

Conversión de modelos PowerDEVS al lenguaje Modelica

Tesinista: Luciano Andrade

Director: Federico Bergero, Co-Director: Ernesto Kofman

19 de enero de 2015

1. Resumen

En este trabajo se describe la implementación de una aplicación para convertir modelos descritos en la herramienta PowerDEVS a modelos en el lenguaje Modelica, más específicamente en μ Modelica, con el fin de aprovechar la velocidad de simulación del 'QSS Solver', permitiendo describir las simulaciones en el entorno PowerDEVS y ejecutando las simulaciones en 'QSS Solver'

2. Introducción

2.1. Motivación y Objetivos

PowerDEVS es una herramienta de simulación de sistemas híbridos, basado en el formalismo DEVS, con una interfaz grafica orientada a bloques, donde los bloques pueden ser conectados entre si, modificado los parametros, posibilidad de conectarse con el entorno Scilab para poder utilizar expresiones y herramientas de cálculo provistas por este entorno.

La resolución de ecuaciones diferenciales ordinarias, requiere el uso de métodos de integración numérica. Todos los algoritmos tradicionales de integración se basan en la discretización de la variable independiente (que generalmente representa el tiempo). Las rutinas que implementan estos algoritmos, se denominan solvers y existen gran variedad de implementaciones de los mismos en diferentes lenguajes de programación. Los Métodos de Integración Numérica QSS (Quantized State System), a diferencia de los métodos de integración tradicionales, realizan la discretización sobre las variables de estado. En consecuencia, convierten los sistemas continuos en sistemas de eventos discretos, y tienen grandes ventajas para simular sistemas con discontinuidades. Si bien PowerDEVS, implementa la totalidad de los algoritmos de QSS, resultan ineficientes, dado que malgastan gran parte de la carga computacional en la transmisión de eventos entre submodelos.

Para solventar este hecho se desarrollo una familia de QSS stand-solver, el cual requiere un modelo descripto en lenguaje C el cual contiene las ecuaciones difereciales, las funciones de cruce de cero asi como la información estructural requerida por los algoritmos QSS. Estos solvers obtienen una mejora de performace de hasta un orden de maginitud comparado con otras implementaciones DEVS. Sobre este se desarrollo una herramineta la cual genera a partir de un modelo μ -Modelica (un subconjunto del lenguaje Modelica) el modelo requerido para el QSS solver.

Con el objetivo de utilizar los mejoras de velocidad y mantener un entorno amigable con el usuario, se creo una herramienta capas de convertir un modelo PowerDEVS en un modelo μ -Modelica.

2.2. Trabajo relacionado

¿Cuales?

2.3. Alcance

DEVS, Discrete Event System Specification (Especificación de Sistemas de Eventos Discretos), es un formalismo modular y jerarquico para modelar y analizar sistemas que pueden ser de eventos de tiempo discreto mediante tablas de transición, y con estados continuos mediante que pueden ser descriptos mediante ecuaciones diferenciales.

En el formalismo clasico DEVS, los modelos atómicos capturan el comportamiento del sistema, mientras los modelos Acoplados describen la estructura del sistema.

En particular los modelos atómicos en PowerDEVS son descriptos en clases C++, mientras que la estructura se encuentra definida en archivos pds y pdm.

Modelica es un lenguaje de modelado, orientado a objetos, declarativo, para el modelado orientado a componentes de sistemas complejos.

Para poder realizar nuestro objetivo es necesario primero contar con un modelo en modelica para cada atómico PowerDEVS que deseemos convertir.

De esto se desprenden las siguientes limitaciones importantes:

- La semantica de los modelos convertidos depende de los modelos equivalentes a los DEVS atómicos
- Solamente podemos convertir modelos cuyos componentes atómicos esten o puedan ser convertidos a modelica.

3. Conceptos Previos

3.1. Modelado y Simulación

3.1.1. Sistemas Continuos y Discretos

¿DONDE?

3.1.2. Métodos de Integración numérica

Un sistema continuo puede ser descripto por un modelo en espacios de estados de la forma:

$$x(t) = f(x(t), u(t)) \quad (1)$$

3.2. Formalismo DEVS

Ber12 pag. 20

3.2.1. Atómicos

3.2.2. Acoplados

3.2.3. Modelos Vectoriales

BK11

3.3. Métodos de integración QSS

QSS paper

3.4. PowerDEVS

Powerdevs Paper

3.5. Modelica

3.6. QSS Stand Alone Solver

QSSPaper

3.6.1. μ Modelica

4. Conversión de modelos DEVS

4.1. Modelos Atómicos

Cómo se traducen (es conocimiento del modelador, no automático)

4.2. Modelos Vectoriales

Consideraciones y anotaciones en modelica, modificaciones

4.3. Modelos Acoplados Planos

Modelos acoplados solo con modelos atómicos adentro.

Mencionar el algoritmo (traducción de conexiones y “aplanado” de cada uno de los atómicos hijos“)

4.4. Equivalencia semántica de la conversión

5. Modelos Acoplados Jerárquicos

Explicar cuándo se utilizan y que resolvemos el problema aplanando los acoplados

5.1. Algoritmo de aplanado

Describir el algoritmo para PDS

5.2. Comparación de performance

¿No debería estar después de .Ejemplos de Aplicación?

6. Detalles de Implementación

API Powerdevs, AST Modelica Traverser Modelica Transformer

7. Ejemplos de Aplicación

tamaños de los vectores y comparativas de

7.1. Vector/airs

7.2. Vector/lcline

8. Conclusiones y Trabajo a futuro

Referencias