Projects offered by Prof. Gabriel Wainer

https://www.sce.carleton.ca/faculty/wainer/

Further information:

https://arslab.sce.carleton.ca

Project Title	Description
A platform for development and testing of simulation and real-time applications	This project focuses on advancing the development of the Cadmium simulation framework, and the Real-Time Cadmium tool across multiple platforms. Cadmium, implemented in C++, is used for discrete-event modeling and simulation, as well as developing real-time control applications in microcontrollers. The objective of this project is to integrate the existing Cadmium components into a unified, user-friendly development environment that supports modeling, execution, and analysis of simulations. The final outcome will be a functional, well-documented Integrated Development Environment (IDE) for simulation development and execution, using web-based interfaces, The resulting platform will be generic and modular, ensuring that it is not tied to any specific modeling tool or application domain. Instead, it will serve as a flexible, extensible environment to support ongoing and future simulation research and development.
Development of cyber-physical systems using a model-based approach	Cyber-physical systems (CPS) integrate computational and physical components that interact closely with their surrounding environment. These systems are fundamental to a wide range of engineering domains, including robotics, transportation, manufacturing, industrial automation, and safety-critical applications. CPS must satisfy real-time constraints, as failures to meet these timing constraints can lead to incorrect behavior, equipment damage, or, in extreme cases, risks to human life. The design and implementation of distributed cyber-physical controllers remain technically challenging due to the complexity of coordinating computation, communication, and physical processes. These challenges often result in lengthy development cycles and high associated costs. To address these issues, we use a systematic, model-based methodology supported by a suite of tools that reduce development complexity, and time-to-deployment. The methodology is based on a simulation-driven workflow in which system components are

	initially modeled and tested within a simulated environment, with simulated elements being progressively replaced by their physical counterparts as development advances. In this capstone project, students will gain hands-on experience with this model-based methodology for the development of embedded cyber-physical applications. The team will select a target application domain aligned with their background and interests. Students will design and implement a complete application from the ground up, beginning with system modeling and simulation and ending in a fully operational implementation using real hardware components. By the end of the project, students will have acquired practical experience in CPS modeling, real-time embedded control, hardware–software integration, and systematic system development—skills that are directly applicable to modern engineering practice.
Modelling Aircraft Collisions	This project is in collaboration with the National Research Council's Aerospace Research Centre, and it focuses on improving an existing Matlab model and converting it into Python code. This software is designed to run Monte Carlo simulations to model collisions between two aircrafts, with one aircraft executing collision avoidance maneuvers based on the implemented avoidance algorithm. The ideal candidate will need proficiency in Python to develop the new software and familiarity with Matlab to understand the existing code. Additionally, the current software is not optimized and part of the project will involve improving the efficiency and user-friendliness. The software consists of several modules: simulation of drone/manned aircraft flight path, transponder and camera models, Kalman filter, conflict prediction and avoidance. Upon completion, the student's software will be posted on the NRC GitHub, with the student recognized as a contributor and co-author on the NRC-maintained software repository. This project is co-supervised by Dr. Iryna Borshchova at NRC (Adjunct Professor, SCE).

Tools for development of Discrete event controllers: the case of a helicopter supervisory system	The AASCEND project at the National Research Council's Flight Research Laboratory is preparing for Canada's first autonomous helicopter flight. A key component of the autonomy stack is a supervisory controller that manages mission progress—for example, initialization, takeoff, navigation, landing, and emergency handling. Because these mission phases are best represented as discrete transitions between well-defined states, a controller is built on a discrete-event system (DES) model. This capstone project focuses on improving and enhancing an object-oriented C++ library and IDE that can model discrete-event systems cleanly and efficiently. The goal is to create a flexible foundation that allows flight engineers to define events, states, transitions, guards, and actions in a structured manner. Project Goals 1. Improve and enhance an existing library that supports states and state hierarchy, events and event handling, transition rules and guards, actions associated with transitions or states, logging and debugging interfaces to help validate controller behavior 2. Expand the existing Supervisory Controller: the mission-level state machine will be enhanced, including new safety and fallback states Required Technical Knowledge Object-oriented C++ programming Software testing methodologies API-based communication
YATM - Yet another Travel Map	YATM (Yet Another Travel Map) is a web-based platform designed to help users document, visualize, and manage their travel history in an interactive and data-driven way. While numerous travel-logging applications currently exist, many lack flexibility in data management, interoperability, and integration with modern geospatial tools. This project aims to design and implement a full-stack web application that addresses these limitations through a robust architecture and rich feature set. The system will include: • A centralized database for securely storing user accounts, travel records, geospatial coordinates, routes, and metadata. • Integration with Geographic Information Systems (GIS) to support map-based visualization and spatial data operations.

	• A comprehensive user-management system, including authentication, role-based permissions, and personalized data views. • Import and export capabilities supporting widely used geospatial formats such as KML, JSON, and XML, enabling smooth interoperability with external mapping applications and devices. • An intuitive and interactive user interface, allowing users to input, edit, and visualize travel information directly through GIS-based tools. • Compatibility with existing travel-mapping examples, such as the reference available at https://bit.ly/3yFBfBd, while extending functionality and offering enhanced customization and scalability.