

# Specification Formalisms for LTSs

Xinxin Liu

Institute of Software  
Chinese Academy of Sciences

BASICS2009

Outline:

1. **Background**
2. Issues in specification formalisms
3. Some specification formalisms
4. HML with single alternation of simultaneous recursive definitions
5. Conclusion

Stepwise refinement:

$S_1 \triangleright S_2 \dots \triangleright S_n$ , and  $P$  implements  $S_n$

Programs: states in a LTS

$\langle \mathcal{P}, Act, \longrightarrow \rangle$

Specifications: describe properties that some programs should satisfy

A specification formalism (for  $\langle \mathcal{P}, Act, \longrightarrow \rangle$ ) consists of

$$\langle \models, \mathcal{S} \rangle$$

where

$\mathcal{S}$ : a set (of specifications)

$\models$ : a binary relation on  $\mathcal{P} \times \mathcal{S}$  (the satisfaction relation)

Read  $P \models S$  as

” $P$  satisfies (is an implementation of)  $S$ ”

$S$  describes a set of states in the LTS, i.e.

$$\{P \in \mathcal{P} \mid P \models S\}$$

Logical specification formalisms: Hennessy-Milner Logic,

$\mu$ -Calculus,...

$\langle \models, \mathcal{L} \rangle$

Transitional specification formalisms:

$\langle \sim, \mathcal{P} \rangle, \langle \approx, \mathcal{P} \rangle, \langle \approx_b, \mathcal{P} \rangle, \dots$

Outline:

1. Background
2. Issues in specification formalisms
3. Some specification formalisms
4. HML with single alternation of simultaneous recursive definitions
5. Conclusion

1. Given  $P$  and  $S$ , does it hold that  $P \models S$ ?  
(**model checking**)
2. Given a property (set of states), can it be expressed as a specification? (**expressiveness**)
3. Are the properties expressible in one formalism always expressible in another? (**relative expressiveness**)

4. Whether a given spec is implementable at all?  
**(consistency check)** If it is, how to construct an implementation? **(model synthesis)**
5. Given  $S_1, S_2$ , does  $S_1 \triangleleft S_2$  hold in the sense that the implementations of  $S_1$  are also implementations of  $S_2$ ?  
**(refinement analysis)**

6. Given a process context  $C$  and a spec  $S$  let

$$sop(C, S) = \{C[P] \mid P \models S\}$$

$$wip(C, S) = \{P \mid C[P] \models S\}$$

are there specs for  $sop(C, S)$  (**compositionality**)

and  $wip(C, S)$  (**decompositionality**)

A refinement step:

$$sop(C, wip(C, S)) \triangleleft S$$

Outline:

1. Background
2. Issues in specification formalisms
3. Some specification formalisms
4. HML with single alternation of simultaneous recursive definitions
5. Conclusion

## $\mu$ -Calculus

$$F ::= \mathbf{tt} \mid \mathbf{ff} \mid X \mid F \wedge F \mid F \vee F \mid [a]F \mid \langle a \rangle F \mid \mu X.F \mid \nu X.F$$

◇ very expressive:

be able to express regular properties

for finite state process  $Q$ , there exists  $F \equiv Q$  such that

$$P \equiv Q \text{ if and only if } P \models F_Q$$

◇ good decompositionality:

$\{P \in \mathcal{P} \mid C[P] \models F\}$  can be expressed for any  $F$  and for  $C$  from a big class of contexts

◇ consistency check and model synthesis:

EXPTIME-complete [Street&Emerson 89, Walukiewicz 95]

◇ refinement analysis:

$$F_1 \triangleleft F_2 \iff \overline{F_1} \vee F_2 \text{ is satisfied by all } P \in \mathcal{P}$$

$$\iff F_1 \wedge \overline{F_2} \text{ is not satisfiable (inconsistent)}$$

reduces refinement analysis to consistency check.

Generalizing transitional specification formalisms —  
 Modal Transition Systems (MTS) [Larsen&Thomsen 88]:

$$\mathcal{M} = \langle \mathcal{S}, Act, \longrightarrow_A, \longrightarrow_R \rangle,$$

where  $\longrightarrow_R \subseteq \longrightarrow_A \subseteq \mathcal{S} \times Act \times \mathcal{S}$ .

Define  $\models$  to be the largest relation on  $\mathcal{P} \times \mathcal{S}$  such that whenever  $P \models S$  then the following holds:

1.  $P \xrightarrow{a} P' \Rightarrow S \xrightarrow{a} S'$  for some  $S'$  with  $P' \models S'$ ;
2.  $S \xrightarrow{a} S' \Rightarrow P \xrightarrow{a} P'$  for some  $P'$  with  $P' \models S'$ .

Example:

Let  $S, U$  be two specifications where

$$S \xrightarrow{a_1} R \quad S, S \xrightarrow{a_1} A \quad S, S \xrightarrow{a_2} A \quad S, \text{ and}$$

for all  $a \in Act, U \xrightarrow{a} A \quad U, U \not\rightarrow R$ .

Let

$$A \stackrel{\text{def}}{=} a_1.A$$

$$B \stackrel{\text{def}}{=} a_1.B + a_2.B$$

then  $A \models S$  and  $B \models S$ .

$P \models U$  for all  $P \in \mathcal{P}$ .

- ◇ more expressive than  $\langle \equiv, \mathcal{P} \rangle$ :
- able to express properties beyond equivalence classes
- ◇ always satisfiable
- ◇ not closed under decomposition, i.e.
  - $wip(C, S)$  is not expressible in general
- ◇ refinement analysis is EXPTIME-complete

[Benes, Kretinsky, Larsen, Srba 08]

## Disjunctive Modal Transition Systems (DMTS)

[Larsen&Liu 90]

- ◇ can express  $\{Q \mid C_1[Q] \sim P_1, \dots, C_n[Q] \sim P_n\}$
- ◇ closed under decomposition: can express  $wip(C, S)$
- ◇ consistency check and model synthesis are EXPTIME
- ◇ refinement analysis is EXPTIME-complete
- ◇ cannot express  $\{Q \mid C_1[Q] \approx P_1, \dots, C_n[Q] \approx P_n\}$

## Extending Disjunctive Modal Transition Systems

◇ can express  $\{Q \mid C_1[Q] \equiv P_1, \dots, C_2[Q] \equiv P_n\}$

where  $\equiv$  can be  $\sim, \approx, \approx_b, \dots$ , or even mixture of

those

- ◇ closed under decomposition: can express  $wip(C, S)$
- ◇ consistency check and model synthesis are EXPTIME
- ◇ refinement analysis — conjecture: EXPTIME

In summary, with the increasing of the expressiveness, the hardness of analysis also increases. Are we able to find a good balance between expressiveness and the ease of analysis?

Outline:

1. Background
2. Issues in specification formalisms
3. Some specification formalisms
4. HML with single alternation of simultaneous recursive definitions
5. Conclusion

Consider the following set of equations

$$X_1 = F_1$$

...

$$X_n = F_n$$

$$Y_1 = E_1$$

...

$$Y_m = E_m$$

where  $F_i$ ,  $E_j$  are HML formulae, and the system is closed in that all the variables on the left hand side are defined.

Then taking the weakest meaning for all  $X_i$  and the strongest meaning for all  $Y_j$  we can use these  $X_i, Y_j$  to express properties expressible in all the modal transition specifications and extensions mentioned above.

- ◇ closed under decomposition: can express  $wip(C, S)$
- ◇ consistency check and model synthesis are EXPTIME
- ◇ refinement analysis — conjecture: EXPTIME