$1,1$

$$\begin{tikzcd} a \arrow[r, "ab"] \arrow[d, "ac"'] & b \\ c & \end{tikzcd}$$

If $Task(ab) = Task(ac)$

Then: $ab = ac$ by TaskReductionUniqueness

Else:
$$\begin{tikzcd} a \arrow[r, "ab"] \arrow[d, "ac"'] & b \arrow[d, "ac"] \\ c \arrow[r, "ab"'] & d \end{tikzcd}$$
by Commutativity

$X,Y = Y,X$

by renaming

$X+1, Y$ with IHs $\underline{X,Y}$ and $\underline{1,Y}$

$$\begin{tikzcd} a \arrow[r, "ab^X"] \arrow[d, "ac^Y"'] \arrow[dr, phantom, "\boxed{\substack{X\\Y}}"] & b \arrow[r, "bb'"] \arrow[d] \arrow[dr, phantom, "\boxed{\substack{1\\Y}}"] & b' \arrow[d] \\ c \arrow[r] & d \arrow[r] & d' \end{tikzcd}$$

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

$\forall a,b,c \in S.$
$$\begin{tikzcd} a \arrow[r, "ab^{X'}"] \arrow[d, "ac^{Y'}"'] & b \\ c & \end{tikzcd}$$
$\Rightarrow \exists\, d \in S.\ b \xrightarrow{bd^*} d \wedge c \xrightarrow{cd^*} d$

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

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

IH: X,Y
$$\begin{tikzcd} a \arrow[r, "ab^*"] \arrow[d, "ac^*"'] & b \arrow[d, "bd^*"] \\ c \arrow[r, "cd^*"'] & d \end{tikzcd}$$

IH: $1,Y$
$$\begin{tikzcd} b \arrow[r, "bb'"] \arrow[d, "bd^*"'] & b' \arrow[d, "b'd'^*"] \\ d \arrow[r, "dd'^*"'] & d' \end{tikzcd}$$

$$\Rightarrow \begin{tikzcd} a \arrow[r] \arrow[rr, bend left, "ab'^X"] \arrow[d, "ac^Y"'] & b \arrow[r] \arrow[d, "bd^*"] & b' \arrow[d, "b'd'^*"] \\ c \arrow[r] \arrow[rr, bend right, "cd'^*"'] & d \arrow[r] & d' \end{tikzcd}$$

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

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

$\square$

Induction completeness:

X,Y reached by $\quad 1,1 \xrightarrow[(x,y)\to(x+1,y)]{(Y-1)\text{ times}} Y,1 \xrightarrow{\text{symm}} 1,Y \xrightarrow{X-1\text{ times}} X,Y$

for $X,Y \geq 1$

Task Reduction Uniqueness

Proof by case distinction on term syntax:

Always exactly one rule applicable

Commutativity:

$$\begin{tikzcd}
 & H_1, TS_1 \arrow[r, squiggly, "E\text{-}Y"] & H_{12}, TS_{12} \arrow[d, phantom, "= \quad = ?"] \\
H, TS \arrow[ur, squiggly, "E\text{-}X"] \arrow[r, squiggly, "E\text{-}Y"'] & H_2, TS_2 \arrow[r, squiggly, "E\text{-}X"] & H_{21}, TS_{21}
\end{tikzcd}$$

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 => 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 Task T_X is the task reduced by E-X:

$\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:

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

$\Rightarrow H = H' \vee \exists o, v.\quad H[o \to v] = H' \wedge accRoot(o, T')$

by case distinction on "$\rightsquigarrow$"

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

$H, \{T\} \uplus TS \rightsquigarrow H', \{T'\} \uplus TS \ \wedge\ isolation(H, \{T\} \uplus TS) \wedge \forall q.\ accRoot(q, T)$
$\Rightarrow \neg reach(H, q, q')$

$\Rightarrow H[q' \to v], \{T\} \uplus TS \rightsquigarrow H'[q' \to v], \{T'\} \uplus TS$

Cases of "$\rightsquigarrow$"

NULL, VAR, SELECT, ASSIGN, INVOKE, RETURN1+2, OPEN trivial because

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

NEW: Assume $o \neq q'$ WLOG

BOX, CAPTURE, SWAP

Stack dropping unaffected by $[q' \to v]$

ASYNC etc.

Heap unused

$$H, \{T_1, T_2\} \uplus TS \rightsquigarrow H', \{T_1', T_2\} \uplus TS \quad \wedge \quad H, \{T_1, T_2\} \uplus TS \rightsquigarrow H'', \{T_1, T_2'\} \uplus TS$$

$$\Rightarrow \quad H, \{T_1, T_2'\} \uplus TS \rightsquigarrow H', \{T_1', T_2'\} \uplus TS$$

Reduction of T2 does not change its parent-id f. As T1 could be

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