Reasoning about correctness of transactional memory with operational semantics

Andri Saar
(Joint work with Tarmo Uustalu)

Institute of Cybernetics at
Tallinn University of Technology

Joint Estonian-Latvian Theory Days, 3 Oct 2010

Grammar

$$
\begin{aligned}
a &::= a_1 + a_2 \mid a_1 - a_2 \mid a_1 * a_2 \\
b &::= a_1 < a_2 \mid a_1 = a_2 \mid \neg b \mid b_1 \vee b_2 \mid b_1 \wedge b_2 \\
s &::= x := a \mid \texttt{if}\ b\ \texttt{then}\ s_1\ \texttt{else}\ s_2 \mid \texttt{while}\ b\ \texttt{do}\ s \mid \texttt{skip} \\
&\mid s_1 ; s_2
\end{aligned}
$$

Grammar

$$
\begin{aligned}
a ::= &\ a_1 + a_2 \mid a_1 - a_2 \mid a_1 * a_2 \\
b ::= &\ a_1 < a_2 \mid a_1 = a_2 \mid \neg b \mid b_1 \vee b_2 \mid b_1 \wedge b_2 \\
s ::= &\ x := a \mid \texttt{if}\ b\ \texttt{then}\ s_1\ \texttt{else}\ s_2 \mid \texttt{while}\ b\ \texttt{do}\ s \mid \texttt{skip} \\
\mid &\ s_1 ; s_2 \mid s_1 \parallel s_2 \mid \texttt{trans}\ s
\end{aligned}
$$

Grammar

$$
\begin{aligned}
a &::= a_1 + a_2 \mid a_1 - a_2 \mid a_1 * a_2 \\
b &::= a_1 < a_2 \mid a_1 = a_2 \mid \neg b \mid b_1 \vee b_2 \mid b_1 \wedge b_2 \\
s &::= x := a \mid \texttt{if } b \texttt{ then } s_1 \texttt{ else } s_2 \mid \texttt{while } b \texttt{ do } s \mid \texttt{skip} \\
&\mid s_1; s_2 \mid s_1 \parallel s_2 \mid \texttt{trans } s
\end{aligned}
$$

At every small step we are free to choose whether we proceed one step with s_1 or s_2 and the whole statement will not terminate until both branches have terminated.

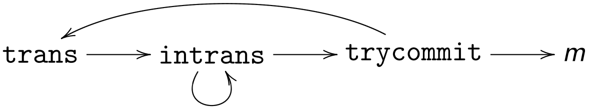
Grammar

$$
\begin{aligned}
a &::= a_1 + a_2 \mid a_1 - a_2 \mid a_1 * a_2 \\
b &::= a_1 < a_2 \mid a_1 = a_2 \mid \neg b \mid b_1 \vee b_2 \mid b_1 \wedge b_2 \\
s &::= x := a \mid \texttt{if } b \texttt{ then } s_1 \texttt{ else } s_2 \mid \texttt{while } b \texttt{ do } s \mid \texttt{skip} \\
&\mid s_1 ; s_2 \mid s_1 \parallel s_2 \mid \texttt{trans } s
\end{aligned}
$$

Transactions should be executed atomically without interference from the surrounding code. Unsuccessful transactions should not be visible.

"Weak" model

$$
\begin{aligned}
xs_W ::=&\ x := a \mid \texttt{if}\ b\ \texttt{then}\ s_1\ \texttt{else}\ s_2 \mid \texttt{while}\ b\ \texttt{do}\ s \mid \texttt{skip} \\
&\mid xs_W ; s \mid xs_{W1} \parallel xs_{W2} \mid \texttt{trans}\ s \\
&\mid \texttt{intrans}\ xs_W\ s\ wc\ rc \mid \texttt{trycommit}\ s\ wc\ rc \\
m_W ::=&\ m \mid wc : rc : m_W \\
cfg_W ::=&\ m_W \mid \langle xs_W , m_W \rangle
\end{aligned}
$$



Some helper functions for the weak model

$$\text{read}(x, c : m) = \begin{cases} c\ x & \text{if } x \in \text{dom } c \\ \text{read}(x, m) & \text{otherwise} \end{cases}$$

$$\text{write}\,(c, wc : m) = \left[\begin{array}{ll} c\ x & \text{if } x \in \text{dom } c \\ wc\ x & \text{otherwise} \end{array} \;\middle|\; x \right] : m$$

$$\text{rcupdate}(c, wc : rc : m) = wc : \left[\begin{array}{ll} c\ x & \text{if } x \notin \text{dom } rc \wedge \\ & \quad x \notin \text{dom } wc \\ rc\ x & \text{otherwise} \end{array} \;\middle|\; x \right] : m$$

Operational semantics for the "weak" model (excerpt)

$$\frac{}{\langle x, m_W\rangle \rightarrow_{WA} \langle \mathsf{read}(x, m_W), \mathsf{rcupdate}([x \mapsto v], m_W)\rangle}$$

$$\frac{\langle a, m_W\rangle \rightarrow_{WA} \langle v, m'_W\rangle}{\langle x := a, m_W\rangle \rightarrow_W \mathsf{write}\left([x \mapsto v], m'_W\right)}$$

$$\frac{}{\langle \texttt{trans}\ s, m_W\rangle \rightarrow_W \langle \texttt{intrans}\ s\ s\ []\ []\ , m_W\rangle}$$

$$\frac{\langle xs_W, wc : rc : m_W\rangle \rightarrow_W \langle xs'_W, wc' : rc' : m_W\rangle}{\langle \texttt{intrans}\ xs_W\ s\ wc\ rc, m_W\rangle \rightarrow_W \langle \texttt{intrans}\ xs'_W\ s\ wc'\ rc', m_W\rangle}$$

$$\frac{\langle xs_W, wc : rc : m_W\rangle \rightarrow_W wc' : rc' : m_W}{\langle \texttt{intrans}\ xs_W\ s\ wc\ rc, m_W\rangle \rightarrow_W \langle \texttt{trycommit}\ s\ wc'\ rc', m_W\rangle}$$

$$\frac{}{\langle \texttt{trycommit}\ s\ wc\ rc, m_W\rangle \rightarrow_W \langle \texttt{trans}\ s, m_W\rangle}$$

$$\frac{\forall x \in \mathsf{dom}\ rc\ \ \mathsf{read}(x, m_W) = rc\ x}{\langle \texttt{trycommit}\ s\ wc\ rc, m_W\rangle \rightarrow_W \mathsf{write}\left(wc, \mathsf{rcupdate}(rc, m_W)\right)}$$

Example evaluation

Program

$\texttt{trans}\ (y := x; x := 7)$

$||$

$x := 5; \texttt{trans}\ (z := x; x := 1)$

$\begin{bmatrix} x\ y\ z \\ 0\ 0\ 0 \end{bmatrix}$

$$\frac{}{\langle \texttt{trans}\ s, m\rangle \xrightarrow[\mathrm{W}]{} \left\langle \texttt{intrans}\ s\ s \left[\ \right] \left[\ \right], m \right\rangle}$$

Example evaluation

Program

$\texttt{intrans}\ (y := x; x := 7)\ (y := x; x := 7)\ \Big[\ \Big]\ \Big[\ \Big]$

$\|$

$x := 5; \texttt{trans}\ (z := x; x := 1)$

$$\begin{bmatrix} x\ y\ z \\ 0\ 0\ 0 \end{bmatrix}$$

$$\dfrac{\dfrac{\langle 5, m\rangle \underset{\text{WA}}{\longrightarrow} \langle 5, m\rangle \quad m' = \text{write}\left(\begin{bmatrix} x \\ 5 \end{bmatrix}, m\right)}{\langle x := 5, m\rangle \underset{\text{W}}{\longrightarrow} m'}}{\langle x := 5; s, m\rangle \underset{\text{W}}{\longrightarrow} \langle s, m'\rangle}$$

Example evaluation

Program

$\texttt{intrans}\ (y := x; x := 7)\ (y := x; x := 7)\ \begin{bmatrix} \end{bmatrix} \begin{bmatrix} \end{bmatrix}$

$\|$

$\texttt{trans}\ (z := x; x := 1)$

$\begin{bmatrix} x & y & z \\ 5 & 0 & 0 \end{bmatrix}$

$$\dfrac{\dfrac{\dfrac{\left\langle x, \begin{bmatrix} \end{bmatrix} : \begin{bmatrix} \end{bmatrix} : m \right\rangle \xrightarrow[\text{WA}]{} \left\langle 5, \begin{bmatrix} \end{bmatrix} : \begin{bmatrix} x \\ 5 \end{bmatrix} : m \right\rangle \quad \begin{bmatrix} y \\ 5 \end{bmatrix} : \begin{bmatrix} x \\ 5 \end{bmatrix} : m = \text{write}\left(\begin{bmatrix} y \\ 5 \end{bmatrix}, \begin{bmatrix} \end{bmatrix} : \begin{bmatrix} x \\ 5 \end{bmatrix} : m \right)}{\left\langle y := x, \begin{bmatrix} \end{bmatrix} : \begin{bmatrix} \end{bmatrix} : m \right\rangle \xrightarrow[\text{W}]{} \begin{bmatrix} y \\ 5 \end{bmatrix} : \begin{bmatrix} x \\ 5 \end{bmatrix} : m}}{\left\langle y := x; s_2, \begin{bmatrix} \end{bmatrix} : \begin{bmatrix} \end{bmatrix} : m \right\rangle \xrightarrow[\text{W}]{} \left\langle s_2, \begin{bmatrix} y \\ 5 \end{bmatrix} : \begin{bmatrix} x \\ 5 \end{bmatrix} : m \right\rangle}}{\left\langle \texttt{intrans}\ y := x; s_2\ s_1 \begin{bmatrix} \end{bmatrix} \begin{bmatrix} \end{bmatrix}, m \right\rangle \xrightarrow[\text{W}]{} \left\langle \texttt{intrans}\ s_2\ s_1 \begin{bmatrix} y \\ 5 \end{bmatrix} \begin{bmatrix} x \\ 5 \end{bmatrix}, m \right\rangle}$$

Example evaluation

Program

$\texttt{intrans}\ (x := 7)\ (y := x; x := 7) \begin{bmatrix} y \\ 5 \end{bmatrix} \begin{bmatrix} x \\ 5 \end{bmatrix}$

$\|$

$\texttt{trans}\ (z := x; x := 1)$

$\begin{bmatrix} x\ y\ z \\ 5\ 0\ 0 \end{bmatrix}$

$$\dfrac{\dfrac{\left\langle 7, \begin{bmatrix} y \\ 5 \end{bmatrix} : \begin{bmatrix} x \\ 5 \end{bmatrix} : m \right\rangle \xrightarrow[\text{WA}]{} \left\langle 7, \begin{bmatrix} y \\ 5 \end{bmatrix} : \begin{bmatrix} x \\ 5 \end{bmatrix} : m \right\rangle \quad \begin{bmatrix} x\ y \\ 7\ 5 \end{bmatrix} : \begin{bmatrix} x \\ 5 \end{bmatrix} : m = \text{write}\left(\begin{bmatrix} x \\ 7 \end{bmatrix}, \begin{bmatrix} y \\ 5 \end{bmatrix} : \begin{bmatrix} x \\ 5 \end{bmatrix} : m \right)}{\left\langle x := 7, \begin{bmatrix} y \\ 5 \end{bmatrix} : \begin{bmatrix} x \\ 5 \end{bmatrix} : m \right\rangle \xrightarrow[\text{W}]{} \begin{bmatrix} x\ y \\ 7\ 5 \end{bmatrix} : \begin{bmatrix} x \\ 5 \end{bmatrix} : m}}{\left\langle \texttt{intrans}\ (x := 7)\ s \begin{bmatrix} y \\ 5 \end{bmatrix} \begin{bmatrix} x \\ 5 \end{bmatrix}, m \right\rangle \xrightarrow[\text{W}]{} \left\langle \texttt{trycommit}\ s\ \begin{bmatrix} x\ y \\ 7\ 5 \end{bmatrix} \begin{bmatrix} x \\ 5 \end{bmatrix}, m \right\rangle}$$

Example evaluation

Program

$\texttt{trycommit}\ (y := x; x := 7)\ \begin{bmatrix} x\ y \\ 7\ 5 \end{bmatrix} \begin{bmatrix} x \\ 5 \end{bmatrix}$

$\|$

$\texttt{trans}\ (z := x; x := 1)$

$\begin{bmatrix} x\ y\ z \\ 5\ 0\ 0 \end{bmatrix}$

$$\frac{}{\langle \texttt{trans}\ s, m \rangle \underset{\mathrm{W}}{\longrightarrow} \left\langle \texttt{intrans}\ s\ s \left[\ \right] \left[\ \right], m \right\rangle}$$

Example evaluation

Program

$$\texttt{trycommit}\ (y := x; x := 7)\ \begin{bmatrix} x\,y \\ 7\,5 \end{bmatrix} \begin{bmatrix} x \\ 5 \end{bmatrix}$$

$$\|$$

$$\texttt{intrans}\ (z := x; x := 1)\ (z := x; x := 1)\ \begin{bmatrix} \ \end{bmatrix} \begin{bmatrix} \ \end{bmatrix}$$

$$\begin{bmatrix} x\,y\,z \\ 5\,0\,0 \end{bmatrix}$$

$$\dfrac{\dfrac{\dfrac{\left\langle x, \begin{bmatrix} \ \end{bmatrix} : \begin{bmatrix} \ \end{bmatrix} : m \right\rangle \xrightarrow[\text{WA}]{} \left\langle 5, \begin{bmatrix} \ \end{bmatrix} : \begin{bmatrix} x \\ 5 \end{bmatrix} : m \right\rangle \quad \begin{bmatrix} z \\ 5 \end{bmatrix} : \begin{bmatrix} x \\ 5 \end{bmatrix} : m = \text{write}\left(\begin{bmatrix} z \\ 5 \end{bmatrix}, \begin{bmatrix} \ \end{bmatrix} : \begin{bmatrix} x \\ 5 \end{bmatrix} : m \right)}{\left\langle z := x, \begin{bmatrix} \ \end{bmatrix} : \begin{bmatrix} \ \end{bmatrix} : m \right\rangle \xrightarrow[\text{W}]{} \begin{bmatrix} z \\ 5 \end{bmatrix} : \begin{bmatrix} x \\ 5 \end{bmatrix} : m}}{\left\langle z := x; s_2, \begin{bmatrix} \ \end{bmatrix} : \begin{bmatrix} \ \end{bmatrix} : m \right\rangle \xrightarrow[\text{W}]{} \left\langle s_2, \begin{bmatrix} z \\ 5 \end{bmatrix} : \begin{bmatrix} x \\ 5 \end{bmatrix} : m \right\rangle}}{\left\langle \texttt{intrans}\ z := x; s_2\ s_1 \begin{bmatrix} \ \end{bmatrix} \begin{bmatrix} \ \end{bmatrix}, m \right\rangle \xrightarrow[\text{W}]{} \left\langle \texttt{intrans}\ s_2\ s_1 \begin{bmatrix} z \\ 5 \end{bmatrix} \begin{bmatrix} x \\ 5 \end{bmatrix}, m \right\rangle}$$

Example evaluation

Program

$\texttt{trycommit}\ (y := x; x := 7)\ \begin{bmatrix} x\ y \\ 7\ 5 \end{bmatrix}\ \begin{bmatrix} x \\ 5 \end{bmatrix}$

$\|$

$\texttt{intrans}\ (x := 1)\ (z := x; x := 1)\ \begin{bmatrix} z \\ 5 \end{bmatrix}\ \begin{bmatrix} x \\ 5 \end{bmatrix}$

$\begin{bmatrix} x\ y\ z \\ 5\ 0\ 0 \end{bmatrix}$

$$\dfrac{\dfrac{\left\langle 1, \begin{bmatrix} z \\ 5 \end{bmatrix} : \begin{bmatrix} x \\ 5 \end{bmatrix} : m \right\rangle \xrightarrow[\text{WA}]{} \left\langle 1, \begin{bmatrix} z \\ 5 \end{bmatrix} : \begin{bmatrix} x \\ 5 \end{bmatrix} : m \right\rangle \quad \begin{bmatrix} x\ z \\ 1\ 5 \end{bmatrix} : \begin{bmatrix} x \\ 5 \end{bmatrix} : m = \text{write}\left(\begin{bmatrix} x \\ 1 \end{bmatrix}, \begin{bmatrix} z \\ 5 \end{bmatrix} : \begin{bmatrix} x \\ 5 \end{bmatrix} : m \right)}{\left\langle x := 1, \begin{bmatrix} z \\ 5 \end{bmatrix} : \begin{bmatrix} x \\ 5 \end{bmatrix} : m \right\rangle \xrightarrow[\text{W}]{} \begin{bmatrix} x\ z \\ 1\ 5 \end{bmatrix} : \begin{bmatrix} x \\ 5 \end{bmatrix} : m}}{\left\langle \texttt{intrans}\ (x := 1)\ s\ \begin{bmatrix} z \\ 5 \end{bmatrix}\ \begin{bmatrix} x \\ 5 \end{bmatrix}, m \right\rangle \xrightarrow[\text{W}]{} \left\langle \texttt{trycommit}\ s\ \begin{bmatrix} x\ z \\ 1\ 5 \end{bmatrix}\ \begin{bmatrix} x \\ 5 \end{bmatrix}, m \right\rangle}$$

Example evaluation

Program

$\texttt{trycommit}\ (y := x; x := 7)\ \begin{bmatrix} x\, y \\ 7\, 5 \end{bmatrix} \begin{bmatrix} x \\ 5 \end{bmatrix}$

$||$

$\texttt{trycommit}\ (z := x; x := 1)\ \begin{bmatrix} x\, z \\ 1\, 5 \end{bmatrix} \begin{bmatrix} x \\ 5 \end{bmatrix}$

$\begin{bmatrix} x\, y\, z \\ 5\, 0\, 0 \end{bmatrix}$

$$\frac{\forall var \in \mathrm{dom} \begin{bmatrix} x \\ 5 \end{bmatrix} \quad \mathrm{read}(var, m_W) = \begin{bmatrix} x \\ 5 \end{bmatrix} var}{\left\langle \texttt{trycommit}\ s\ \begin{bmatrix} x\, y \\ 7\, 5 \end{bmatrix} \begin{bmatrix} x \\ 5 \end{bmatrix}, m \right\rangle \underset{\mathrm{W}}{\longrightarrow} \mathrm{write}\left(wc, \mathrm{rcupdate}(rc, m_W)\right)}$$

Example evaluation

Program

$$\texttt{trycommit}\ (z := x; x := 1)\ \begin{bmatrix} x\, z \\ 1\, 5 \end{bmatrix} \begin{bmatrix} x \\ 5 \end{bmatrix}$$

$$\begin{bmatrix} x\, y\, z \\ \color{red}{7}\, \color{red}{5}\, 0 \end{bmatrix}$$

$$\frac{}{\left\langle \texttt{trycommit}\ s\ \begin{bmatrix} x\, z \\ 1\, 5 \end{bmatrix} \begin{bmatrix} x \\ 5 \end{bmatrix}, m \right\rangle \xrightarrow[\mathrm{W}]{} \langle \texttt{trans}\ s, m \rangle}$$

Example evaluation

Program

$\texttt{trans}\ (z := x; x := 1)$

$\begin{bmatrix} x\ y\ z \\ 7\ 5\ 0 \end{bmatrix}$

$$\frac{}{\langle \texttt{trans}\ s, m \rangle \underset{\mathsf{W}}{\longrightarrow} \left\langle \texttt{intrans}\ s\ s \left[\ \right] \left[\ \right], m \right\rangle}$$

Example evaluation

Program

$\texttt{intrans}\ (z := x; x := 1)\ (z := x; x := 1)\ \big[\ \big]\ \big[\ \big]$

$\begin{bmatrix} x & y & z \\ 7 & 5 & 0 \end{bmatrix}$

$$\dfrac{\dfrac{\dfrac{\left\langle x, [\ ] : [\ ] : m \right\rangle \xrightarrow[\mathrm{WA}]{} \left\langle 7, [\ ] : \begin{bmatrix} x \\ 7 \end{bmatrix} : m \right\rangle \quad \begin{bmatrix} z \\ 7 \end{bmatrix} : \begin{bmatrix} x \\ 7 \end{bmatrix} : m = \mathrm{write}\left(\begin{bmatrix} z \\ 7 \end{bmatrix}, [\ ] : \begin{bmatrix} x \\ 7 \end{bmatrix} : m \right)}{\left\langle z := x, [\ ] : [\ ] : m \right\rangle \xrightarrow[\mathrm{W}]{} \begin{bmatrix} z \\ 7 \end{bmatrix} : \begin{bmatrix} x \\ 7 \end{bmatrix} : m}}{\left\langle z := x; s_2, [\ ] : [\ ] : m \right\rangle \xrightarrow[\mathrm{W}]{} \left\langle s_2, \begin{bmatrix} z \\ 7 \end{bmatrix} : \begin{bmatrix} x \\ 7 \end{bmatrix} : m \right\rangle}}{\left\langle \texttt{intrans}\ z := x; s_2\ s_1\ [\ ]\ [\ ], m \right\rangle \xrightarrow[\mathrm{W}]{} \left\langle \texttt{intrans}\ s_2\ s_1\ \begin{bmatrix} z \\ 7 \end{bmatrix} \begin{bmatrix} x \\ 7 \end{bmatrix}, m \right\rangle}$$

Example evaluation

Program

$\texttt{intrans}\ (x := 1)\ (z := x; x := 1) \begin{bmatrix} z \\ 7 \end{bmatrix} \begin{bmatrix} x \\ 7 \end{bmatrix}$

$\begin{bmatrix} x\ y\ z \\ 7\ 5\ 0 \end{bmatrix}$

$$\frac{\dfrac{\left\langle 1, \begin{bmatrix} z \\ 7 \end{bmatrix} : \begin{bmatrix} x \\ 7 \end{bmatrix} : m \right\rangle \xrightarrow[\text{WA}]{} \left\langle 1, \begin{bmatrix} z \\ 7 \end{bmatrix} : \begin{bmatrix} x \\ 7 \end{bmatrix} : m \right\rangle \quad \begin{bmatrix} x\ z \\ 1\ 7 \end{bmatrix} : \begin{bmatrix} x \\ 7 \end{bmatrix} : m = \text{write}\left(\begin{bmatrix} x \\ 1 \end{bmatrix}, \begin{bmatrix} z \\ 7 \end{bmatrix} : \begin{bmatrix} x \\ 7 \end{bmatrix} : m \right)}{\left\langle x := 1, \begin{bmatrix} z \\ 7 \end{bmatrix} : \begin{bmatrix} x \\ 7 \end{bmatrix} : m \right\rangle \xrightarrow[\text{W}]{} \begin{bmatrix} x\ z \\ 1\ 7 \end{bmatrix} : \begin{bmatrix} x \\ 7 \end{bmatrix} : m}}{\left\langle \texttt{intrans}\ (x := 1)\ s \begin{bmatrix} z \\ 7 \end{bmatrix} \begin{bmatrix} x \\ 7 \end{bmatrix}, m \right\rangle \xrightarrow[\text{W}]{} \left\langle \texttt{trycommit}\ s\ \begin{bmatrix} x\ z \\ 1\ 7 \end{bmatrix} \begin{bmatrix} x \\ 7 \end{bmatrix}, m \right\rangle}$$

Example evaluation

Program

$\texttt{trycommit}\ (z := x; x := 1)\ \begin{bmatrix} x\ z \\ 1\ 7 \end{bmatrix} \begin{bmatrix} x \\ 7 \end{bmatrix}$

$\begin{bmatrix} x\ y\ z \\ 7\ 5\ 0 \end{bmatrix}$

$$\frac{\forall var \in \mathrm{dom} \begin{bmatrix} x \\ 7 \end{bmatrix} \quad \mathrm{read}(var, m_W) = \begin{bmatrix} x \\ 7 \end{bmatrix} var}{\left\langle \texttt{trycommit}\ s\ \begin{bmatrix} x\ z \\ 1\ 7 \end{bmatrix} \begin{bmatrix} x \\ 7 \end{bmatrix}, m \right\rangle \underset{\mathrm{W}}{\longrightarrow} \mathrm{write}\,(wc, \mathrm{rcupdate}(rc, m_W))}$$

Example evaluation

Program

$$\begin{bmatrix} x & y & z \\ 1 & 5 & 7 \end{bmatrix}$$

Database Correctness Criteria

Atomicity A transaction has to execute to completion, or, in case of failure, to appear not to have executed at all. An aborted transaction should have no side effects.

Consistency Transactions transform the database from one consistent state to another consistent state.

Isolation Execution of a transaction must not affect the result of concurrently executing transactions.

Opacity

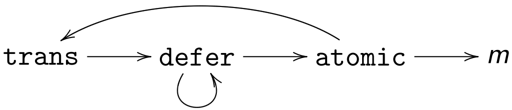
(Guerraoui and Kapałka, 2008)

We have to show that for every run in the weak model there exists some equivalent run, that satisfies the following properties:

1. It is *sequential*, that is, contains no two transactions that are concurrent.
2. It contains no live transactions.
3. It preserves the real-time order of the original run.
4. Every transaction in the equivalent run is *legal*.

“Strong” model

$$
\begin{aligned}
xs_S ::= {} & x := a \mid \texttt{if } b \texttt{ then } s_1 \texttt{ else } s_2 \mid \texttt{while } b \texttt{ do } s \mid \texttt{skip} \\
& \mid xs_S ; s \mid xs_{S1} \parallel xs_{S2} \mid \texttt{trans } s \\
& \mid \texttt{defer } s \mid \texttt{atomic } s \\
cfg_S ::= {} & m \mid \langle xs_S, m \rangle
\end{aligned}
$$



Operational semantics for the "strong" model (excerpt)

$$\frac{\langle a, m\rangle \rightarrow_{SA} v}{\langle x := a, m\rangle \rightarrow_S m[x \mapsto v]} \qquad \frac{}{\langle \texttt{trans}\ s, m\rangle \rightarrow_S \langle \texttt{defer}\ s, m\rangle}$$

$$\frac{}{\langle \texttt{defer}\ s, m\rangle \rightarrow_S \langle \texttt{defer}\ s, m\rangle} \qquad \frac{}{\langle \texttt{defer}\ s, m\rangle \rightarrow_S \langle \texttt{atomic}\ s, m\rangle}$$

$$\frac{}{\langle \texttt{atomic}\ s, m\rangle \rightarrow_S \langle \texttt{trans}\ s, m\rangle} \qquad \frac{\langle s, m\rangle \rightarrow_S^* m'}{\langle \texttt{atomic}\ s, m\rangle \rightarrow_S m'}$$

Relating the weak model to the strong model

$$\frac{\forall x\ \mathrm{read}(x, m_W) = m\ x}{m_W \sim m} \qquad \frac{m_W \sim m}{\langle x := a, m_W \rangle \sim \langle x := a, m \rangle}$$

$$\frac{m_W \sim m}{\langle \mathtt{trans}\ s, m_W \rangle \sim \langle \mathtt{trans}\ s, m \rangle}$$

$$\frac{\langle s, [] : [] : m_W \rangle \to_W^* \langle xs_W, wc : rc : m_W \rangle \quad m_W \sim m}{\langle \mathtt{intrans}\ xs_W\ s\ wc\ rc, m_W \rangle \sim \langle \mathtt{defer}\ s, m \rangle}$$

$$\frac{\langle s, [] : [] : m_W \rangle \to_W^* wc : rc : m_W \quad m_W \sim m}{\langle \mathtt{trycommit}\ s\ wc\ rc, m_W \rangle \sim \langle \mathtt{atomic}\ s, m \rangle}$$

Example

Weak model

$\texttt{trans}\ (y := x; x := 7)$

$\|$

$x := 5; \texttt{trans}\ (z := x; x := 1)$

$\sim$

$$\begin{bmatrix} x & y & z \\ 0 & 0 & 0 \end{bmatrix}$$

Strong model

$\texttt{trans}\ (y := x; x := 7)$

$\|$

$x := 5; \texttt{trans}\ (z := x; x := 1)$

Example

Weak model

$\texttt{intrans}\ (y := x; x := 7)\ (y := x; x := 7) \left[\ \right] \left[\ \right]$

$\|$

$x := 5; \texttt{trans}\ (z := x; x := 1)$

$\sim$

$\begin{bmatrix} x & y & z \\ 0 & 0 & 0 \end{bmatrix}$

Strong model

$\texttt{defer}\ (y := x; x := 7)$

$\|$

$x := 5; \texttt{trans}\ (z := x; x := 1)$

Example

Weak model

intrans $(y := x; x := 7)$ $(y := x; x := 7)$ $\left[\;\right] \left[\;\right]$

$||$

trans $(z := x; x := 1)$

$\sim$

$$\begin{bmatrix} x & y & z \\ 5 & 0 & 0 \end{bmatrix}$$

Strong model

defer $(y := x; x := 7)$

$||$

trans $(z := x; x := 1)$

Example

Weak model

intrans $(x := 7)\ (y := x; x := 7) \begin{bmatrix} y \\ 5 \end{bmatrix} \begin{bmatrix} x \\ 5 \end{bmatrix}$

$||$

trans $(z := x; x := 1)$

$\sim$

$\begin{bmatrix} x\ y\ z \\ 5\ 0\ 0 \end{bmatrix}$

Strong model

defer $(y := x; x := 7)$

$||$

trans $(z := x; x := 1)$

Example

Weak model

$\texttt{trycommit}\ (y := x; x := 7)$ $\begin{bmatrix} x\ y \\ 7\ 5 \end{bmatrix}$ $\begin{bmatrix} x \\ 5 \end{bmatrix}$

$\|$

$\texttt{trans}\ (z := x; x := 1)$

$\sim$

$\begin{bmatrix} x\ y\ z \\ 5\ 0\ 0 \end{bmatrix}$

Strong model

$\texttt{atomic}\ (y := x; x := 7)$

$\|$

$\texttt{trans}\ (z := x; x := 1)$

Example

Weak model

$\texttt{trycommit}\ (y := x; x := 7)\ \begin{bmatrix} x\ y \\ 7\ 5 \end{bmatrix} \begin{bmatrix} x \\ 5 \end{bmatrix}$

$\|$

$\texttt{intrans}\ (z := x; x := 1)\ (z := x; x := 1)\ \begin{bmatrix} \ \end{bmatrix} \begin{bmatrix} \ \end{bmatrix}$

$\sim$

$\begin{bmatrix} x\ y\ z \\ 5\ 0\ 0 \end{bmatrix}$

Strong model

$\texttt{atomic}\ (y := x; x := 7)$

$\|$

$\texttt{defer}\ (z := x; x := 1)$

Example

Weak model

trycommit $(y := x; x := 7)$ $\begin{bmatrix} x\ y \\ 7\ 5 \end{bmatrix}$ $\begin{bmatrix} x \\ 5 \end{bmatrix}$

$||$

intrans $(x := 1)$ $(z := x; x := 1)$ $\begin{bmatrix} z \\ 5 \end{bmatrix}$ $\begin{bmatrix} x \\ 5 \end{bmatrix}$

$\sim$

$\begin{bmatrix} x\ y\ z \\ 5\ 0\ 0 \end{bmatrix}$

Strong model

atomic $(y := x; x := 7)$

$||$

defer $(z := x; x := 1)$

Example

Weak model

trycommit $(y := x; x := 7)$ $\begin{bmatrix} x\ y \\ 7\ 5 \end{bmatrix} \begin{bmatrix} x \\ 5 \end{bmatrix}$

$\|$

trycommit $(z := x; x := 1)$ $\begin{bmatrix} x\ z \\ 1\ 5 \end{bmatrix} \begin{bmatrix} x \\ 5 \end{bmatrix}$

$\sim$

$\begin{bmatrix} x\ y\ z \\ 5\ 0\ 0 \end{bmatrix}$

Strong model

atomic $(y := x; x := 7)$

$\|$

atomic $(z := x; x := 1)$

Example

Weak model

$\texttt{trycommit}\ (z := x; x := 1)\ \begin{bmatrix} x & z \\ 1 & 5 \end{bmatrix} \begin{bmatrix} x \\ 5 \end{bmatrix}$

$\sim$

$\begin{bmatrix} x & y & z \\ 7 & 5 & 0 \end{bmatrix}$

Strong model

$\texttt{atomic}\ (z := x; x := 1)$

Example

Weak model

$\texttt{trans}\ (z := x; x := 1)$

$\sim$

$\begin{bmatrix} x\ y\ z \\ 7\ 5\ 0 \end{bmatrix}$

Strong model

$\texttt{trans}\ (z := x; x := 1)$

Example

Weak model

$\texttt{intrans}\ (z := x; x := 1)\ (z := x; x := 1) \begin{bmatrix} \\ \\ \end{bmatrix} \begin{bmatrix} \\ \\ \end{bmatrix}$

$\sim$

$\begin{bmatrix} x & y & z \\ 7 & 5 & 0 \end{bmatrix}$

Strong model

$\texttt{defer}\ (z := x; x := 1)$

Example

Weak model

$\texttt{intrans}\ (x := 1)\ (z := x; x := 1) \begin{bmatrix} z \\ 7 \end{bmatrix} \begin{bmatrix} x \\ 7 \end{bmatrix}$

$\sim$

$\begin{bmatrix} x\ y\ z \\ 7\ 5\ 0 \end{bmatrix}$

Strong model

$\texttt{defer}\ (z := x; x := 1)$

Example

Weak model

trycommit $(z := x; x := 1)$ $\begin{bmatrix} x & z \\ 1 & 7 \end{bmatrix}$ $\begin{bmatrix} x \\ 7 \end{bmatrix}$

$\sim$

$\begin{bmatrix} x & y & z \\ 7 & 5 & 0 \end{bmatrix}$

Strong model

atomic $(z := x; x := 1)$

Example

Weak model

$$\begin{bmatrix} x\ y\ z \\ 1\ 5\ 7 \end{bmatrix}$$

$\sim$

$$\begin{bmatrix} x\ y\ z \\ \color{red}{1}\ 5\ \color{red}{7} \end{bmatrix}$$

Strong model

$$\begin{bmatrix} x\ y\ z \\ 1\ 5\ 7 \end{bmatrix}$$

Correctness argument

Theorem

- *For all* c_W, c'_W *and* c_S *such that* $c_W \rightarrow_W c'_W$ *and* $c_W \sim c_S$, *there exists* c'_S *such that* $c_S \rightarrow_S c'_S$ *and* $c'_W \sim c'_S$.
- *For all* c_W, c'_W *and* c_S *such that* $c_W \rightarrow^*_W c'_W$ *and* $c_W \sim c_S$, *there exists* c'_S *such that* $c_S \rightarrow^*_S c'_S$ *and* $c'_W \sim c'_S$.

Proof.

By mutual induction on the structures of the derivations of $c_W \rightarrow_W c'_W$ and $c_W \rightarrow^*_S c'_W$ with $c_W \sim c_S$. □

Thank you!