CARLETON UNIVERSITY
Department of Systems and Computer Engineering

SYSC 5104. METHODOLOGIES FOR DISCRETE EVENT MODELLING AND SIMULATION

Term project – Fall 2019

The goal of this document is to present a set of possible subjects to be developed as Term Projects. The detailed specifications for each project will be discussed in detail with each of the groups/individuals.

Important dates:

November 13	Term project choice deadline.
November 26	Work in Progress presentations.
December 22	**Delivery of the Term Projects final Report.**

Deliverables and grades

The details of the deliverables will be discussed in detail for each of the projects. In every case, you should include a 15-20 pages Term Paper explaining the results of your project, showing any results obtained, using IEEE, ACM or SCS format). It should include a brief description of each of the models used (as a conceptual model definition), besides the formal specification. The Term Paper should be organized into:

1. Introduction (brief description of the project, ideas, background and results obtained)
2. Background (any detailed information about background needed for the project)
3. Models defined (including an explanation of the models you built, any rules involved, and general description of the ideas implemented)
4. Simulation results (showing execution of the models in the simulation engine, and analyzing the results obtained. This should include videos of interesting simulation results, if available)
5. Conclusions
6. References

Any source code should run cleanly in the simulation tools, using similar guidelines than the ones used in Assignment 1 and 2.

The term projects will be presented in a “mini-conference” to be held in the course during the last days of classes. The presentation is part of the final mark. There will be a “Best-Paper” award. The best projects will be invited to be extended as Departmental Technical Reports, and might be submitted for publication, if the students are interested. The term projects will be published on-line in the course webpage for perusal of interested modelers.

Formatting instructions for the Term Papers:
http://www.acm.org/sigs/publications/proceedings-templates
http://www.scs.org/PDFs/formattingkit.pdf

(a good example on organization of the Term Paper can be found in:
http://www.sce.carleton.ca/faculty/wainer/papers/SIW-05s-Battle.pdf)

List of proposed projects

I. Definition of advanced Cell-DEVS models

There are different projects in this area:

a) Redefining your Assign2 models using a newer version of CD++ that permits defining multiple state variables in each cell.
b) Models of forest fires
c) Executing ATLAS models
d) Analyzing the execution of Cell-DEVS models based on triangular/hexagonal meshes.
e) Building Occupation models
f) A simple model of hydraulic fracturing. The model allows for the study of the 3D pattern of cracks produced by the increase of fluid pressure in boreholes (a technique used for oil exploration and extraction).

II. Development of application model libraries for Cadmium

The Cadmium and CD++ tool allows modelling and simulating complex systems using the DEVS formalism. It has been used to define many different complex systems. In this case, we intend to build a set of libraries to be ready to use by Engineers or scientists intending to analyse system behaviour through modelling and simulation. The idea of this project is to build mappings between formalisms. The following projects can be faced:

a) Building of a multi-model application in CD++ (integrating Petri Nets, Cell-DEVS models, FSM and DEVS models).
b) Building hybrid models using Modelica/CD++.
c) Building hybrid models using Bond Graphs
d) Building a library of Timed Petri Nets or other similar methods
e) Building a library of Petri Nets in Cadmium
f) Building a library of Finite State Machines in Cadmium

III. Advanced pedestrian models visualization

At present we participate in an effort to model the whole campus in 3D visualization tools (Revit, Dynamo, Rhino, Grasshopper). We have developed numerous pedestrian models as Cell-DEVS. The idea is to interface Cell-DEVS models in CD++ for pedestrian flow, and display the results on a Carleton University campus map.

IV. Visualization of DEVS/Cell–DEVS models

Different visualization techniques have been applied to the current engine. The goal of this project is to test and enhance the current visualization techniques for output models. The focus is on the DEVS web viewer but students can also work with Maya or Revit for 3D visualization projects.

V. Real-Time DEVS models

Cadmium has been modified to include a runtime system that allows defining real-time deadlines. The idea of this project is to create a sample application in which there will be Real-Time models built as DEVS components and interfaces microcontrollers for robotics and other real-time applications.

VI. From DEVS coupled model specification in XML to Cadmium code

Implementing models formally defined in DEVS may is sometimes complex to understand, in particular for domain experts that are interested in modeling but not in programming. The objective of this project is to automate the implementation of DEVS coupled models using a graphical interface. You will need to define a library that uses a graphical version of DEVS coupled models and generates an XML file containing models, ports and link definition. An extension (or a different project) would take as input that XML file containing the definition of a DEVS coupled model, and would generate a template to implement the coupled model in Cadmium. The assumption for this project is that all the atomic models used by the modeler are already implemented in Cadmium.

VII. Cell-DEVS model for measuring the effect of the presence of an occupant on CO2 levels.

The objective is to study the flow of CO2 inside a closed room that has a CO2 sensor installed for the purpose of:
(1) measuring the increase in CO2 level, as detected by the sensor, due to the presence of an occupant in the room and
(2) measuring the latency between the arrival of an occupant and the detection of the increase in CO2

VIII. Analyzing occupancy behaviour based on lux levels in buildings

The goal is to use Cell-DEVS models and the CD++ tool to simulate different occupancy behavior of electric light & blind states based on lux levels in the buildings. The occupants have different preferences for lighting levels: bright, medium, and low. The interactive behavior of controls is three states: frequent, occasionally, and rarely. The orientation of the occupants towards the lighting their comfort behavior (glare). Based on the occupancy inputs and lux levels, the model needs to return the possible states of electric light and blinds states and project energy usage.