

Proposal for dynamic constraints in Event-B

(The general framework for this proposal is to be found in "Introducing Dynamic Constraints in B, pp. 83-128, LNCS 1393, Springer, 1998". However, the proof obligations differ slightly.)

Following are proposals for a syntax supporting the statement of dynamic constraints in an Event-B component.

Modality "ESTABLISHES"

General form is

```
FOR_ALL
  y
WHERE
  T
ANY_OF
  L
WHEN
  P
ESTABLISHES
  Q
END
```

where

x (used bellow) are the visible variables of the component
y are fresh variables
L is a list of label of events of the component
T is a predicate on the variables y only
P is a predicate on the variables x and y only
Q is a predicate on the variables x , x\$0 and y only

Related proof obligations are:

For each label ℓ of L ,

$$\forall y \cdot (T \wedge P \wedge \Gamma(\ell) \Rightarrow [x\$0 := x][\Sigma(\ell)]Q)$$

with

$\Gamma(\ell)$ the syntactic guard, in the component, of event ℓ
 $\Sigma(\ell)$ the body, in the component, of event ℓ

These POs are to be proved with the same hypotheses as invariant POs.

One may use the keyword ALL_EVENTS in lieu of the list L , with the obvious meaning.

Without fresh variables, the modality is written

```
ANY_OF
  L
WHEN
  P
```

ESTABLISHES
 Q
 END

where

x (used bellow) are the visible variables of the component
 L is a list of label of events of the component
 P is a predicate on the variables x only
 Q is a predicate on the variables x and x\$0 only

and related POs simplify to

$$P \wedge \Gamma(\ell) \Rightarrow [x\$0 := x][\Sigma(\ell)]Q$$

The WHEN part may be suppressed, the POs becoming

$$\forall y \cdot (T \wedge \Gamma(\ell) \Rightarrow [x\$0 := x][\Sigma(\ell)]Q)$$

or

$$\Gamma(\ell) \Rightarrow [x\$0 := x][\Sigma(\ell)]Q$$

Modality "UNTIL"

General form is

FOR_ALL
 Y
 WHERE
 T
 ANY_OF
 L
 MAINTAINS
 P
 UNTIL
 Q
 DECREASING
 V
 END

where

x (used bellow) are the visible variables of the component
 y are fresh variables
 L is a list of label of events of the component
 T is a predicate on the variables y only
 P is a predicate on the variables x and y only
 Q is a predicate on the variables x and y only
 V is an integer expression on the variables x and y only

Related proof obligations are:

$$\forall y \cdot (T \wedge P \wedge \neg Q \Rightarrow V \in \mathbb{N})$$

$$\forall y \cdot (T \wedge P \wedge \bigwedge \ell \cdot \ell \in L \cdot \neg \Gamma(\ell) \Rightarrow Q)$$

and for each label ℓ of L ,

$$\forall y \cdot (T \wedge P \wedge \neg Q \wedge \Gamma(\ell) \Rightarrow [\Sigma(\ell)](\neg Q \Rightarrow P))$$

$$\forall y \cdot (T \wedge P \wedge \neg Q \wedge \Gamma(\ell) \Rightarrow [n := V] [\Sigma(\ell)] (\neg Q \Rightarrow V < n))$$

with

n some fresh variables
 $\Gamma(\ell)$ the syntactic guard, in the component, of event ℓ
 $\Sigma(\ell)$ the body, in the component, of event ℓ

(Note: The second PO is

$$\forall y \cdot (T \wedge P \wedge \neg Q \Rightarrow \forall \ell \cdot \ell \in \mathcal{L} \cdot \Gamma(\ell))$$

rewritten to avoid the quantified disjunction.)

These POs are to be proved with the same hypotheses as invariant POs.

One may use the keyword `ALL_EVENTS` in lieu of the list \mathcal{L} , with the obvious meaning.

Without fresh variables, the modality is written

```

ANY_OF
  L
MAINTAINS
  P
UNTIL
  Q
DECREASING
  V
END

```

where

x (used below) are the visible variables of the component
 \mathcal{L} is a list of label of events of the component
P is a predicate on the variables x only
Q is a predicate on the variables x only
V is an integer expression on the variables x only

and related POs simplify to

$$P \wedge \neg Q \Rightarrow V \in \mathbb{N}$$

$$P \wedge \bigwedge \ell \cdot \ell \in \mathcal{L} \cdot \neg \Gamma(\ell) \Rightarrow Q$$

and for each label ℓ of \mathcal{L} ,

$$P \wedge \neg Q \wedge \Gamma(\ell) \Rightarrow [\Sigma(\ell)] (\neg Q \Rightarrow P)$$

$$P \wedge \neg Q \wedge \Gamma(\ell) \Rightarrow [n := V] [\Sigma(\ell)] (\neg Q \Rightarrow V < n)$$

The `MAINTAINS` part may be suppressed; the keyword `UNTIL` must then be replaced by the keyword `LEADS_TO`. The POs become

$$\forall y \cdot (T \wedge \neg Q \Rightarrow V \in \mathbb{N})$$

$$\forall y \cdot (T \wedge \bigwedge \ell \cdot \ell \in \mathcal{L} \cdot \neg \Gamma(\ell) \Rightarrow Q)$$

$$\forall y \cdot (T \wedge \neg Q \wedge \Gamma(\ell) \Rightarrow [n := V] [\Sigma(\ell)] (\neg Q \Rightarrow V < n))$$

or

$$\neg Q \Rightarrow V \in \mathbb{N}$$

$$\bigwedge \ell \cdot \ell \in \mathcal{L} \cdot \neg \Gamma(\ell) \Rightarrow Q$$

$$\neg Q \wedge \Gamma(\ell) \Rightarrow [n := V][\Sigma(\ell)] (\neg Q \Rightarrow V < n)$$

Refinement

The statement of an UNTIL modality in a component entails, for each of its refinements, one constraint and one new refinement proof obligation.

We take into account the fact that in Event-B, an (abstract) event may be refined by more than one (concrete) event, and also that an (concrete) event may refine more than one (abstract) event.

Let \mathcal{L} be the list of labels of an UNTIL modality declared in a component C , and C' a refinement of C (directly or through other refinements).

Let \mathcal{L}' be the list of labels of the events of C' refining (transitively) at least one event of C whose label appears in \mathcal{L} .

The constraint alluded above is that each event of C' whose label appears in \mathcal{L}' must not refine an event of C whose label does not appear in \mathcal{L} .

Let \mathcal{E}' be the list of labels of the events of C' newly introduced or refining (transitively) events introduced in a previous refinement of C . These events thus refine the virtual skip event of C .

The new PO is:

$$\forall \ell \cdot \ell \in \mathcal{L} \cdot \Gamma(C, \ell) \Rightarrow \forall \ell' \cdot \ell' \in \mathcal{L}' \cup \mathcal{E}' \cdot \Gamma(C', \ell')$$

with

$\Gamma(C, l)$ the syntactic guard, in component C , of event labeled l

This PO is to be proved with the same hypotheses as invariant POs.

If C' is not a direct refinement of C , the PO involves objects that may be separated by many refinement steps. In that case, one may gain more locality of the PO by strengthening it to the following one:

$$\forall \ell'' \cdot \ell'' \in \mathcal{L}'' \cup \mathcal{E}'' \cdot \Gamma(C'', \ell'') \Rightarrow \forall \ell' \cdot \ell' \in \mathcal{L}' \cup \mathcal{E}' \cdot \Gamma(C', \ell')$$

where

C'' is the direct abstraction of C'
 \mathcal{E}'' is the list of labels of the events of C'' newly introduced or refining (transitively) events introduced in a previous refinement of C .
 \mathcal{L}'' is the list of labels of the events of C'' refining (transitively) at least one event of C whose label appears in \mathcal{L} .