

$$1, 1$$

$$a \xrightarrow{ab} b$$

$$ac \downarrow$$

$$c$$

$$\text{If } \text{Task}(ab) = \text{Task}(ac)$$

Then:  $ab = ac$  by Task Reduction Uniqueness

Else:  $a \xrightarrow{ab} b$

$$\begin{array}{ccc} \downarrow ac & & \downarrow ac \\ c \xrightarrow{ab} d & & \end{array}$$

by Commutativity

$$X, Y = Y, X$$

by renaming

$X+1, Y$  with IHs  $X, Y$  and  $1, Y$

$$\begin{array}{ccccc} a & \xrightarrow{ab^x} & b & \xrightarrow{bb'} & b' \\ ac^y \downarrow & \boxed{x} & \downarrow & \boxed{y'} & \downarrow \\ c & \xrightarrow{\quad} & d & \xrightarrow{\quad} & d' \end{array}$$

$$\text{IH: } \forall X', Y' . X' \leq X \wedge Y' \leq Y \Rightarrow$$

$$\forall a, b, c \in S. a \xrightarrow{ab^x} b \Rightarrow \exists d \in S. b \xrightarrow{bd^*} d \wedge c \xrightarrow{cd^*} d$$

$$\begin{array}{c} ac^y \downarrow \\ c \end{array}$$

$$\wedge |bd^*| \leq X'$$

$$\wedge |cd^*| \leq Y'$$

$$\text{IH: } X, Y \quad \begin{array}{ccc} a & \xrightarrow{ab^x} & b \\ ac^* \downarrow & & \downarrow bd^* \\ c & \xrightarrow{cd^*} & d \end{array}$$

$$\text{IH: } 1, Y \quad \begin{array}{ccc} b & \xrightarrow{bb'} & b' \\ bd^* \downarrow & & \downarrow b'd'^* \\ d & \xrightarrow{dd'^*} & d' \end{array}$$

$$\Rightarrow \begin{array}{ccccc} & & ab^x & & \\ & & \curvearrowright & & \\ a & \xrightarrow{\quad} & b & \xrightarrow{\quad} & b' \\ ac^y \downarrow & & \downarrow bd^* & & \downarrow b'd'^* \\ c & \xrightarrow{\quad} & d & \xrightarrow{\quad} & d' \\ & & \curvearrowleft cd'^* & & \end{array}$$

with  $|cd'^*| = |cd^*| + |dd'^*| \leq |ab^x| + |bb'| = X+1$

$$|b'd'^*| \leq |bd^*| \leq |ac^y| = Y$$

□

Induction completeness:

$X, Y$  reached by  $1, 1 \xrightarrow[(X, Y) \rightarrow (X+1, Y)]{(X-1) \text{ times}} Y, 1 \xrightarrow[\text{symm}]{X-1 \text{ times}} 1, Y \rightarrow X, Y$   
for  $X, Y \geq 1$

Task Reduction Uniqueness

Proof by case distinction on term syntax:

Always exactly one rule applicable

Commutativity:

$$\begin{array}{ccc} H_1, TS_1, WS_1 & \xrightarrow{E-X} & H_{12}, TS_{12}, WS_{12} \\ \uparrow \xrightarrow{E-Y} & & \uparrow \\ H, TS, WS & & H_2, TS_2, WS_2 \\ \downarrow \xrightarrow{E-X} & & \downarrow \\ & & H_{21}, TS_{21}, WS_{21} \end{array}$$
  
$$= ? \quad = ? \quad = ?$$

X and Y affect different tasks

Both can be reduced in the starting state

Taskdrops via Capture etc. affect only waiting tasks

Line 2+3  $\Rightarrow$  Taskdrops are irrelevant for this

Creating new tasks

- Proof that ids are different  
(otherwise finish blocks)

Changing heap

- Only own region

- Not affected by changes in other region

- Tasks are all in different regions (ISO)

0) E-X reduces exactly one ready task

$\exists T_x \in TS$ . E-X doesn't reduce  $H, TS \setminus \{T_x\}$

$T_x \in TS_2$  because E-Y reduces  $T_y \neq T_x \wedge$

$T_x \neq \text{FINISH}$  f with  $(f, k, FS) \in TS \Rightarrow T_x$  is not dropped by E-Box etc.

(1) Task affects only own region:

$H, \{T\} \uplus TS, WS \rightsquigarrow H', \{T'\} \uplus TS, WS$

$\Rightarrow H = H' \vee \exists o, v. H[o \rightarrow v] = H' \wedge \text{accRoot}(o, T')$

by case distinction on " $\rightsquigarrow$ "

(2) Task unaffected by Heap changes in other regions:

$H, \{T\} \uplus TS, WS \rightsquigarrow H', \{T'\} \uplus TS, WS \wedge \text{isolation}(H, \{T\} \uplus TS, WS) \wedge q. \text{accRoot}(q, T)$

$\Rightarrow H[q' \rightarrow v], \{T\} \uplus TS, WS \rightsquigarrow H'[q' \rightarrow v], \{T'\} \uplus TS, WS$   
 $\Rightarrow \neg \text{reach}(H, q, q')$

Cases of " $\rightsquigarrow$ "

NULL, VAR, SELECT, ASSIGN, INVOKE, RETURN+2, OPEN trivial because

NEW: Assume  $o \neq q'$  WLOG

$a \neq b \Rightarrow H[a \rightarrow x][b \rightarrow y] = H[b \rightarrow y][a \rightarrow x]$

BOX, CAPTURE, SWAP

stack dropping unaffected by  $[q' \rightarrow v]$

ASYNC etc.

Heap unused

$$H, \{T_1, T_2\} \wp TS \leadsto H', \{\bar{T}_1', T_2\} \wp TS \quad \wedge \quad H, \{T_1, T_2\} \wp TS \leadsto H'', \{\bar{T}_1, \bar{T}_2'\} \wp TS$$

$$\Rightarrow H, \{T_1, \bar{T}_2'\} \wp TS \leadsto H', \{\bar{T}_1', \bar{T}_2'\} \wp TS$$

Reduction of  $T_2$  does not change its parent-id f. As  $T_1$  could be

0, 1, 2 can be used to show the lemma

Not rigorous yet.

Think about how to best handle all the edge-cases (E-Task-Doze e.g.)