Interfacing EMTP-RV with other software using C++

Jean-Gabriel Roumy
April 30th, 2009

Hydro Québec

Overview

- **EMTP-RV DLL interface**
- **Interface implementation using C++**
- **Usage**
  - Simulink
  - 3rd party software
  - Distributed Simulation
- **Future work**

EMTP DLL Interface

> What is the DLL Interface ?

- DLL is an external, separate piece of code linked to EMTP-RV at run-time
- EMTP-RV offers an API allowing the implementation of user models
- API includes all functions necessary to interact with computational engine
- API functions and structures are written in FORTRAN 95

DLL Interface in C++

- **Why C++ ?**
  - Interface with 3rd-party software
- **C++ interface maps EMTP memory space**
- **C++ skeleton that handles all calls to/from EMTP**
- **User implements interface functions**

Interface Usage

- Linking with control code
- Controllable and observable signals are inputs/outputs to 3rd-party control libraries
- No participation from user model in system matrices

Using Simulink design in EMTP

- **Protection relay algorithm developped at IREQ using Simulink**
- **Impossible to duplicate design in EMTP**
- **Production code and model code stem from single simulink design**

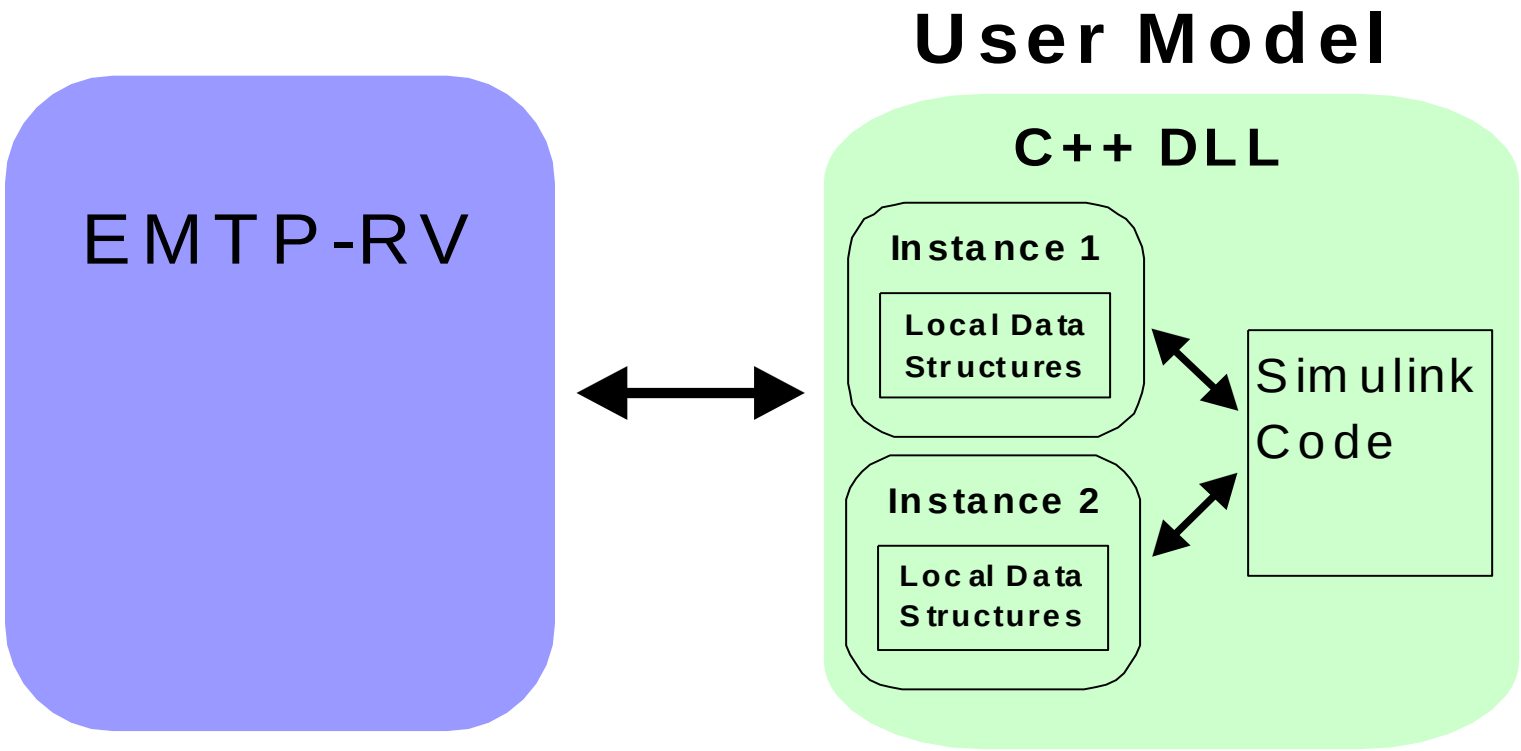
Using Simulink design in EMTP

- **Use existing Simulink designs**
- **Availability of Simulink extensive list of toolboxes**
- **Complex Simulink models**
  - Multiple sample times
  - C-language S-functions
- **Simulink code is portable, compact, fast and efficient**

Using Simulink design in EMTP

- **C++ code generated with Embedded Real-Time toolbox**
- **User defines inputs, outputs, tunable parameters**
- **Compiled design has 3 top-level functions**
  - Initialise
  - Step
  - Terminate

Simulink DLL Interface Architecture



Generating Simulink DLL Interface

- **Integrate Simulink code to interface code and compile together**
- **Use observable and controllable signals as input/output pins in EMTP**
- **Generate symbol in EMTP**

Limitations

- If Simulink design is modified, must regenerate and compile
- Each design requires customizing interface code
- Matlab variable initialisation cannot be compiled
- Fixed time step

Linking with 3rd party software

- Coupling with other C++ models
- 3rd party can provide black-box models generated with their own software
- Example : HVDC controller model generated by manufacturer, provided to HQ as black-box model

Distributed Simulation

- High Level Architecture is IEEE standard
- Federation controlled by RTI
- Distributed simulation can be performed with any HLA compatible software
- EMTP DLL interface acts as bridge between EMTP and RTI

Distributed Simulation

- NS2 (telecom simulator) modified to support HLA
- Co-simulation run with NS2 and EMTP
- Drawback : high overhead, not necessary to share at each timestep
- Early steps of development

Future Work

- Get the process of generating a model more user-friendly and generic
- Co-simulation with Simulink
- Development of HLA co-simulation

The end

> **Questions ?**