Contents

1 m1 Theory

Built: 02 March 2020
Parent Theories: sm

1.1 Datatypes

```
command = i0 | i1

output = o0 | o1

state = S0 | S1 | S2
```

1.2 Theorems

[command_distinct_clauses]

```
⊢ i0 ≠ i1
```

[m1_rules]

```
⊢ (∀ ins outs.
     TR i0 (CFG (i0::ins) S0 outs) (CFG ins S1 (o0::outs))) ∧
  (∀ ins outs.
     TR i1 (CFG (i1::ins) S0 outs) (CFG ins S2 (o1::outs))) ∧
  (∀ ins outs.
     TR i0 (CFG (i0::ins) S1 outs) (CFG ins S0 (o0::outs))) ∧
  (∀ ins outs.
     TR i1 (CFG (i1::ins) S1 outs) (CFG ins S0 (o0::outs))) ∧
  (∀ ins outs.
     TR i0 (CFG (i0::ins) S2 outs) (CFG ins S2 (o1::outs))) ∧
  ∀ ins outs.
    TR i1 (CFG (i1::ins) S2 outs) (CFG ins S2 (o1::outs))
```

[M1ns_def]

```
⊢ (M1ns S0 i0 = S1) ∧ (M1ns S0 i1 = S2) ∧ (M1ns S1 i0 = S0) ∧
  (M1ns S1 i1 = S0) ∧ (M1ns S2 i0 = S2) ∧ (M1ns S2 i1 = S2)
```

[M1ns_ind]

```
⊢ ∀ P.
    P S0 i0 ∧ P S0 i1 ∧ P S1 i0 ∧ P S1 i1 ∧ P S2 i0 ∧ P S2 i1 ⇒
    ∀ v v1. P v v1
```

[M1out_def]

$\vdash$ (M1out S0 i0 = o0) $\wedge$ (M1out S0 i1 = o1) $\wedge$
(M1out S1 i0 = o0) $\wedge$ (M1out S1 i1 = o0) $\wedge$
(M1out S2 i0 = o1) $\wedge$ (M1out S2 i1 = o1)

[M1out_ind]

$\vdash \forall P.$
P S0 i0 $\wedge$ P S0 i1 $\wedge$ P S1 i0 $\wedge$ P S1 i1 $\wedge$ P S2 i0 $\wedge$ P S2 i1 $\Rightarrow$
$\forall v\ v_1.\ P\ v\ v_1$

[m1TR_clauses]

$\vdash$ ($\forall x\ x1s\ s_1\ out1s\ x2s\ out2s\ s_2.$
TR x (CFG $x1s$ s_1 $out1s$) (CFG $x2s$ s_2 $out2s$) $\iff$
$\exists NS\ Out\ ins.$
($x1s$ = x::ins) $\wedge$ ($x2s$ = ins) $\wedge$ (s_2 = NS s_1 x) $\wedge$
($out2s$ = Out s_1 x::$out1s$)) $\wedge$
$\forall x\ x1s\ s_1\ out1s\ x2s\ out2s.$
TR x (CFG $x1s$ s_1 $out1s$)
(CFG $x2s$ (M1ns s_1 x) (M1out s_1 x::$out2s$)) $\iff$
$\exists ins.$ ($x1s$ = x::ins) $\wedge$ ($x2s$ = ins) $\wedge$ ($out2s$ = $out1s$)

[m1TR_rules]

$\vdash \forall s\ x\ ins\ outs.$
TR x (CFG (x::ins) s $outs$)
(CFG ins (M1ns s x) (M1out s x::$outs$))

[m1Trans_Equiv_TR]

$\vdash$ TR x (CFG (x::ins) s $outs$)
(CFG ins (M1ns s x) (M1out s x::$outs$)) $\iff$
Trans x s (M1ns s x)

[output_distinct_clauses]

$\vdash$ o0 $\neq$ o1

[state_distinct_clauses]

$\vdash$ S0 $\neq$ S1 $\wedge$ S0 $\neq$ S2 $\wedge$ S1 $\neq$ S2

2 sm Theory

Built: 02 March 2020

Parent Theories: indexedLists, patternMatches

2.1 Datatypes

$configuration$ = `CFG ('input list) 'state ('output list)`

2.2 Definitions

[TR_def]

$$
\begin{array}{l}
\vdash \texttt{TR} = \\
\quad (\lambda\, a_0\ a_1\ a_2 . \\
\quad\quad \forall\, TR' . \\
\quad\quad\quad (\forall\, a_0\ a_1\ a_2 . \\
\quad\quad\quad\quad (\exists\, NS\ Out\ s\ ins\ outs . \\
\quad\quad\quad\quad\quad (a_1 = \texttt{CFG}\ (a_0 :: ins)\ s\ outs) \land \\
\quad\quad\quad\quad\quad (a_2 = \texttt{CFG}\ ins\ (NS\ s\ a_0)\ (Out\ s\ a_0 :: outs))) \Rightarrow \\
\quad\quad\quad\quad TR'\ a_0\ a_1\ a_2) \Rightarrow \\
\quad\quad\quad TR'\ a_0\ a_1\ a_2)
\end{array}
$$

[Trans_def]

$$
\begin{array}{l}
\vdash \texttt{Trans} = \\
\quad (\lambda\, a_0\ a_1\ a_2 . \\
\quad\quad \forall\, Trans' . \\
\quad\quad\quad (\forall\, a_0\ a_1\ a_2 .\ (\exists\, NS .\ a_2 = NS\ a_1\ a_0) \Rightarrow Trans'\ a_0\ a_1\ a_2) \Rightarrow \\
\quad\quad\quad Trans'\ a_0\ a_1\ a_2)
\end{array}
$$

2.3 Theorems

[configuration_one_one]

$$
\begin{array}{l}
\vdash \forall\, a_0\ a_1\ a_2\ a_0'\ a_1'\ a_2' . \\
\quad\quad (\texttt{CFG}\ a_0\ a_1\ a_2 = \texttt{CFG}\ a_0'\ a_1'\ a_2') \iff \\
\quad\quad (a_0 = a_0') \land (a_1 = a_1') \land (a_2 = a_2')
\end{array}
$$

[TR_cases]

$$
\begin{array}{l}
\vdash \forall\, a_0\ a_1\ a_2 . \\
\quad\quad \texttt{TR}\ a_0\ a_1\ a_2 \iff \\
\quad\quad \exists\, NS\ Out\ s\ ins\ outs . \\
\quad\quad\quad (a_1 = \texttt{CFG}\ (a_0 :: ins)\ s\ outs) \land \\
\quad\quad\quad (a_2 = \texttt{CFG}\ ins\ (NS\ s\ a_0)\ (Out\ s\ a_0 :: outs))
\end{array}
$$

[TR_clauses]

$$
\begin{array}{l}
\vdash (\forall\, x\ x1s\ s_1\ out1s\ x2s\ out2s\ s_2 . \\
\quad\quad\quad \texttt{TR}\ x\ (\texttt{CFG}\ x1s\ s_1\ out1s)\ (\texttt{CFG}\ x2s\ s_2\ out2s) \iff \\
\quad\quad\quad \exists\, NS\ Out\ ins . \\
\quad\quad\quad\quad (x1s = x :: ins) \land (x2s = ins) \land (s_2 = NS\ s_1\ x) \land
\end{array}
$$

$(out2s = Out\ s_1\ x::out1s)) \wedge$
$\forall NS\ Out\ x\ x1s\ s_1\ out1s\ x2s\ out2s.$
$\mathtt{TR}\ x\ (\mathtt{CFG}\ x1s\ s_1\ out1s)$
$(\mathtt{CFG}\ x2s\ (NS\ s_1\ x)\ (Out\ s_1\ x::out2s)) \iff$
$\exists ins.\ (x1s = x::ins) \wedge (x2s = ins) \wedge (out2s = out1s)$

[TR_complete]

$\vdash \forall s\ x\ ins\ outs.$
$\exists s'\ out.$
$\mathtt{TR}\ x\ (\mathtt{CFG}\ (x::ins)\ s\ outs)\ (\mathtt{CFG}\ ins\ s'\ (out::outs))$

[TR_deterministic]

$\vdash \forall NS\ Out\ x_1\ ins_1\ s_1\ outs_1\ ins_2\ ins_2'\ outs_2\ outs_2'.$
$\mathtt{TR}\ x_1\ (\mathtt{CFG}\ (x_1::ins_1)\ s_1\ outs_1)$
$(\mathtt{CFG}\ ins_2\ (NS\ s_1\ x_1)\ (Out\ s_1\ x_1::outs_2)) \wedge$
$\mathtt{TR}\ x_1\ (\mathtt{CFG}\ (x_1::ins_1)\ s_1\ outs_1)$
$(\mathtt{CFG}\ ins_2'\ (NS\ s_1\ x_1)\ (Out\ s_1\ x_1::outs_2')) \iff$
$(\mathtt{CFG}\ ins_2\ (NS\ s_1\ x_1)\ (Out\ s_1\ x_1::outs_2) =$
$\mathtt{CFG}\ ins_2'\ (NS\ s_1\ x_1)\ (Out\ s_1\ x_1::outs_2')) \wedge$
$\mathtt{TR}\ x_1\ (\mathtt{CFG}\ (x_1::ins_1)\ s_1\ outs_1)$
$(\mathtt{CFG}\ ins_2\ (NS\ s_1\ x_1)\ (Out\ s_1\ x_1::outs_2))$

[TR_ind]

$\vdash \forall TR'.$
$(\forall NS\ Out\ s\ x\ ins\ outs.$
$TR'\ x\ (\mathtt{CFG}\ (x::ins)\ s\ outs)$
$(\mathtt{CFG}\ ins\ (NS\ s\ x)\ (Out\ s\ x::outs))) \Rightarrow$
$\forall a_0\ a_1\ a_2.\ \mathtt{TR}\ a_0\ a_1\ a_2 \Rightarrow TR'\ a_0\ a_1\ a_2$

[TR_rules]

$\vdash \forall NS\ Out\ s\ x\ ins\ outs.$
$\mathtt{TR}\ x\ (\mathtt{CFG}\ (x::ins)\ s\ outs)$
$(\mathtt{CFG}\ ins\ (NS\ s\ x)\ (Out\ s\ x::outs))$

[TR_strongind]

$\vdash \forall TR'.$
$(\forall NS\ Out\ s\ x\ ins\ outs.$
$TR'\ x\ (\mathtt{CFG}\ (x::ins)\ s\ outs)$
$(\mathtt{CFG}\ ins\ (NS\ s\ x)\ (Out\ s\ x::outs))) \Rightarrow$
$\forall a_0\ a_1\ a_2.\ \mathtt{TR}\ a_0\ a_1\ a_2 \Rightarrow TR'\ a_0\ a_1\ a_2$

[TR_Trans_lemma]

$\vdash \mathtt{TR}\ x\ (\mathtt{CFG}\ (x::ins)\ s\ outs)$
$(\mathtt{CFG}\ ins\ (NS\ s\ x)\ (Out\ s\ x::outs)) \Rightarrow$
$\mathtt{Trans}\ x\ s\ (NS\ s\ x)$

[Trans_cases]

$\vdash \forall a_0\ a_1\ a_2.\ \texttt{Trans}\ a_0\ a_1\ a_2 \iff \exists NS.\ a_2 = NS\ a_1\ a_0$

[Trans_Equiv_TR]

$\vdash \texttt{TR}\ x\ (\texttt{CFG}\ (x::ins)\ s\ outs)$
$\quad (\texttt{CFG}\ ins\ (NS\ s\ x)\ (Out\ s\ x::outs)) \iff \texttt{Trans}\ x\ s\ (NS\ s\ x)$

[Trans_ind]

$\vdash \forall Trans'.$
$\quad (\forall NS\ s\ x.\ Trans'\ x\ s\ (NS\ s\ x)) \Rightarrow$
$\quad \forall a_0\ a_1\ a_2.\ \texttt{Trans}\ a_0\ a_1\ a_2 \Rightarrow Trans'\ a_0\ a_1\ a_2$

[Trans_rules]

$\vdash \forall NS\ s\ x.\ \texttt{Trans}\ x\ s\ (NS\ s\ x)$

[Trans_strongind]

$\vdash \forall Trans'.$
$\quad (\forall NS\ s\ x.\ Trans'\ x\ s\ (NS\ s\ x)) \Rightarrow$
$\quad \forall a_0\ a_1\ a_2.\ \texttt{Trans}\ a_0\ a_1\ a_2 \Rightarrow Trans'\ a_0\ a_1\ a_2$

[Trans_TR_lemma]

$\vdash \texttt{Trans}\ x\ s\ (NS\ s\ x) \Rightarrow$
$\texttt{TR}\ x\ (\texttt{CFG}\ (x::ins)\ s\ outs)\ (\texttt{CFG}\ ins\ (NS\ s\ x)\ (Out\ s\ x::outs))$

Index