CARLETON UNIVERSITY
Department of Systems and Computer Engineering

SYSC 5104. METHODOLOGIES FOR DISCRETE EVENT MODELLING AND SIMULATION

Term project – Details

The goal of this document is to give you details about the development in the projects you have chosen and give you guidance about the steps to be followed in order to meet the requirements.

Deliverables (everybody)

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; it should be organized as 1. Motivation (WHY are you doing this?); 2. Goals (WHAT are you doing it?); 3. Contributions (HOW did you do it?))
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)
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. In order to ensure proper execution, try to execute the same steps explained in the assignments by yourselves, before delivering the software.

DEADLINE: April 26, 2025 (earlier submissions are accepted).

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)

The authors of the best projects will be invited to be extended and submitted for publication, if the students are interested. The final projects will be published on-line in the course webpage for perusal of interested modelers.

The source code should run cleanly. Do not include the final executable file. Zip all the source code (header files, C++ code, .ma files, .ev files, .pal files). Include shell scripts to run the model, explaining how to run them, which input files are needed for each case. Projects not running in the first attempt **will not be graded, and the marks will be lost**. The zip file should include the document of the final report. Follow the guidelines for Assignment 1 and 2 for submitting. Use meaningful names everywhere ("FinalProject" or

“MyName” are examples of names you should not be using; instead, names like “WirelessNetworkSimulator.zip” are better).

The reports will be analyzed as follows:
. The files in the zip file submitted will be recompiled using your files, generating the simulator.
. The examples will be run using the scripts you provided.
. After these basic steps are carried out, different changes will be studied, in order to find possible errors.

In order to ensure proper execution, try to execute these same steps by yourselves, before delivering the software.

Include 'read.me' files explaining the detailed behavior for each one of the scripts provided. Include the electronic version of your final reports in the zip file.

ALL THESE MATERIALS MUST BE PROVIDED. In each of the projects, I include particular information to be included ('Detailed deliverables'), which DOES NOT exclude presenting the materials discussed here.

Below you will find generic descriptions of the topics; in most cases students have coordinated the details with Prof. Wainer: if there are missing details, please consult.

Details of the proposed projects

I. Building Advanced Cell-DEVS models

We used Cadmium to develop different models in class. The goal of this project is to define advanced examples of complex physical systems using Cell-DEVS and Cadmium v2. The initial version of the project should be an extended version of Assignment 2. The definitive version should be an improved version, including phenomena not originally included in the papers you used to define the models.

i) Implement your own examples
ii) Enhance the model described in Assign 2 with new rules, extending the original ones
iii) Read the documentation Asynchronous Cell-DEVS in the article and video uploaded to Brightspace
iv) Include videos of visualizations

Detailed deliverables:
Submission: similar to Assignment 1 and 2 (follow the To-do List)

- Definition of different examples with different input values.
- Execution results of these examples
- Final Report

Information:

. Occupancy Models (De Santis)

- Use the Elephant House and Building Evacuation models as a starting point (CD++ models)
- Check the articles uploaded
- Check the information below
- Build your own models

https://www.sce.carleton.ca/faculty/wainer/wbgraf/doku.php?id=model_samples:start
https://github.com/SimulationEverywhere/co2_models
https://github.com/HazelGriffith/CellDevs_RoomEvac
https://github.com/HazelGriffith/CellDevs_RoomEvac

. Dengue models

- Use the Dengue models below as a starting point (CD++ models)
- Check the articles uploaded
- Check the information below
- Build your own models

https://github.com/SimulationEverywhere-Models/CDPP-DengueModels

II. Definition of advanced Ecological models (Forest fires, plant vegetation) using Cell-DEVS

The goal of this project is to define advanced models of fire spreading and compare the results of different techniques for this model.. The goal here is to show the applicability of the approach when used in timed Cell-DEVS. The students will execute diverse models, and interface them with a GIS tool.

i) Run the original model using the GIS software
ii) Read the documentation Asynchronous Cell-DEVS in the article uploaded to Brightspace
iii) Implement the model using Asynchronous Cell-DEVS and repeat the exercise

Detailed deliverables:
Submission: similar to Assignment 1 and 2 (follow the To-do List)
Definition of different examples with different input values.
Execution results of these examples
Final Report

III. 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.

Details will be discussed with Prof. Wainer. The idea is to build_a number of sample models like the ones defined in Assignment 1 but using E-Cadmium.

Detailed deliverables:
Submission: similar to Assignment 1 (follow the To-do List; you must include videos of any experiments done with the robots – a digital camera can be provided if needed)

- Definition of different examples with different input values.
- Execution results of these examples
- Final Report