§ 8 CONSTRUCTING ES-POLICIES

8.1 LEMMA: H is feasible for $[G, \mathcal{F}]$ iff

(1) $G \prec H$

(2) For every $F \in \mathcal{F}$, there are $i_F, j_F \in F$ with $i_F <_H j_F$

[F is <u>destroyed</u> by $i_F < j_F$,

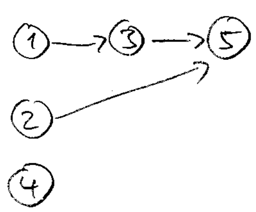
the resource conflict given by F is <u>settled</u> by letting j_F wait for i_F]

Proof: obvious since no $F \in \mathcal{F}$ is an antichain of H □

Idea: solve conflicts on forbidden sets by telling "who must wait for whom"

may be contradictory

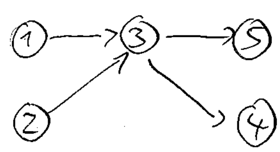
8.2 Example:

G 

$\mathcal{F} = \{ \{2,3\} \ \{3,4\} \ \{1,2,4\} \}$

$\Rightarrow$ $2 \cdot 2 \cdot 6 = 24$ independent choices of $i_F < j_F$

only 9 lead to (feasible) orders

e.g. $2<3$, $3<4$, $1<4$ $\Rightarrow$ 

$2<3$, $3<4$, $4<1$ $\Rightarrow$ contradiction

$E_H \cup \{(2,3)\ (3,4)\ (4,1)\}$ not acyclic

FACT: H + choice of $i_F < j_F$ defines an ES policy

$\iff$ H + choice is acyclic

Organize construction in a tree, the <u>conflict settling tree</u>

root = G

nodes = extensions of G

children of a node H = all extensions H' obtained by settling the conflict on <u>one</u> (yet unsettled) forbidden set

may use a <u>suitable ordering of the forbidden sets</u>

ordering determines the tree!

leaves = feasible orders

8.3 THEOREM: The conflict settling tree is a suborder (not (necessarily induced) of $\mathcal{E}(G)$ containing all minimal feasible orders

Proof: • suborder is trivial.

• not induced: All the order relations $H_1 < H_2$ in the tree are also present in $\mathcal{E}(G)$. But there may be more if there are several $F \in \mathcal{F}$ with $|F| > 2$.

• all minimal feasible orders are contained

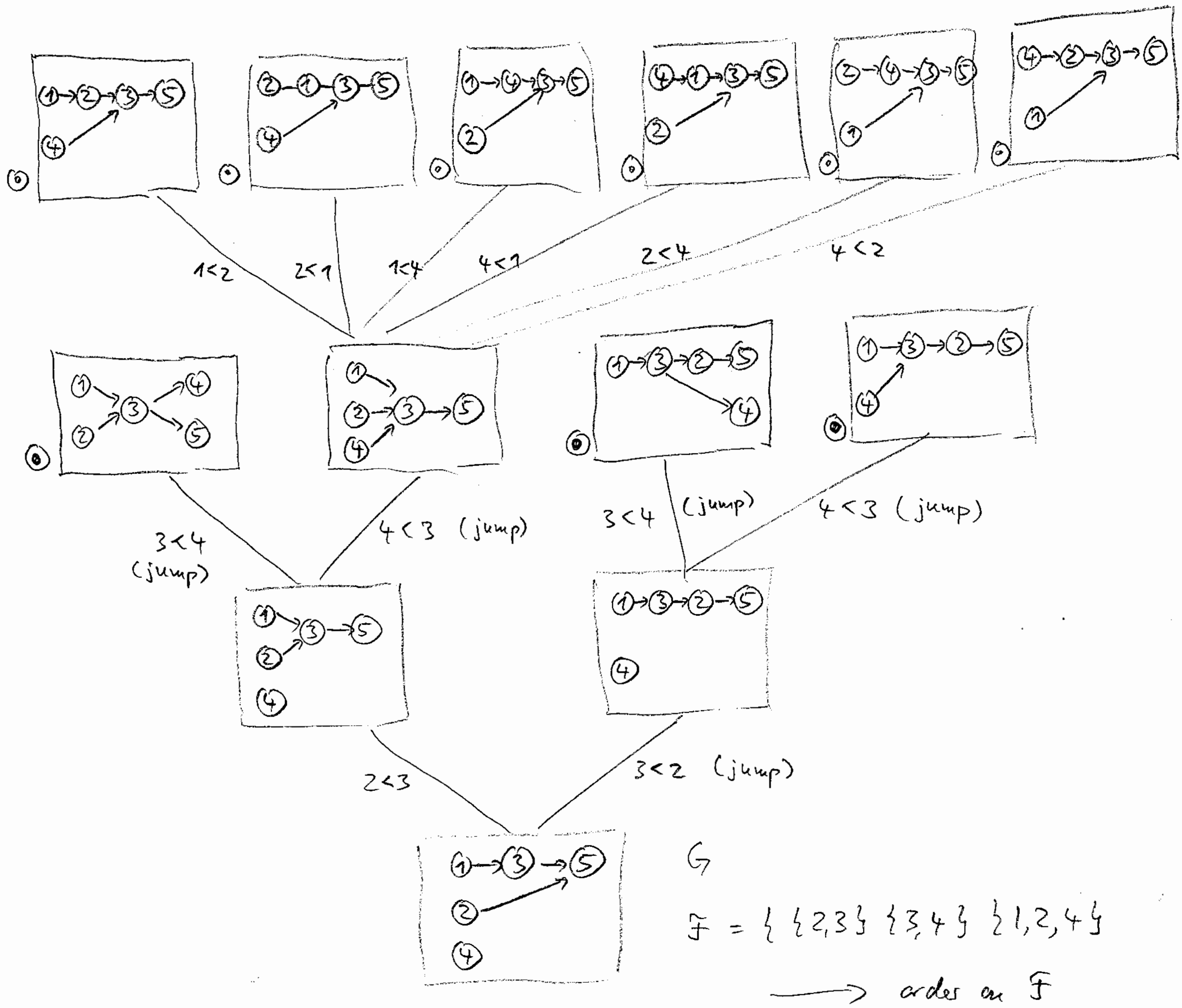
Let H be minimal feasible and $F_1 \dots F_k$ be the order defining the tree

Lemma 8.1 $\Rightarrow$ For every F_r there is some $i_{F_r} < j_{F_r}$ in H

$\Rightarrow$ adding the $i_{F_r} < j_{F_r}$ in the order $1,\dots,k$ yields H $\square$
(forget those induced by transitivity)

Note: $\boxed{\text{NOT every leaf of conflict settling tree is minimal feasible!}}$

Example:



G

$\mathcal{F} = \{ \{2,3\} \{3,4\} \{1,2,4\} \}$

$\longrightarrow$ order on $\mathcal{F}$

Remarks:

Conflict settling tree may be used for Branch & Bound algorithms for calculating an optimal schedule for given k and x

B&B can be combined with dynamic decision model
[Branch only at decision times]
In this way only a subset of all forbidden sets will be considered

8.5 REMARK: In the stochastic case, the optimal values obtained by priority / ES-policies are incomparable

[Remember: in deterministic case, ES is better]

8.6. EXAMPLE:

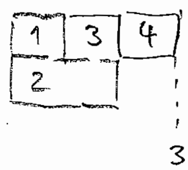
(1) $OPT^{PRIOR} < OPT^{ES}$

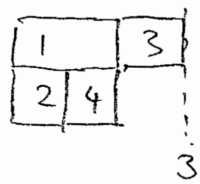
$G:$ $1 \to 3$ $\quad \mathcal{F} = \{\{3,4\}\}$ $\quad \kappa = C_{max}$

$2 \to 4$

$$\left.\begin{array}{l} x^1 = (1, 2, 1, 1) \\ x^2 = (2, 1, 1, 1) \end{array}\right\} \text{ each with prob. } 1/2$$

π priority policy $\Rightarrow$

$\pi[x^1]$: 

$\pi[x^2]$: 

$\Rightarrow E[\kappa^{\pi}] = 3$

minimal feasible orders are

H_1: $1 \to 3 \to 4$, $2 \to 4$

H_2: $1 \to 3$, $2 \to 4 \to 3$

$\Rightarrow E[\kappa^{H_1}] = E[\kappa^{H_2}] = 3.5$

(2) $OPT^{ES} < OPT^{PRIOR}$ already in deterministic case

Exercises:

8.1 Construct an example in which the conflict settling tree contains leaves that are not minimal feasible

8.2 What is the complexity of adding a precedence constraint $i_F < j_F$ to an order H? What is a good data structure for the orders in the conflict settling tree to allow fast updating w.r.t. to adding precedence constraints?