CONCURRENCY

- SHARED MEMORY MODEL
- MESSAGE PASSING MODEL

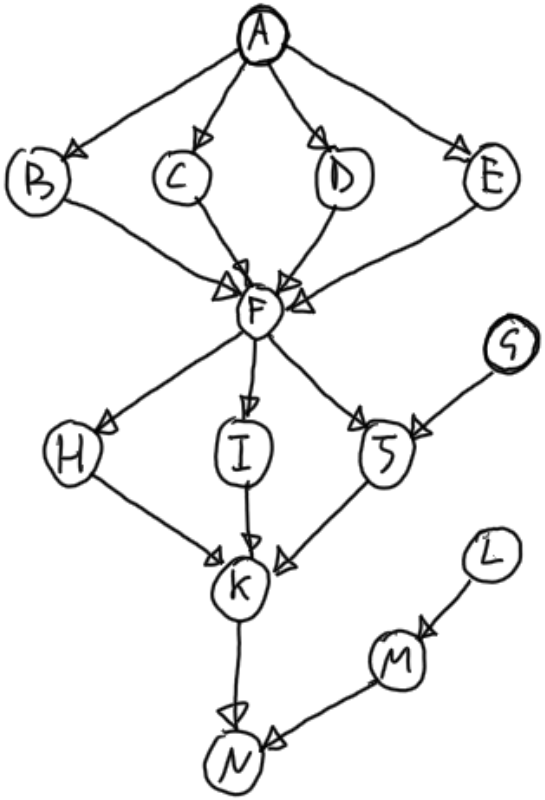
PROCESSES THREADS

|| FORMAL REPRESENTATION
|| OF A (CONCURRENT) COMPUTATION

ACTIONS A

PRECEDENCE CONSTRAINTS PC

A) TAKE A LARGE BOWL
B) SIFT FLOOR INTO BOWL
C) " BAKING POWDER "
D) " SALT "
E) " SUGAR "
F) MAKE A WELL
G) MELT BUTTER
H) POUR IN MILK
I) " " EGG
J) POUR IN MELTED BUTTER
K) MIX
L) OIL FRYING PAN
M) HEAT FRYING PAN
N) POUR IN THE MIXTURE
⋮



DAG

DIRECTED ACYCLIC GRAPH

SET OF RELATIONS ~~OF RELATIONS~~ ATE A THE SO-CALLED HASSE DIAGRAM

SET OF PRECEDENCE CONSTRAINTS PC

PC → PARTIAL ORDER OVER ELEMENTS IN A : $\preceq_P = PC^+$ → TRANSITIVE CLOSURE

NON-STRICT PARTIAL ORDER ($\preceq$)

- REFLEXIVITY $a \preceq a$
- ANTISYMMETRY if $a \preceq b$ and $b \preceq a$ then $a \equiv b$
  - this means if $a \preceq b$ and $a \neq b$ then ~~$b \preceq a$~~ → $\neg\, b \preceq a$
- TRANSITIVITY if $a \preceq b$ and $b \preceq c$ then $a \preceq c$

STRICT PARTIAL ORDER ($\prec$)

- ~~REFLEXIVITY~~ IRREFLEXIVITY $a \not\prec a$
- all the rest of the properties

$\prec_P = PC^+$ A SET OF ACTIONS

$(A, \prec_P)$ POSET

$PC = \{(A,B), (B,C)\}$

$PC^+ = \{(A,B), (B,C), (A,C)\}$

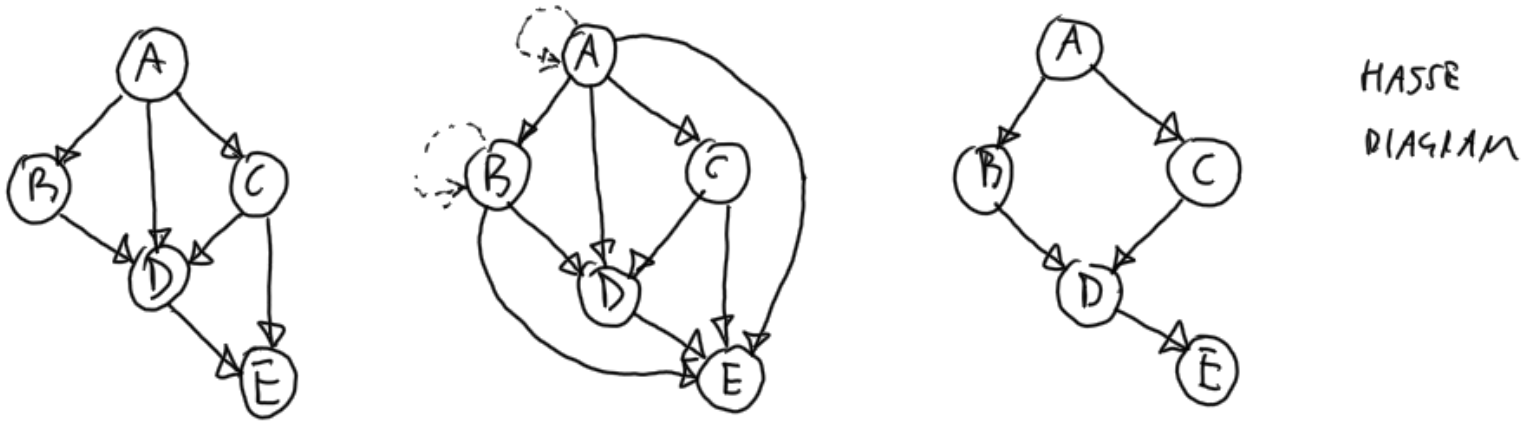
POSET

HASSE DIAGRAM

IDEA: EXPLOIT THE TRANSITIVE REDUCTION OF THE PARTIAL ORDER

I.E. THE SMALLEST R such that $R^+ \equiv PC^+$

FOR DAGs, IT'S UNIQUE



HASSE DIAGRAM

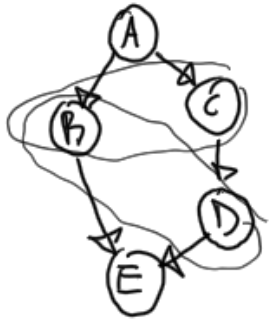
FORMALIZATIO OF THE NOTION OF CONCURRENCY

ACTIONS $a_i \in A$ $a_j \in A$ are said concurrent

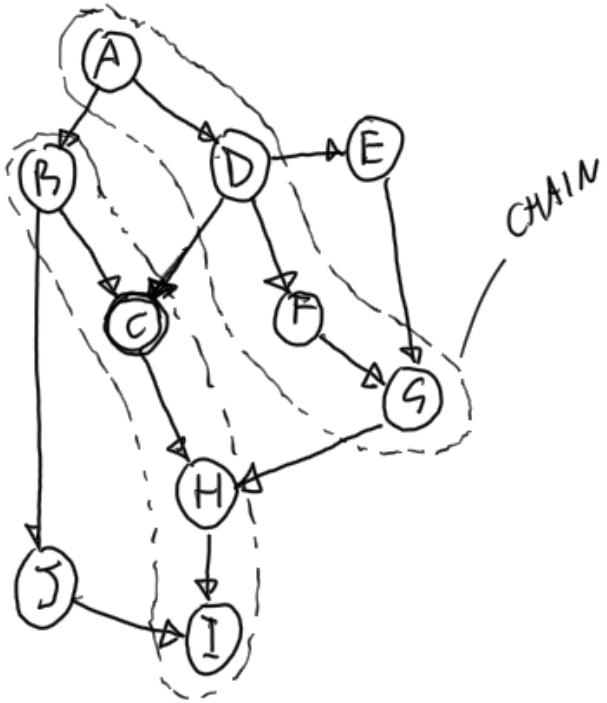
iff neither $a_i \lesssim_p a_j$ nor $a_j \lesssim_p a_i$ holds $\rightsquigarrow a_i \| a_j$

PAY ATTENTION: $\|$ is not a transitive relation!

$B \| C$ $C \| B$ $B \| D$ $\not\Rightarrow$ $C \| D$ FALSE!



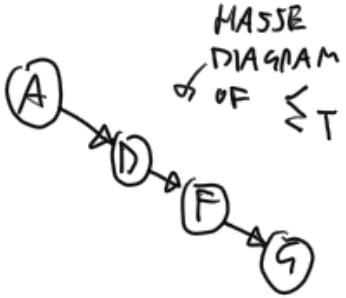
WHAT ABOUT THE NOTION OF "PROCESS"

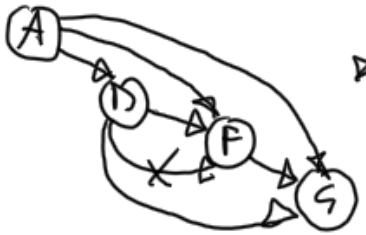


CHAIN: A SEQUENTIAL PATH IN THE DAG

FORMALLY: A SUBSET OF THE POSET, WHOSE PARTIAL ORD. REDUCES TO A TOTAL ORDER

TOTAL ORDER: STRICT PARTIAL ORDER + TOTALITY, i.e. $\forall a_i, a_j \in A$, either $a_i \leq_T a_j$ or $a_j \leq_T a_i$





representation of $\leq_T$

IMPORTANT: LONGEST CHAIN IN THE POSET / HASSE DIAGRAM:

CRITICAL PATH

WE CAN PARTITON THE EXECUTION IN CHAINS, EACH ASSIGNED TO A "WORKER"

→ THE PRECEDENCE CONSTRAINT NOT IN THE CHAIN MUST BE ENFORCED ANYWAY!

→ SYNCRONIZATION MECHANISMS

WHAT IF WE HAVE ONE SINGLE WORKER?

→ ALL THE ACTIVITIES MUST BE ARRANGED IN A TOTAL ORDER, STILL SATISFYING THE P.O.

(↳ ONE OF THE POSSIBLE WAYS)

TOPOLOGICAL SORTING (a.k.a. LINEAR EXTENSION)

SO, if $a_i \leq_P a_j \rightarrow a_i \leq_T a_j$

KAHN'S AGORITHM

START NODES: SN — NO INCOMING ARCS

ALL THE OTHERS: M

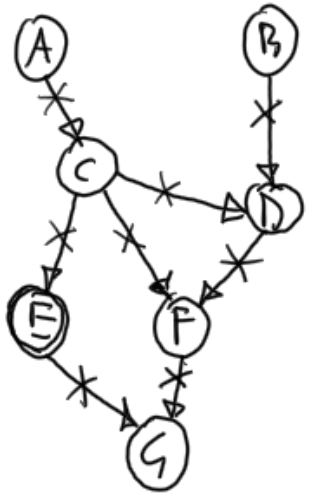
SOLUTION LIST: S — "S" as "SORTED"

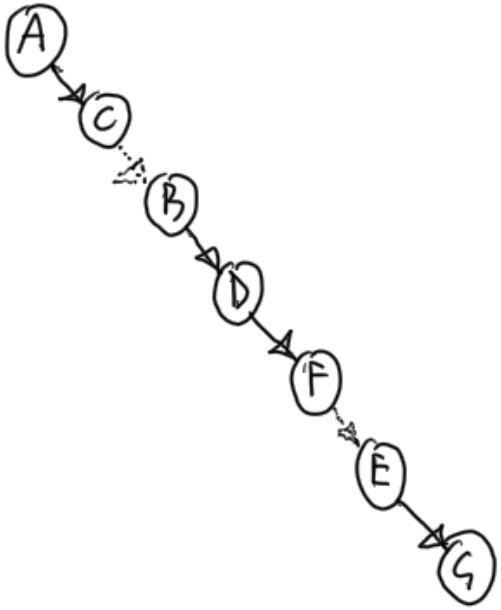
structure:

WHILE SN IS NOT EMPTY:
    PICK A NODE n FROM SN
    PUT IT TO THE TAIL OF S
    FOR EACH NODE m REACHABLE DIRECTLY FROM n BY ARC a:
        REMOVE a FROM THE GRAPH
        IF m HAS NO OTHER INCOMING ACS:
            MOVE m FROM M TO SN
IF GRAPH HAS EDGES → ERROR (NOT A DAG) !
ELSE: RETURN S

KAHN'S ALGORITHM : EXAMPLE

S_N	M	S
$\{A, B\}$	$\{C, D, E, F, G\}$	$[\,]$
$\{B, C\}$	$\{D, E, F, G\}$	$[A]$
$\{B, E\}$	$\{D, F, G\}$	$[A, C]$
$\{E, D\}$	$\{F, G\}$	$[A, C, B]$
$\{E, F\}$	$\{G\}$	$[A, C, B, D]$
$\{E\}$	$\{G\}$	$[A, C, B, D, F]$
$\{G\}$	$\emptyset$	$[A, C, B, D, F, E]$
$\emptyset$		$[A, C, B, D, F, E, G]$





"TRUE" CONCURRENCY VS "INTERLEAVING" CONCURRENCY

LET'S PARALLELIZE!

P – SEQUENTIAL, ORIGINAL PROGRAM P' – IMPROVED, PARALLELIZED VERSION

$T(P)$ runtime of P $T(P')$ runtime of P' HOPEFULLY, $T(P') \leq T(P)$

SPEED-UP $\sigma = \frac{T(P)}{T(P')}$

$T(P)$ ⇝ TWO DIFFERENT PORTIONS: $T(P) = S + P$ (purely sequential; parallelizable)

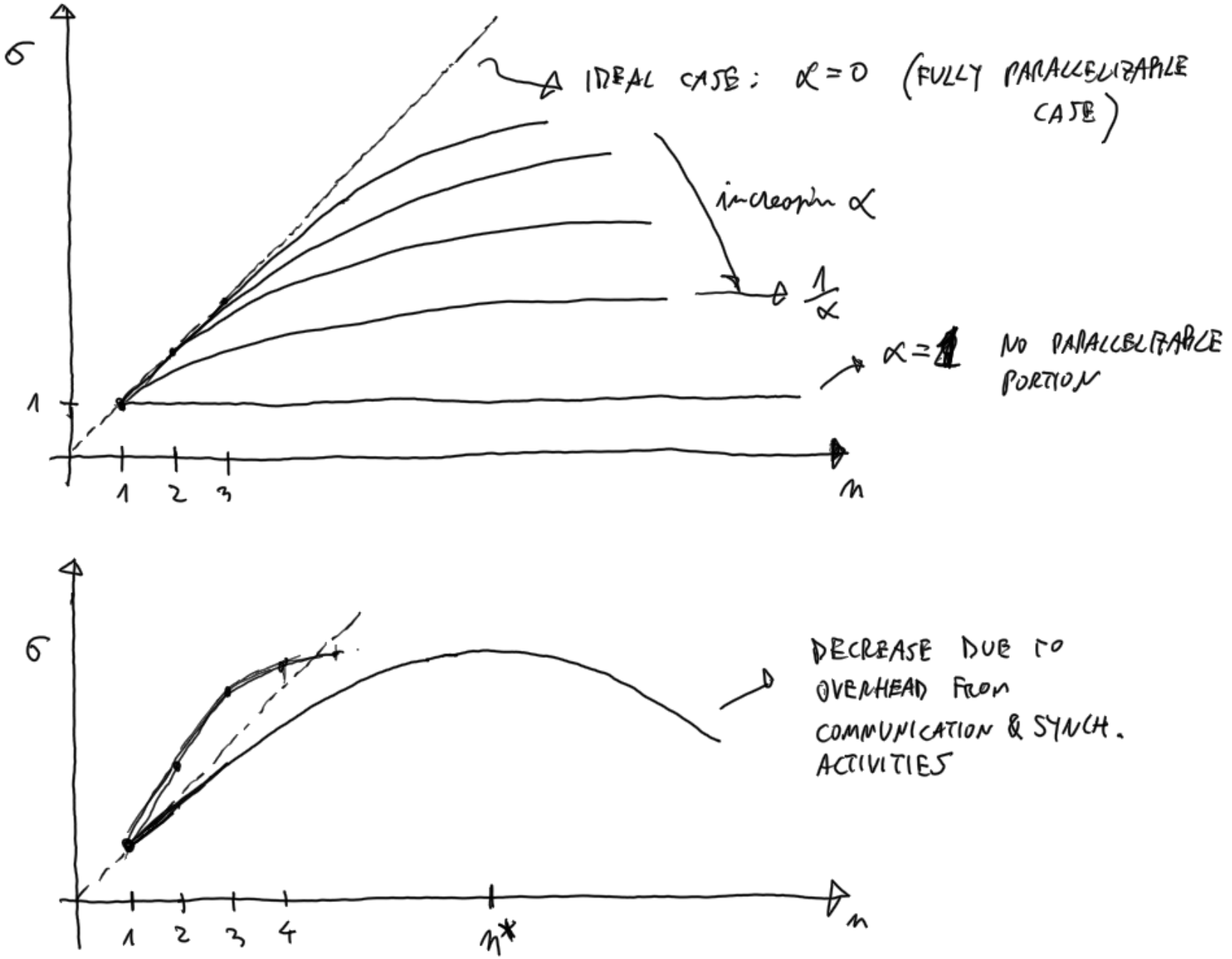
SEQUENTIAL PORTION: $\alpha = \frac{S}{S+P}$

SUPPOSE WE HAVE n COMPUTING UNITS, P ⇝ (can be reduced) $\frac{P}{n}$; SO, THE SPEED-UP IS

$$\sigma(n, \alpha) = \frac{1}{\alpha + \frac{1-\alpha}{n}}$$

$$\lim_{n \to \infty} \sigma(n, \alpha) = \frac{1}{\alpha}$$

AMDAHL'S LAW


G
IDEAL CASE: $\alpha = 0$ (FULLY PARALLELIZABLE CASE)
increasing α
$\frac{1}{\alpha}$
$\alpha = 1$
NO PARALLELIZABLE PORTION
1
1 2 3
n
G
DECREASE DUE TO OVERHEAD FROM COMMUNICATION & SYNCH. ACTIVITIES
1 2 3 4
n^*
n