The SynchAADL2Maude Tool Demo

Kyungmin Bae\textsuperscript{1}, Peter Ölveczky\textsuperscript{2}, Abdullah Al-Nayeem\textsuperscript{1}, and José Meseguer\textsuperscript{1}

\textsuperscript{1}University of Illinois at Urbana-Champaign
\textsuperscript{2}University of Oslo
Outline

1. Basic OSATE
2. Invoking SynchAADL2Maude
3. Synchronous AADL Constraints Checker
4. Code Generation and Simulation
5. Model Checking Synchronous AADL Models
Outline

1 Basic OSATE

2 Invoking SynchAADL2Maude

3 Synchronous AADL Constraints Checker

4 Code Generation and Simulation

5 Model Checking Synchronous AADL Models
OSATE is a toolset for AADL given by a set of Eclipse plugins. This is the first screen that you can see when you execute OSATE.
We start with a simple example.

First, we will import the Active Standby example.
The active standby example in our tool webpage can be imported as an existing project.
Main.aadl is a top-level system file that shows a brief architecture.

SynchAADL properties are declared here, to express that this system is in Synchronous AADL.
The AADL graphical model of the active standby example is also given in the file Main.aaxldi
The AADL XML model of the active standby example is automatically generated by OSATE in the file Main.aaxl.
We can create an instance model from a system implementation by pressing the \textit{Instantiate system} button.

The top level system implementation of the active standby system is instantiated here.
Outline

1 Basic OSATE

2 Invoking SynchAADL2Maude

3 Synchronous AADL Constraints Checker

4 Code Generation and Simulation

5 Model Checking Synchronous AADL Models
Invoking the SynchAADL2Maude Window

The SynchAADL2-Maude window can be invoked from an AADL instant model.
From the File menu, we can create an AADL Maude Property Editor file.
We can choose any valid AADL instance model from the wizard.
This screen shows the SynchAADL2Maude window.

There are four buttons in this window: Constraints Check, Code Generation, Perform Simulation, and Perform Verification.
Outline

1. Basic OSATE
2. Invoking SynchAADL2Maude
3. Synchronous AADL Constraints Checker
4. Code Generation and Simulation
5. Model Checking Synchronous AADL Models
Checking SynchAADL Constraints

- We can check SynchAADL constraints by clicking on the Constraints Check button.
What if some SynchAADL constraint is not satisfied?

We add an invalid immediate connection, and see what happened.
Our tool then notifies errors.
Outline

1. Basic OSATE
2. Invoking SynchAADL2Maude
3. Synchronous AADL Constraints Checker
4. Code Generation and Simulation
5. Model Checking Synchronous AADL Models
The Active Standby Example

- Let us go back to the correct model.
We can automatically create the corresponding Real-Time Maude model from a Synchronous AADL model by clicking on the Code Generation button.
We can find the generated Real-Time Maude model on the AADL navigator sidebar.
Maude Development Tool Setting

- When a Maude file is first executed, the MDT setting window is popped-up.

- The correct paths of both a Maude binary file and a Full Maude file should be inserted.

- If “logging to file” is enabled, we should also insert a console log directory.
We can simulate a given model within some bound by pressing the Perform Simulation button.

The result will be shown in the Maude Console.
Outline

1. Basic OSATE
2. Invoking SynchAADL2Maude
3. Synchronous AADL Constraints Checker
4. Code Generation and Simulation
5. Model Checking Synchronous AADL Models
AADL Maude property files are actually XML files.

We can see and modify the content of the file by clicking on the right tab at the bottom.
XML Property File (II)

- The LTL formulas can be defined by `definition` tags.
- The LTL specifications to be verified are defined in `command` tags.
- Let us copy and paste the property definitions from the active standby example in the tool webpage.
The LTL specification to be verified are shown in the AADL Property Requirement table.
When we press the **Perform Verification** button, the LTL properties in the table are model checked in Real-Time Maude.

The model checking result will be shown in the Maude Console.
Here is the model checking result of the active standby example in a larger window.
SynchAADL2-Maude creates the Real-Time Maude verification model from a XML property file.

The verification model can be also found in the AADL Navigator sidebar.
If a given LTL property is not satisfied in a model, then a counterexample is generated.

We illustrate such counterexamples with an incorrect LTL specification for the active standby model.
Here is a generated counterexample in SynchAADL2-Maude.

For each state, a component name and its local variables are displayed.
Thank you!