Reference Card

Jean-Raymond Abrial and Thai Son Hoang

April 2008

Contents

Basic Inference Rules of Mathematical Reasoning

$$\frac{}{H,\ P \vdash P}\ \text{HYP}$$

$$\frac{H \vdash Q}{H,\ P \vdash Q}\ \text{MON}$$

$$\frac{H \vdash P \qquad H, P \vdash Q}{H \vdash Q}\ \text{CUT}$$

Propositional Calculus Rules of Inference (1)

- Rules about conjunction

$$\begin{array}{c} \mathbf{H, P, Q} \vdash \mathbf{R} \\ \hline\hline \mathbf{H,\ P \land Q} \vdash \mathbf{R} \end{array} \quad \textbf{AND\_L}$$

$$\frac{\mathbf{H} \vdash \mathbf{P} \qquad \mathbf{H} \vdash \mathbf{Q}}{\mathbf{H} \vdash \mathbf{P \land Q}} \quad \textbf{AND\_R}$$

- Rules about implication

$$\frac{\mathbf{H, P, Q} \vdash \mathbf{R}}{\mathbf{H,\ P,\ P \Rightarrow Q} \vdash \mathbf{R}} \quad \textbf{IMP\_L}$$

$$\begin{array}{c} \mathbf{H, P} \vdash \mathbf{Q} \\ \hline\hline \mathbf{H} \vdash \mathbf{P \Rightarrow Q} \end{array} \quad \textbf{IMP\_R}$$

Note: Rules with a double horizontal line can be applied in both directions

Propositional Calculus Rules of Inference (2)

- Rules about negation

$$\frac{}{P, \neg P \vdash Q} \quad \text{NOT\_L}$$

$$\frac{}{\bot \vdash P} \quad \text{CNTR}$$

$$\frac{H, P \vdash Q \qquad H, P \vdash \neg Q}{H \vdash \neg P} \quad \text{NOT\_R}$$

$$\frac{H, \neg P \vdash Q \qquad H, \neg P \vdash \neg Q}{H \vdash P} \quad \text{NOT\_R}$$

Propositional Calculus Rules of Inference (3)

- Rules about disjunction

$$\frac{\mathbf{H}, \mathbf{P} \vdash \mathbf{R} \qquad \mathbf{H}, \mathbf{Q} \vdash \mathbf{R}}{\mathbf{H},\ \mathbf{P} \vee \mathbf{Q} \vdash \mathbf{R}} \quad \textbf{OR\_L}$$

$$\frac{\mathbf{H} \vdash \mathbf{P}}{\mathbf{H} \vdash \mathbf{P} \vee \mathbf{Q}} \quad \textbf{OR\_R1} \qquad\qquad \frac{\mathbf{H} \vdash \mathbf{Q}}{\mathbf{H} \vdash \mathbf{P} \vee \mathbf{Q}} \quad \textbf{OR\_R2}$$

- Transforming a disjunctive goal

$$\begin{array}{c} \mathbf{H}, \neg \mathbf{P} \vdash \mathbf{Q} \\ \hline\hline \mathbf{H} \vdash \mathbf{P} \vee \mathbf{Q} \end{array} \quad \textbf{NEG}$$

Predicate Calculus Rules of Inference

$$\frac{H,\ \forall x \cdot P(x),\ P(E)\ \vdash\ Q}{H,\ \forall x \cdot P(x)\ \vdash\ Q}\ \textbf{ALL\_L}$$

$$\frac{H\ \vdash\ P(x)}{H\ \vdash\ \forall x \cdot P(x)}\ \textbf{ALL\_R}$$

$$\frac{H,\ P(x)\ \vdash\ Q}{H,\ \exists x \cdot P(x)\ \vdash\ Q}\ \textbf{XST\_L}$$

$$\frac{H\ \vdash\ P(E)}{H\ \vdash\ \exists x \cdot P(x)}\ \textbf{XST\_R}$$

- In rule **ALL_L** and **XST_R**, **E** is an expression

- In rule **ALL_R** and **XST_L**, variable **x** is not free in **H**.

Equality Rules of Inference

$$\frac{\mathsf{H(F),\ E = F \ \vdash \ P(F)}}{\mathsf{H(E),\ E = F \ \vdash \ P(E)}} \quad \mathbf{EQ\_LR}$$

$$\frac{\mathsf{H(E),\ E = F \ \vdash \ P(E)}}{\mathsf{H(F),\ E = F \ \vdash \ P(F)}} \quad \mathbf{EQ\_RL}$$

$$\frac{}{\vdash \ \mathsf{E = E}} \quad \mathbf{EQL}$$

$$\frac{\mathsf{H \ \vdash \ E \ = \ G \ \wedge \ F \ = \ I}}{\mathsf{H \ \vdash \ E \mapsto F \ = \ G \mapsto I}} \quad \mathbf{PAIR}$$

Basic Set Operator Memberships (Axioms)

These axioms are defined by equivalences.

Left Part	Right Part
$\boldsymbol{E} \mapsto \boldsymbol{F} \in \boldsymbol{S} \times \boldsymbol{T}$	$\boldsymbol{E} \in \boldsymbol{S} \;\wedge\; \boldsymbol{F} \in \boldsymbol{T}$
$\boldsymbol{S} \in \mathbb{P}(\boldsymbol{T})$	$\forall \boldsymbol{x} \cdot \boldsymbol{x} \in \boldsymbol{S} \Rightarrow \boldsymbol{x} \in \boldsymbol{T}$
$\boldsymbol{E} \in \{\boldsymbol{x} \cdot \boldsymbol{x} \in \boldsymbol{S} \;\wedge\; \boldsymbol{P}(\boldsymbol{x}) \mid \boldsymbol{F}(\boldsymbol{x})\}$	$\exists \boldsymbol{x} \cdot \boldsymbol{x} \in \boldsymbol{S} \;\wedge\; \boldsymbol{P}(\boldsymbol{x}) \;\wedge\; \boldsymbol{E} = \boldsymbol{F}(\boldsymbol{x})$
$\boldsymbol{E} \in \{\boldsymbol{x} \mid \boldsymbol{x} \in \boldsymbol{S} \;\wedge\; \boldsymbol{P}(\boldsymbol{x})\}$	$\boldsymbol{E} \in \boldsymbol{S} \;\wedge\; \boldsymbol{P}(\boldsymbol{E})$

Set Inclusion and Extensionality Axiom

Left Part	Right Part
$\boldsymbol{S} \subseteq \boldsymbol{T}$	$\boldsymbol{S} \in \mathbb{P}(\boldsymbol{T})$
$\boldsymbol{S} = \boldsymbol{T}$	$\boldsymbol{S} \subseteq \boldsymbol{T} \quad \wedge \quad \boldsymbol{T} \subseteq \boldsymbol{S}$

The first rule is just a syntactic extension

The second rule is the Extensionality Axiom

Elementary Set Operator Memberships

$\boldsymbol{E \in S \cup T}$	$\boldsymbol{E \in S \quad \vee \quad E \in T}$
$\boldsymbol{E \in S \cap T}$	$\boldsymbol{E \in S \quad \wedge \quad E \in T}$
$\boldsymbol{E \in S \setminus T}$	$\boldsymbol{E \in S \quad \wedge \quad E \notin T}$
$\boldsymbol{E \in \{a, \ldots, b\}}$	$\boldsymbol{E = a \quad \vee \quad \ldots \quad \vee \quad E = b}$
$\boldsymbol{E \in \varnothing}$	$\bot$

Generalizations of Elementary Operator Memberships

$E \;\in\; \mathrm{union}\,(S)$	$\exists s \cdot s \in S \;\land\; E \in s$
$E \;\in\; \bigcup\, x \cdot x \in S \;\land\; P(x) \mid T(x)$	$\exists x \cdot x \in S \;\land\; P(x) \;\land\; E \in T(x)$
$E \;\in\; \mathrm{inter}\,(S)$	$\forall s \cdot s \in S \;\Rightarrow\; E \in s$
$E \;\in\; \bigcap\, x \cdot x \in S \;\land\; P(x) \mid T(x)$	$\forall x \cdot x \in S \;\land\; P(x) \;\Rightarrow\; E \in T(x)$

Well-definedness condition for case 3: $S \neq \varnothing$

Well-definedness condition for case 4: $\exists\, x \cdot x \in S \;\land P(x)$

Binary Relation Operator Memberships (1)

Left Part	Right Part
$r \in S \leftrightarrow T$	$r \subseteq S \times T$
$E \in \mathrm{dom}\,(r)$	$\exists y \cdot E \mapsto y \;\in\; r$
$F \in \mathrm{ran}\,(r)$	$\exists x \cdot x \mapsto F \;\in\; r$
$E \mapsto F \in r^{-1}$	$F \mapsto E \in r$

Binary Relation Operator Memberships (2)

Left Part	Right Part
$r \in S \mathrel{\leftarrow\mkern-14mu\twoheadrightarrow} T$	$r \in S \leftrightarrow T \ \land \ \mathrm{ran}(r) = T$
$r \in S \mathrel{\twoheadleftarrow\mkern-14mu\rightarrow} T$	$r \in S \leftrightarrow T \ \land \ \mathrm{dom}(r) = T$
$r \in S \mathrel{\twoheadleftarrow\mkern-14mu\twoheadrightarrow} T$	$r \in S \mathrel{\leftarrow\mkern-14mu\twoheadrightarrow} T \ \land \ r \in S \mathrel{\twoheadleftarrow\mkern-14mu\rightarrow} T$

Binary Relation Operator Memberships (3)

Left Part	Right Part
$\boldsymbol{E} \mapsto \boldsymbol{F} \;\in\; \boldsymbol{S} \lhd \boldsymbol{r}$	$\boldsymbol{E} \in \boldsymbol{S} \;\wedge\; \boldsymbol{E} \mapsto \boldsymbol{F} \in \boldsymbol{r}$
$\boldsymbol{E} \mapsto \boldsymbol{F} \;\in\; \boldsymbol{r} \rhd \boldsymbol{T}$	$\boldsymbol{E} \mapsto \boldsymbol{F} \in \boldsymbol{r} \;\wedge\; \boldsymbol{F} \in \boldsymbol{T}$
$\boldsymbol{E} \mapsto \boldsymbol{F} \;\in\; \boldsymbol{S} \mathbin{⩤} \boldsymbol{r}$	$\boldsymbol{E} \notin \boldsymbol{S} \;\wedge\; \boldsymbol{E} \mapsto \boldsymbol{F} \in \boldsymbol{r}$
$\boldsymbol{E} \mapsto \boldsymbol{F} \;\in\; \boldsymbol{r} \mathbin{⩥} \boldsymbol{T}$	$\boldsymbol{E} \mapsto \boldsymbol{F} \in \boldsymbol{r} \;\wedge\; \boldsymbol{F} \notin \boldsymbol{T}$

Binary Relation Operator Memberships (4)

$F \in r[w]$	$\exists x \cdot x \in w \quad \land \quad x \mapsto F \in r$
$E \mapsto F \in (p\,;q)$	$\exists x \cdot E \mapsto x \in p \quad \land \quad x \mapsto F \in q$
$p \mathbin{\lhd\mkern-9mu-} q$	$(\mathrm{dom}\,(q) \mathbin{⩤} p) \quad \cup \quad q$
$E \mapsto F \in \mathsf{id}\,(S)$	$E \in S \quad \land \quad F = E$

Binary Relation Operator Memberships (5)

$E \mapsto (F \mapsto G) \in p \otimes q$	$E \mapsto F \in p \quad \wedge \quad E \mapsto G \in q$
$(E \mapsto F) \mapsto G \in \mathsf{prj}_1(S, T)$	$E \in S \quad \wedge \quad F \in T \quad \wedge \quad G = E$
$(E \mapsto F) \mapsto G \in \mathsf{prj}_2(S, T)$	$E \in S \quad \wedge \quad F \in T \quad \wedge \quad G = F$
$(E \mapsto G) \mapsto (F \mapsto H) \in p \parallel q$	$E \mapsto F \in p \quad \wedge \quad G \mapsto H \in q$

Some Useful Modelling Concepts

Given a relation r such that $r \in S \leftrightarrow S$

$r = r^{-1}$	r is symmetric
$r \cap r^{-1} = \varnothing$	r is asymmetric
$r \cap r^{-1} \subseteq \mathrm{id}(S)$	r is antisymmetric
$\mathrm{id}(S) \subseteq r$	r is reflexive
$r \cap \mathrm{id}(S) = \varnothing$	r is irreflexive
$r ; r \subseteq r$	r is transitive

Function Operator Memberships (1)

Left Part	Right Part
$f \in S \mathrel{⇸} T$	$f \in S \leftrightarrow T \quad \wedge \quad (f^{-1} \, ; f) = \mathrm{id}(\mathrm{ran}(f))$
$f \in S \longrightarrow T$	$f \in S \mathrel{⇸} T \quad \wedge \quad S = \mathrm{dom}(f)$
$f \in S \mathrel{⤔} T$	$f \in S \mathrel{⇸} T \quad \wedge \quad f^{-1} \in T \mathrel{⇸} S$
$f \in S \rightarrowtail T$	$f \in S \longrightarrow T \quad \wedge \quad f^{-1} \in T \mathrel{⇸} S$

Function Operator Memberships (2)

Left Part	Right Part
$f \in S ⤀ T$	$f \in S ⇸ T \quad \land \quad T = \mathrm{ran}(f)$
$f \in S \twoheadrightarrow T$	$f \in S \longrightarrow T \quad \land \quad T = \mathrm{ran}(f)$
$f \in S ⤖ T$	$f \in S \rightarrowtail T \quad \land \quad f \in S \twoheadrightarrow T$

Applying a Function

Given a partial function $\boldsymbol{f}$, we have

Left Part	Right Part
$\boldsymbol{F = f(E)}$	$\boldsymbol{E \mapsto F \;\in\; f}$

Well-definedness conditions: $\boldsymbol{f}$ is a partial function

Context Structure

context
$< context\_identifier >$
extends $\star$
$< context\_identifier >$
$\ldots$
sets $\star$
$< set\_identifier >$
$\ldots$
constants $\star$
$< constant\_identifier >$
$\ldots$
axioms $\star$
$< label >$: $< predicate >$
$\ldots$
theorems $\star$
$< label >$: $< predicate >$
$\ldots$
end

- Sections with "$\star$" might be empty

- All keyword sections are predefined in the Rodin Platform

- All labels are generated automatically by the Rodin Platform (but can be modified)

Machine Structure

machine
$< machine\_identifier >$
refines $\star$
$< machine\_identifier >$
sees $\star$
$< context\_identifier >$
$\ldots$
variables
$< variable\_identifier >$
$\ldots$
invariants
$< label >$: $< predicate >$
$\ldots$
theorems $\star$
$< label >$: $< predicate >$
$\ldots$
events
initialisation $\ldots$
$\ldots$
variant $\star$
$< variant >$
end

- Each machine has exactly one **initialisation** event

- All keyword sections are predefined in the Rodin Platform

- All labels are generated automatically by the Rodin Platform (but can be modified)

Event Structure

```
$< event\_identifier > \; \widehat{=}$
  status
    $\{ordinary, convergent, anticipated\}$
  refines $\star$
    $< event\_identifier >$
    $\ldots$
  any $\star$
    $< parameter\_identifier >$
    $\ldots$
  where $\star$
    $< label >$: $< predicate >$
    $\ldots$
  with $\star$
    $< label >$: $< witness >$
    $\ldots$
  then $\star$
    $< label >$: $< action >$
    $\ldots$
  end
```

- Notice that keyword "**where**" becomes "**when**" in the Rodin Platform Pretty Print when there is no "**any**".

- Again, all keyword sections are predefined in the Rodin Platform.

- All labels are generated automatically by the Rodin Platform (but can be modified)

Formal Definition of Invariant Preservation (INV)

evt
  any x **where**
    $G(x, s, c, v)$
  then
    $v :\mid BAP(x, s, c, v, v')$
  end

s	:	seen sets
c	:	seen constants
v	:	variables
$A(s, c)$	:	seen axioms and thms
$I(s, c, v)$	:	invariants and thms.
evt	:	specific event
x	:	event parameters
$G(x, s, c, v)$	:	event guards
$BAP(x, s, c, v, v')$	:	event before-after predicate
$inv(s, c, v')$	:	modified specific invariant

Axioms Invariants Guards of the event Before-after predicate of the event ⊢ Modified Specific Invariant	$evt/inv/$INV

$A(s, c)$
$I(s, c, v)$
$G(x, s, c, v)$
$BAP(x, s, c, v, v')$
$\vdash$
$inv(s, c, v')$

- In case of the initialization event, $I(s, c, v)$ is removed from the hypotheses

Formal Definition of the Feasibility PO (FIS)

```
evt
  any $x$ where
    $G(x,s,c,v)$
  then
    $v :\mid BAP(x,s,c,v,v')$
  end
```

s : seen sets
c : seen constants
v : variables
$A(s,c)$: seen axioms and thms
$I(s,c,v)$: invariants and thms.
evt : specific event
x : event parameters
$G(x,s,c,v)$: event guards
$BAP(x,s,c,v,v')$: event action

Axioms Invariants Guards of the event $\vdash$ $\exists v' \cdot$ Before-after predicate	$evt/act/$FIS

$$
\begin{array}{l}
A(s,c) \\
I(s,c,v) \\
G(x,s,c,v) \\
\vdash \\
\exists v' \cdot BAP(x,s,c,v,v')
\end{array}
$$

Formal Definition of the Guard Strengthening PO (GRD)

```
evt0
  any
    $x$
  where
    $g(x, s, c, v)$
    $\dots$
  then
    $\dots$
  end
```

```
evt
  refines
    evt0
  any
    $y$
  where
    $H(y, s, c, w)$
  with
    $x : W(x, y, s, c, w)$
  then
    $\dots$
  end
```

s : seen sets
c : seen constants
v : abstract variables
w : concrete variables
$A(s, c)$: seen axioms and thms
$I(s, c, v)$: abs. invts. and thms.
$J(s, c, v, w)$: conc. invts. and thms.
evt : specific concrete event
x : abstract event parameter
y : concrete event parameter
$g(x, s, c, v)$: abstract event specific guard
$H(y, s, c, w)$: concrete event guards

Axioms Abstract invariants and thms. Concrete invariants and thms. Concrete event guards witness predicate $\vdash$ Abstract event specific guard	$evt/grd/$GRD

$$\begin{array}{l} A(s, c) \\ I(s, c, v) \\ J(s, c, v, w) \\ H(y, s, c, w) \\ W(x, y, s, c, w) \\ \vdash \\ g(x, s, c, v) \end{array}$$

- It is simplified when there are no parameters

Formal Definition of the Simulation PO (SIM)

```
evt0
  any
    $x$
  where
    $\dots$
  then
    $v :\mid BA1(v, v', \dots)$
  end
```

```
evt
  refines
    evt0
  any
    $y$
  where
    $H(y, s, c, w)$
  with
    $x : W1(x, y, s, c, w)$
    $v' : W2(y, v', s, c, w)$
  then
    $w :\mid BA2(w, w', \dots)$
  end
```

s : seen sets
c : seen constants
v : abstract vrbls
w : concrete vrbls
$A(s, c)$: seen axioms and thms
$I(s, c, v)$: abs. invts. and thms.
$J(s, c, v, w)$: conc. invts. and thms.
evt : concrete event
x : abstract prm
y : concrete prm
$H(y, s, c, w)$: concrete guards
$BA1(v, v')$: abstract action
$BA2(w, w')$: concrete action

Axioms Abstract invariants and thms. Concrete invariants and thms. Concrete event guards witness predicate witness predicate Concrete before-after predicate $\vdash$ Abstract before-after predicate	$evt/act/$SIM

$A(s, c)$
$I(s, c, v)$
$J(s, c, v, w)$
$H(y, s, c, w)$
$W1(x, y, s, c, w)$
$W2(y, v', s, c, w)$
$BA2(w, w', \dots)$
$\vdash$
$BA1(v, v', \dots)$

Formal Definition of the Numeric Variant PO (NAT)

machine
  m
refines
  $\ldots$
sees
  $\ldots$
variables
  v
invariants and thms.
  $I(s, c, v)$
theorems
  $\ldots$
events
  $\ldots$
variant
  $n(s, c, v)$
end

evt
  status
    convergent
  any x **where**
    $G(x, s, c, v)$
  then
    A
  end

s : seen sets
c : seen constants
v : variables
$A(s, c)$: seen axioms and thms
$I(s, c, v)$: abs. invts. and thms.
$J(s, c, v, w)$: conc. invts. and thms.
evt : specific event
x : event parameters
$G(x, s, c, v)$: event guards
$n(s, c, v)$: numeric variant

Axioms Abstract invariants and thms. Concrete invariants and thms. Event guards $\vdash$ a numeric variant is a natural number	evt/NAT

$A(s, c)$
$I(s, c, v)$
$J(s, c, v, w)$
$G(x, s, c, v)$
$\vdash$
$n(s, c, v) \in \mathbb{N}$

Formal Definition of the Set Variant PO (FIN)

machine
m
refines
$\ldots$
sees
$\ldots$
variables
v
invariants and thms.
$J(s,c,v,w)$
theorems
$\ldots$
events
$\ldots$
variant
$t(s,c,v)$
end

s : seen sets
c : seen constants
v : variables
$A(s,c)$: seen axioms and thms
$I(s,c,v)$: abs. invts. and thms.
$J(s,c,v,w)$: conc. invts. and thms.
$t(s,c,v)$: set variant

Axioms Abstract invariants and thms. Concrete invariants and thms. $\vdash$ Finiteness of set variant	FIN

$A(s,c)$
$I(s,c,v)$
$J(s,c,v,w)$
$\vdash$
$\mathrm{finite}(t(s,c,v))$

Formal Definition of Numeric Variant Decreasing PO (VAR)

evt
status
convergent
any x **where**
$G(x,s,c,w)$
then
$v :\mid BAP(x,s,c,w,w')$
end

s : seen sets
c : seen constants
v : variables
$A(s,c)$: seen axioms and thms
$I(s,c,v)$: abs. invts. and thms.
$J(s,c,v,w)$: conc. invts. and thms.
evt : specific event
x : event parameters
$G(x,s,c,v)$: event guards
$BAP(x,s,c,w,w')$: event before-after predicate
$n(s,c,w)$: numeric variant

Axioms Abstract invariants and thms. Concrete invariants and thms. Guards of the event Before-after predicate of the event $\vdash$ Modified variant smaller than variant	evt/VAR

$A(s,c)$
$I(s,c,v)$
$J(s,c,v,w)$
$G(x,s,c,w)$
$BAP(x,s,c,w,w')$
$\vdash$
$n(s,c,w') < n(s,c,w)$

Formal Definition of the Set Variant Decreasing PO (VAR)

```
evt
  status
    convergent
  any  $x$  where
    $G(x, s, c, w)$
  then
    $v :| BAP(x, s, c, w, w')$
  end
```

s : seen sets
c : seen constants
v : variables
$A(s, c)$: seen axioms and thms
$I(s, c, v)$: abs. invts. and thms.
$J(s, c, v, w)$: conc. invts. and thms.
evt : specific event
x : event parameters
$G(x, s, c, v)$: event guards
$BAP(x, s, c, w, w')$: event before-after predicate
$t(s, c, w)$: set variant

Axioms Abstract Invariants Concrete Invariants Guards of the event Before-after predicate of the event $\vdash$ Modified variant strictly included in variant	evt/VAR

$A(s, c)$
$I(s, c, v)$
$J(s, c, v, w)$
$G(x, s, c, v)$
$BAP(x, s, c, w, w')$
$\vdash$
$t(s, c, w') \subset t(s, c, w)$

Formal Definition of the Witness Feasibility PO (WFIS)

```
evt
  refines
    evt0
  any
    $y$
  where
    $H(y, s, c, w)$
  with
    $x : W(x, y, s, c, w)$
  then
    $\ldots$
  end
```

s : seen sets
c : seen constants
v : abstract variables
w : concrete variables
$A(s, c)$: seen axioms and thms
$I(s, c, v)$: abs. invts. and thms.
$J(s, c, v, w)$: conc. invts. and thms.
evt : specific concrete event
x : abstract event parameter
y : concrete event parameter
$H(y, s, c, w)$: concrete event guards
$W(x, y, s, c, w)$: witness predicate

Axioms Abstract invariants and thms. Concrete invariants and thms. Concrete event guards $\vdash$ $\exists x \cdot$ Witness	$evt/x/$WFIS

$$
\begin{array}{l}
A(s, c) \\
I(s, c, v) \\
J(s, c, v, w) \\
H(y, s, c, w) \\
\vdash \\
\exists x \cdot W(x, y, s, c, w)
\end{array}
$$

Formal Definition of the Context Theorem PO (THM)

context
    ctx
extends
    $\ldots$
sets
    s
constants
    c
axioms
    $A(s,c)$
theorems
    $\ldots$
    $thm: \quad P(s,c)$
    $\ldots$
end

s : seen sets
c : seen constants
$A(s,c)$: seen axioms and previous thms
$P(s,c)$: specific theorem

Axioms $\vdash$ Theorem	thm/THM

$$\begin{array}{l} \quad A(s,c) \\ \vdash \\ \quad P(s,c) \end{array}$$

Formal Definition of the Machine Theorem PO (THM)

machine
$m0$
refines
$\ldots$
sees
$\ldots$
variables
v
invariants and thms.
$I(s,c,v)$
theorems
$\ldots$
$thm: \quad P(s,c,v)$
$\ldots$
events
$\ldots$
end

s : seen sets
c : seen constants
v : variables
$A(s,c)$: seen axioms and thms
$I(s,c,v)$: invariants and previous thms.
$P(s,c,v)$: specific theorem

Axioms Invariants ⊢ Theorem	thm/THM

$A(s,c)$
$I(s,c,v)$
$\vdash$
$P(s,c,v)$

Formal Definition of the Well-definedness PO (WD)

- It depends on the potentially ill-defined expression

$\mathrm{inter}\,(S)$	$S \neq \varnothing$
$\bigcap\, x \cdot x \in S \;\wedge\; P(x) \mid T(x)$	$\exists\, x \cdot x \in S \;\wedge P(x)$
$f(E)$	f is a partial function $E \in \mathrm{dom}(f)$
E/F	$F \neq 0$
$E \;\mathrm{mod}\; F$	$F \neq 0$
$\mathrm{card}(S)$	$\mathrm{finite}(S)$
$\min(S)$	$S \subseteq \mathbb{Z}$ $\exists x \cdot x \in \mathbb{Z} \;\wedge\; (\forall n \cdot n \in S \;\Rightarrow\; x \leq n)$
$\max(S)$	$S \subseteq \mathbb{Z}$ $\exists x \cdot x \in \mathbb{Z} \;\wedge\; (\forall n \cdot n \in S \;\Rightarrow\; x \geq n)$

Grd. Strengthening PO when Merging Abs. Events (MRG)

```
evt01
  any
    $x$
  where
    $G1(x, s, c, v)$
  then
    $A$
  end
```

```
evt02
  any
    $x$
  where
    $G2(x, s, c, v)$
  then
    $A$
  end
```

```
evt
  refines
    evt01
    evt02
  any
    $x$
  where
    $H(x, s, c, v)$
  then
    $A$
  end
```

s : seen sets
c : seen constants
v : abstract vrbls
$A(s, c)$: seen axioms and thms
$I(s, c, v)$: abs. invts. and thms.
evt : concrete event
x : similar prm
$H(x, s, c, v)$: concrete guards
$G1(x, s, c, v)$: abstract event guards
$G2(x, s, c, v)$: abstract event guards
A : similar abs. and cnc. actions

Axioms Abstract invariants and thms. Concrete event guards $\vdash$ Disjunction of abstract guards	evt/MRG

$$\begin{array}{l} A(s, c) \\ I(s, c, v) \\ H(x, s, c, v) \\ \vdash \\ G1(x, s, c, v) \lor G2(x, s, c, v) \end{array}$$

ASCII Representations of the Mathematical Symbols (1)

- Atomic Symbols

ASCII	Symbol
`true`	$\top$
`false`	$\bot$
`INT`	$\mathbb{Z}$

ASCII	Symbol
`NAT`	$\mathbb{N}$
`NAT1`	$\mathbb{N}_1$
`BOOL`	$\mathbf{BOOL}$

ASCII	Symbol
`TRUE`	$\mathbf{TRUE}$
`FALSE`	$\mathbf{FALSE}$
`{}`	$\varnothing$

- Assignment Operators

ASCII	Symbol
`:=`	$:=$

ASCII	Symbol
`:|`	$:\mid$

ASCII	Symbol
`::`	$:\in$

ASCII Representations of the Mathematical Symbols (2)

- Unary Operators

ASCII	Symbol
`not`	$\neg$
`finite`	**finite**
`card`	**card**
`POW`	$\mathbb{P}$
`POW1`	$\mathbb{P}_1$

ASCII	Symbol
`union`	**union**
`inter`	**inter**
`dom`	**dom**
`ran`	**ran**
`prj1`	$\mathbf{prj}_1$

ASCII	Symbol
`prj2`	$\mathbf{prj}_2$
`id`	**id**
`min`	**min**
`max`	**max**
`-`	$-$

ASCII Representations of the Mathematical Symbols (3)

- Binary Operators

ASCII	Symbol
`&`	$\wedge$
`or`	$\vee$
`=>`	$\Rightarrow$
`<=>`	$\Leftrightarrow$
`=`	$=$
`/=`	$\neq$
`:`	$\in$
`<<:`	$\subset$

ASCII	Symbol
`/<<:`	$\not\subset$
`<:`	$\subseteq$
`/<:`	$\not\subseteq$
`<`	$<$
`<=`	$\leq$
`>`	$>$
`>=`	$\geq$
`/:`	$\notin$

ASCII	Symbol
`|->` or `,,`	$\mapsto$
`<->`	$\leftrightarrow$
`<<->`	$\twoheadleftarrow\!\!\rightarrow$
`<->>`	$\leftarrow\!\!\twoheadrightarrow$
`<<->>`	$\twoheadleftarrow\!\!\twoheadrightarrow$
`+->`	$⇸$
`-->`	$\rightarrow$
`+->>`	$⤀$

ASCII Representations of the Mathematical Symbols (4)

- Binary Operators (Cont.)

ASCII	Symbol
`-->>`	$\twoheadrightarrow$
`>+>`	$⤔$
`>->`	$\rightarrowtail$
`>->>`	$⤖$
`/\`	$\cap$
`\/`	$\cup$
`\`	$\backslash$
`**`	$\times$

<table>
<tr><th>ASCII</th><th>Symbol</th></tr>
<tr><td><code><+</code></td><td>$\lhd\!\!\!+$</td></tr>
<tr><td><code>||</code></td><td>$\parallel$</td></tr>
<tr><td><code>><</code></td><td>$\otimes$</td></tr>
<tr><td><code>;</code></td><td>$;$</td></tr>
<tr><td><code><|</code></td><td>$\lhd$</td></tr>
<tr><td><code><<|</code></td><td>$⩤$</td></tr>
<tr><td><code>|></code></td><td>$\rhd$</td></tr>
<tr><td><code>|>></code></td><td>$⩥$</td></tr>
</table>

ASCII	Symbol
`*`	$*$
`/`	$\div$
`mod`	$\mathbf{mod}$
`..`	$..$
`^`	$\frown$
`~`	$^{-1}$
`+`	$+$
`-`	$-$

ASCII Representations of the Mathematical Symbols (5)

- Quantifiers

ASCII	Symbol
`!`	$\forall$
`#`	$\exists$
`%`	λ

ASCII	Symbol
`UNION`	$\bigcup$
`INTER`	$\bigcap$

ASCII	Symbol
`|`	$\mid$
`.`	$\cdot$

- Bracketing

ASCII	Symbol
`(`	$($
`)`	$)$

ASCII	Symbol
`[`	$[$
`]`	$]$

ASCII	Symbol
`{`	$\{$
`}`	$\}$