattu kaarikonsistenssi

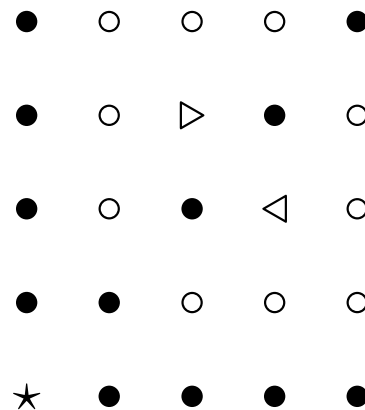
$$x_1 \in \{1\}, x_2 \in \{3, 4, 5\}, x_3 \in \{2, 4, 5\}, x_4 \in \{2, 3, 5\}, x_5 \in \{2, 3, 4\}$$



jokaiseen myöhempään sarakkeeseen jää ruutu

## kaarikonsistenssi

$$x_1 \in 1, x_2 \in \{3, 4, 5\}, x_3 \in \{2, 4, 5\}, x_4 \in \{2, 5\}, x_5 \in \{2, 3, 4\}$$



kaikkien nimiöimättömien suhteen iteroidaan : jos tämä nimiöidään  
näin niin jollekin muulle ei jää nimiöitävää

**Algorithm 0.1:** BACKTRACKFREEPROPMODULE( $C, D$ )

TYPE:  $domains = domain^n$

TYPE:  $instantiation = element[n]$

**procedure** BACKTRACKFREEPROP( $j, D, success$ )

**while**  $D[j] \neq \emptyset$  **and not**  $success$

**do**  $\left\{ \begin{array}{l} d \leftarrow some(D[j]) \\ D[j] \leftarrow D[j] \setminus \{d\} \\ \mathbf{if} \ const(inst, j, d) \end{array} \right.$

**then**  $\left\{ \begin{array}{l} inst[j] \leftarrow d \\ success \leftarrow (j = n) \\ \mathbf{if} \ \mathbf{not} \ success \end{array} \right.$

**then**  $\left\{ \begin{array}{l} prop(j, D, failure) \\ \mathbf{if} \ \mathbf{not} \ failure \\ \mathbf{then} \ j \leftarrow j + 1 \end{array} \right.$

**main**

$inst : instantiation$

$failure : Boolean$

$success \leftarrow FALSE$

$prop(0, D, failure)$

**if not**  $failure$

**then**  $backtrackfreeprop(1, D, success)$

**Algorithm 0.2:** BACKTRACKPROPMODULE( $C, D$ )

TYPE:  $domains = domain^n$

TYPE:  $instantiation = element[n]$

**procedure** BACKTRACKPROP( $j, D, success$ )

**while**  $D[j] \neq \emptyset$  **and not**  $success$

**do**  $\left\{ \begin{array}{l} d \leftarrow some(D[j]) \\ D[j] \leftarrow D[j] \setminus \{d\} \\ \mathbf{if} \ const(inst, j, d) \end{array} \right.$

**then**  $\left\{ \begin{array}{l} inst[j] \leftarrow d \\ success \leftarrow (j = n) \\ \mathbf{if} \ \mathbf{not} \ success \end{array} \right.$

**then**  $\left\{ \begin{array}{l} prop(j, D, failure) \\ \mathbf{if} \ \mathbf{not} \ failure \\ \mathbf{then} \ backtrackprop(j + 1, D, success) \end{array} \right.$

**main**

$inst : instantiation$

$failure : Boolean$

$success \leftarrow FALSE$

$prop(0, D, failure)$

**if not**  $failure$

**then**  $backtrackprop(1, D, success)$

## Teoreemat

- (8.14) **backtrackfree** kerää *inst*-listan alkuun vastaavan **karsitun** nimiöintipuun vasemmanpuoleisen haaran tiedot ja asettaa *success*-muuttujan arvoksi *TRUE* jos tämä haara esittää ratkaisua.
- (8.16) **backtrackfreeprop** kerää *inst*-listan alkuun vastaavan **vyörytettävän** nimiöintipuun vasemmanpuoleisen lehden tiedot ja asettaa *success*-muuttujan arvoksi *TRUE* jos tämä lehti on tasolla  $2n$ .
- (8.17) Jos ratkaisu on olemassa **backtrack** kerää *inst*-listaan vastaavan **karsitun** nimiöintipuun vasemmanpuoleisen, jotain ratkaisua esittävän haaran, tiedot. Muutoin se ilmoittaa epäonnistumisesta asettamalla *success*-muuttujan arvoksi *FALSE*.
- (8.18) Jos ratkaisu on olemassa **backtrackprop** kerää *inst*-listaan vastaavan **vyörytettävän** nimiöintipuun vasemmanpuoleisen, jotain ratkaisua esittävän lehden, tiedot. Muutoin se ilmoittaa epäonnistumisesta asettamalla *success*-muuttujan arvoksi *FALSE*.