Formal Models for Parallel and Distributed Systems Exercise 3 (June 27)

Wolfgang Schreiner
Wolfgang.Schreiner@risc.jku.at

May 30, 2011

The exercise is to be submitted by the deadline stated above via the Moodle interface as a single .zip or .tgz file containing

1. a PDF file with a decent cover page (mentioning the title of the course, your full name and Matrikelnummer) with
   - listings of the model/configuration files and
   - the output of the TLC model checker runs,
2. all .tla and .cfg files used in the specification and model checking.

A Client/Server System: π-calculus

We generalize the client/server system of the previous exercises to a system with S servers and C clients where each server maintains a single instance of a resource. The system schedules the S instances of the resource to the C clients in the following way:

1. Every client iteratively issues a request for a resource. The request contains a fresh client port to which the answer is to be delivered; the client then waits on that port for the grant of the resource.
2. Any server that holds a free resource may accept the request and send a response to the received client port. The response contains a fresh server port by which the resource is to be returned to the server; the server then waits on that port for the return of the resource.
3. After a client has been granted a resource, it uses it for some time and then returns it via the received server port. It then spends some time without the resource and then issues another request.
4. After a server has been given back the resource, it may spend some time without the resource and then accept another request.

First develop a specification in the polyadic π-calculus that describes this system.

Then translate the specification into the language of SPIM. Assign rate 0.01 to all channels of the SPIM model (which models an average message transmission time of 100 time units) and use the operation `delay@`d to model that a client holds a resource for in average d time units, respectively `delay@`e to model that the client spends e time units without the resource.

Annotate the model such that the simulation plots the number of servers that are ready to accept a request, the number of servers that are waiting for the return of the resource, the number of clients that are waiting for a resource, the number of clients that hold a resource, and the number of clients that spend time without the resource.

Run simulations for at least five (significantly different) choices for S, C, d, e and show the corresponding screenshots. Can you interpret/explain the results?