Characterizing Encapsulation with Bisimulation

Pablo F Castro[1] Tom Maibaum[2]

[1]Departamento de Computación, Universidad Nacional de Rio Cuarto, Argentina

[2]Department of Computing, McMaster University, Canada.

2010

Talk Outline

- The Problem.
- The Logic.
- Components and Bisimulation.
- Some Results.
- An Example.
- Conclusions and further work.

The Problem

How to modularize logical specifications in such a way that local properties of specifications are preserved in arbitrary environments.

Fiadeiro and Maibaum presented an approach for a linear temporal logic:

- Modules are temporal theories.
- Connections between modules are formalized using morphisms between theories.
- Locality (encapsulation) says that only the actions of a module are able to modify the data of this module.
- The composition of modules is achieved using finite colimits.
- We consider an open semantics: components are embedded in a wider environment.

How can we use these ideas in branching temporal logics?

A Temporal Deontic Logic

- A finite set of actions: $a_1, \ldots, a_n$.
- Modal and Deontic Predicates: $\mathsf{P}(\alpha), \mathsf{P_w}(\alpha), \mathsf{F}(\alpha), \langle\alpha\rangle\varphi$.
- Temporal operators: $\mathsf{A}(\varphi\mathcal{U}\psi), \mathsf{AN}\varphi, \mathsf{E}(\varphi\mathcal{U}\psi)$.
- Boolean combinators on actions: U, $\overline{a}$, $a \sqcup b$, $a \sqcap b$.

Semantics: We interpret each action as a set of "events" (or an "event" as a set of actions witnessing that event):

- $I(\alpha) = \{e_1, ..., e_n\}$.
- $I(\alpha \sqcup \beta) = I(\alpha) \sqcup I(\beta)$
- $I(\alpha \sqcap \beta) = I(\alpha) \sqcap I(\beta)$
- $I(\overline{\alpha}) = E - I(\alpha)$

Our models are tuples: $\langle w_0, W, R, E, P, I\rangle$, where w_0 is the initial state, W is a set of states, R is an E-labeled relation between states, P is a relation between events and states that captures the notion of being allowed and I is an interpretation.

Capturing Encapsulation

Encapsulation in linear temporal logic can be captured restricting the possible linear executions:

$V(i)(a) = V(i+1)(a)$, where i is an instant not observed by the component, and a is an attribute of this component, and V is an interpretation or valuation.

We can characterize these kinds of models with the following axiom (say L) given in Fiadeiro and Maibaum's logic:

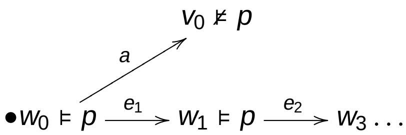
$$(\bigwedge_{g \in \Gamma} \exists x_g : g(x_g)) \vee (\bigwedge_{a \in A} \forall x_a : (\mathrm{X}(a(x_a)) = a(x_a))$$

where A is the set of attributes and Γ is the set of actions.

A morphism between two components (theories) capture the notion of being a component. The preservation of axioms together with the preservation of L ensures the preservation of properties.

Locality in Branching Time

A requirement of the kind used by Fiadeiro and Maibaum is not enough in our category. We want again that external actions to be silent wrt the local state of a component, in the sense that these transitions do not effect the local state. Consider the following model:



where the component has a propositional variable p, an action a and e_i represent executions of external actions. The execution of e_1 preserves the value of p, although after e_1 occurs, it is no longer possible to execute a. That is, we must require that the local non-determinism of an action be preserved.

Locality in Branching Time II

How can we characterize the relation of being-part-of between components?

Semantically:

- We define a relation of bisimulation.
- A standard model is one without external events.
- A locus model is one which is bisimilar to a standard model.

Syntactically:

- Some axiomatic schemes can be used to characterize locus models:
  - $\tau(\varphi) \rightarrow [\overline{\tau(\mathsf{U})}]\tau(\varphi)$, the execution of external actions preserves state properties (τ is a given translation).
  - $\langle\tau(\mathsf{U})\rangle\top \rightarrow \mathsf{AFdone}(\tau(\mathsf{U}))$, fair scheduling: a component cannot diverge beacuse of non-local events when it can execute local actions.
  - $\langle\tau(\gamma)\rangle\top \rightarrow \langle\tau(\gamma) \sqcap \overline{a_1} \sqcap \cdots \sqcap \overline{a_n}\rangle\top$, an indepence requirement: the actions of the component when translated do not depend on actions of the system.
  - $\langle\tau(\gamma)\rangle\tau(\varphi) \rightarrow [\tau(\gamma)]\tau(\varphi)$, the actions external to the component do not add new non-determinism (with respect to the behavior of the local action).

Some Notation...

Given a structure M, we use the following notation:

- $v \xrightarrow{e} v'$, there is an e-labeled transition from state v to state v'.
- $v \stackrel{\epsilon}{\Rightarrow} v'$, the state v' is reacheable from v using a finite number of transitions labeled with external events.
- $w \stackrel{\infty}{\Rightarrow}$, from state w we have an infinite path composed of transitions labeled with external events.
- $L(v)$ is the set of state properties true in that state

Capturing Locality with Bisimulation

Given two structures M_1, M_2 over the same vocabulary and the same events, a relation $Z \subseteq W_1 \times W_2$ between the set of states of M_1 and M_2 is a local bisimulation when:

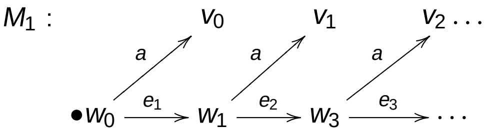
- If wZv, then $L(w) = L(v)$.
- If wZv, and $w \overset{\infty}{\Rightarrow}$, then either $v \overset{\infty}{\Rightarrow}$ or there is a v' such that $v \overset{\epsilon}{\Rightarrow} v'$ and v' has no successors by $\rightarrow$ in M_2.
- if wZv and $w \overset{e}{\rightarrow} w'$. then $w'Zv$ if e is non-local. Otherwise we have some v' such that $v \overset{e}{\rightarrow} v'$ and $w'Zv'$.
- $Z^{\smile}$ also satisfies the above conditions (where $Z^{\smile}$ is the converse of Z).

We have the following property.

Local bisimilar structures are indistinguishable by our logic, i.e., if $M_1 \sim M_2$ then $M_1 \vDash \varphi \Leftrightarrow M_2 \vDash \varphi$, for every formula φ.

An Example

Consider the two following models:



and

$$M_2 : \bullet w_0 \xrightarrow{a} w_1 \xrightarrow{a} w_3 \xrightarrow{a} \cdots$$

these models are not bisimilar since M_1 diverges by external events, and M_2 does not.

Locus Structures

We say that a structure M is a locus iff there is a local bisimulation between M and a standard model M'.

Roughly speaking, locus models are those in which there are occurrence of external events, but they enjoy similar properties to standard models.

Components

A component is a theory presentation $C = \langle V, A \rangle$, where V is a vocabulary and A is a set of axioms.

- A translation between two languages is defined as usual, taking care that the universal action is relative to a component.
- A morphism between components is a translation between languages which preserves axioms and, in addition, $\vdash_{C_2} Loc(\tau)$, where $Loc(\tau)$ are the locality axioms corresponding to the translation τ.

Composing Components

theorem

Given two components $C_1 = \langle V_1, A_1 \rangle$, $C_2 = \langle V_2, A_2 \rangle$, if a translation $\tau : V_1 \rightarrow V_2$ preserves axioms and $\vdash_{C_2} \tau(\varphi)$, for every $\vdash_{C_1} \varphi$.

The collection of components together with morphisms between them constitutes a category **Comp**.

The category **Comp** is finitely cocomplete.

If we have a finite diagram (a design) in **Comp**, its colimit give us the resulting system which preserves properties of its components.

Further Remarks

The principal flaw is that composition of systems may give us inconsistent theories. Some future work:

- Using tableaux we can prove properties, and also prove consistency of a finite set of axioms (finding models). We want to investigate an abstract theory which allows us to put together tableaux systems.
- The composition of fault-tolerant systems is an interesting issue to investigate.
- We need a specification language at a higher level of abstraction to specify systems, which may use this logical system as as *assembler* language.

- P.F.Castro and T.S.E.Maibaum. *Characterizing Locality (Encapsulation) with Bisimulation.* ICTAC 2010.
- D.Harel, D.Kozen and J.Tiuryn. *Dynamic Logic.* MIT Press, 2000.
- J.J.Meyer. *A Different Approach to Deontic Logic: Deontic Logic Viewed as a Variant of Dynamic Logic* Notre Dame Journal of Formal Logic.
- J.Fiadeiro and T.Maibaum *Temporal Theories as Concurrent Units for Concurrent System Specification.* Formal Aspects of Computing, 1992.
- T.Maibaum, S.Khosla. *The Prescription and Description of State-Based Systems.* In B.Banieqbal, H.Barringer and A.Pnueli (eds). Temporal Logic in Computation, LNCS, Springer-Verlag. 1985.
- R.DeNicola, F.Vaandrager. *Three Logics for Branching Bisimulation.* Journal of the ACM (42). 1995.