Duration Calculus : A real-time semantic for B

Samuel Colin

Samuel.Colin@{inrets.fr, univ-valenciennes.fr}

Directors:

Georges Mariano

Vincent Poirriez

Outlines

- Introduction
- Duration Calculus
- The B method
- Timed B substitutions
- Generating proof obligations
- Conclusion

Real-time with B

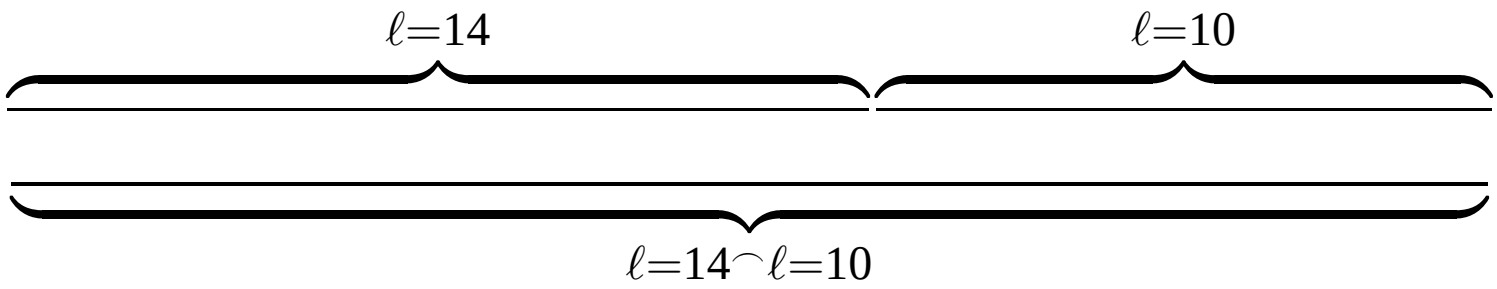
- "Classical" B method still used
- Previous works to add concurrency, real-time properties ([LFD96])
- Numerous works on Duration Calculus and derivatives, showing adequation with programming paradigms

Outlines

- ☑ Introduction
- ❒ Duration Calculus
- ❒ The B method
- ❒ Timed B substitutions
- ❒ Generating proof obligations
- ❒ Conclusion

From Interval Temporal Logic...

- Based on first-order logic
- The "chop" $\frown$ connector
- The special variable ℓ (interval length), real values
- Example :



...to Duration Calculus

- ITL + duration of states $\int$
- States : $0, 1, \vee, \neg$
- Example :

$$
\begin{aligned}
\lceil\!\lceil S \rceil\!\rceil &\equiv \int S = \ell \wedge \ell > 0 \\
\Diamond P &\equiv True^\frown P^\frown True \\
\Box P &\equiv \neg\Diamond\neg P \\
Leak &\equiv Gas \wedge \neg Flame
\end{aligned}
$$

$$\ell > 60 \Rightarrow 20 \int Leak < \ell$$

$$\Box(\lceil\!\lceil Leak \rceil\!\rceil \Rightarrow \ell < 1)$$

$$\Box(\lceil\!\lceil Leak \rceil\!\rceil^\frown \lceil\!\lceil \neg Leak \rceil\!\rceil^\frown \lceil\!\lceil Leak \rceil\!\rceil \Rightarrow \ell \geq 30)$$

DC with iteration

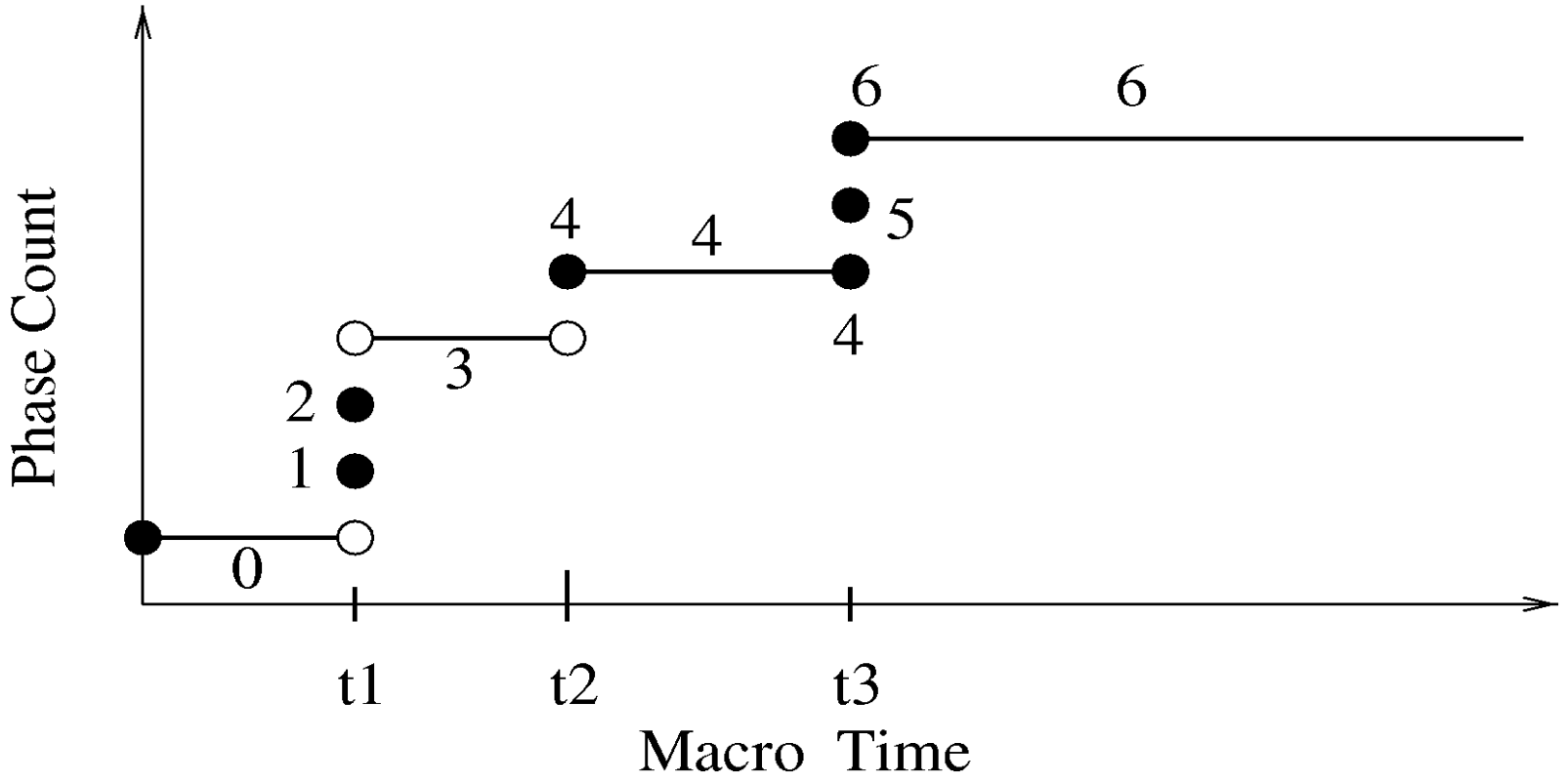
- DC + *repetition* operator *
- Examples (axioms of DC*) :

$$\ell = 0 \Rightarrow \phi^*$$
$$\phi^\frown\phi^* \Rightarrow \phi^*$$

- Ideal to represent the concept of *repetition*
- Note : we can introduce predicates as propositions (not specific to DC*). Example (from [SH01]):

$$\lceil\!\lceil x \neq 0 \wedge \neg \text{CriticalSection} \rceil\!\rceil^*$$

- Weakly monotonic time DC with iteration
- DC* + microtime
- Special variable η, integer value



Projection

- WDC* difficult to handle
- Definition of a projection operator Π
- Example :

$$\begin{aligned} \Pi(\lceil P \rceil^0) &= \ell = 0 \\ \Pi(\lceil P \rceil^0 \frown Unit) &= \lceil\!\lceil P \rceil\!\rceil^* \\ \Pi(D_1 \wedge D_2) &= \Pi(D_1) \wedge \Pi(D_2) \end{aligned}$$

A word about proof support

There exists several implementations of DC in validation tools :

- Based on model-checking ([Pan01])
- PVS [Hei8a]
- Isabelle :
  - Isabelle/DC ([Hei99])
  - Signed Interval Logic in Isabelle ([Ras02])
- Coq ([CPM03])

Outlines

- ☑ Introduction
- ☑ Duration Calculus
- ❒ The B method
- ❒ Timed B substitutions
- ❒ Generating proof obligations
- ❒ Conclusion

The "classical" B method

- Based a lot on Z
- Created by J.-R. Abrial, based on the following paradigms :
  - Set theory
  - Substitutions
  - Object calculus
  - Refinement

Substitutions

A convenient notation to represent state changes :

GSL	$[GSL]P$
$skip$	P
$x := E$	$P[E/x]$
$g \mid S$	$g \wedge [S]P$
$g \Longrightarrow S$	$g \Rightarrow [S]P$
$S;T$	$[S]([T]P)$
$S \llbracket T$	$[S]P \wedge [T]P$
$S \Vert T$	simplified with rewriting rules
...	...

Extensions of B

- Semantic extensions of some substitutions ([BFM00])
- Semantic shift of B machines : concurrency, discrete-time properties ([LFD96])
- Event B (implicit concurrency, *modalities*)

Outlines

- ☑ Introduction
- ☑ Duration Calculus
- ☑ The B method
- ❒ Timed B substitutions
- ❒ Generating proof obligations
- ❒ Conclusion

Extended Hoare triples

Idea first presented in [SH01] :

- Give oneself a simple language
- Express the semantic of this language with WDC*
- Checking pre- and post-conditions of this semantic
- Project DC* formulas from WDC* formulas
- Example :

$$\frac{\{[S]I\}[S, dur([S], I)]\{I\} \quad I \wedge C \Rightarrow [S]I \quad I \wedge \neg C \Rightarrow P}{\{I\} \left[\begin{array}{ll} \text{WHILE } C & \\ \text{DO } S & , dur([S], I)^* \\ \text{INVARIANT } I & \end{array}\right] \{P\}}$$

Timed substitutions

- Resulting timed substitutions :

GSL	$dur([\text{GSL}], P)$
skip	$\lceil\!\lceil \top \rceil\!\rceil$
$x := E$	$\lceil\!\lceil \top \rceil\!\rceil$
delay d	$(\ell = d) \wedge \lceil\!\lceil P \rceil\!\rceil$
$S [\!] T$	$dur([S], P) \vee dur([T], P)$
...	...

- $S || T$ not taken into account (undeterminism, *not* concurrency)
- $[\text{await } C]$: $\lceil\!\lceil \neg C \rceil\!\rceil^*$?

Timed machines

x ⟵Example1 =

TIMING

PRE

$x \geq 1$

THEN

delay 1; x:=x-1;

END

POST

$x \geq 0$

REQUIRES

$\Box(\lceil\!\lceil x \geq 0 \rceil\!\rceil)$

END

Outlines

- ☑ Introduction
- ☑ Duration Calculus
- ☑ The B method
- ☑ Timed B substitutions
- ❒ Generating proof obligations
- ❒ Conclusion

Inclusion, refinement

- Real-time contracts : the **REQUIRES** clause defines a real-time property :

  $dur(Operation, PostCondition) \Rightarrow$ Real-time requirement?

- Refinement :
  - Too early to specify time properties ?
  - Specify a substitution with the same role as **skip** ?
- Development of a UML-based model of the level-crossing problem on the way.

Outlines

- ☑ Introduction
- ☑ Duration Calculus
- ☑ The B method
- ☑ Timed B substitutions
- ☑ Generating proof obligations
- ☐ Conclusion

Conclusion

- WDC* as an adequate semantic for B substitutions
- Difficulty on the side of B :
  - Check the pre- and post-conditions semantic is compatible
  - No simple expression of concurrency
  - B paradigms are somewhat counter-intuitive ($||$, concept of termination,etc)

And after...

- The same proof tool for verifying functional and real-time properties
- Would Event B be more natural ?
- Can previous extensions be expressed with B+DC ?
- B+DC as a semantic for UML ?

References

[BFM00] J.-P. Bodeveix, Mamoun Filali, and C.A. Munoz. Formalisation de la méthode B en COQ et PVS. In *AFADL'2000* [LSR00], pages 96–110.

[CPM03] Samuel Colin, Vincent Poirriez, and Georges Mariano. Thoughts about the implementation of the duration calculus with Coq. In *4th International Workshop on the Implementation of Logics*, volume Technical report ULCS-03-018. University of Liverpool, september 2003. http://www.csc.liv.ac.uk/research/techrepo

[Hei99] Søren T. Heilmann. *Proof Support for Duration Calculus*. Phd-thesis, Department of Information Technology, Technical University of Denmark, Januar 1999.

[Hei8a] Søren T. Heilmann. *PC/DC Users Guide*, 1998(a).

[LFD96] Kevin Lano, J. Fiadeiro, and Jeremy Dick. Extending B AMN with concurrency. Technical

report, Dept. of Computing, Imperial College, 1996.

[LSR00] LSR/IMAG. *Approches Formelles dans l'Assistance au Développement de Logiciels*, LSR/IMAG – BP 72 38402 Saint-Martin d'Heres Cedex – Grenoble – France, January 2000. LSR/IMAG.

[Pan01] Paritosh K. Pandya. Specifying and deciding quantified discrete-time duration calculus formulae using dcvalid. In *RT-TOOLS'2001*, Aalborg, August 2001. (affiliated with CONCUR 2001). Technical report TCS-00-PKP-1, Tata Institute of Fundamental Research, Mumbai, 2000.

[Ras02] Thomas Marthedal Rasmussen. *Interval Logic - Proof Theory and Theorem Proving*. PhD thesis, Informatics and Mathematical Modeling, Technical University of Denmark, january 2002.

[SH01] François Siewe and Dan Van Hung. Deriving real-time programs from duration calculus specifications. In *11th Advanced Research*

Working Conference on Correct Hardware Design and Verification Methods (CHARME 2001), volume LNCS 2144, pages 92–97, Livingston-Edinburgh, Scotland, september 2001. Springer-Verlag. (Technical Report 222, UNU-IIST, P.O. Box 3058, Macau, December 2000).