Conception Formelle : Module Modelisation et Vérification

Travaux dirigés

Alain Griffault

Master 1 Informatique
Université Bordeaux 1
2007-2008

Outline

First manipulation : simulation of models

The AltaRica simulator

- a-simulator is a basic tool to validate AltaRica models.
- You have first to describe your models in a file.
- You can simulate all behaviors with respect to the semantic.

First manipulation : simulation of models

To do

- Electrical circuit (V1, V2 and their corrections).
- Scheduler with and without priority.
- Courses with and without broadcast.

Outline

Second manipulation : validation with a model checker

The AltaRica checker acheck

- acheck is part of the altatools that has been developped for teaching reasons. Its successor arc is a more powerfull checker.
- acheck encodes graphs as graphs, one one hand it is a limit to deal with very big systems, on the oher hand, it permits that all properties can be computed in a linear time in the size of the graph.
- To prove that $M \models P$, you have to compute counter examples for P.
  - `notP := any - P;`
  - `notP := formula-describing-P-counter-examples;`

  and you have to check the result with `test(notP, 0)`.

Second manipulation : validation with a model checker

To do with acheck

You must validate all AltaRica nodes for all examples.

- Compute `deadlock` and `notSCC` properties.
- Check for properties and output results in files.
- For each properties witch is not satisfy, compute a counter example and output it in dot format.
- If the number of configurations is not so big, output in dot format the reachability graph.
- Output in files property's cardinals.

You may also compute properties depending of the system's type.

- Electrical circuit : no loop of reactions.
- Scheduler : the priority between pools of jobs is respected.
- Courses : 3 students can't write at the same time.

Outline

1. First manipulation : simulation of models
2. Second manipulation : validation with a model checker
3. Formal design of a lift
4. Controller synthesis of a tank

Specifications

Informal description

The lift must be use in any building. Its design must no be dependant on the number of floor.

- At each floor, you may call the lift with a button.
- In the lift, there are as many buttons than floors.
- A lighting button means that this request is not yet satisfy.
- When the lift stops, doors open automatically.

At each time, a software controller chooses the next thing to do between : open a door, close a door, go up, go down or nothing.

Specifications

The owner of the building wants that these requirements have been proved.

Requirements

1. When a button is push, it lights.
2. When the corresponding service is done, it lights off.
3. At each floor, the door is close if the lift is not here.
4. Each request must be honored a day.
5. The software opens the door at some floor only if there is some requests for that floor.
6. If there is no request, the lift must stay at the same floor.
7. When the lift moves, it must stop where there is a request.
8. When there are several requests, the software must (if necessary) continue in the same direction than its last move.