ARTICLE IN PRESS

Robotics and Autonomous Systems ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

Unified meta-modeling framework using bond graph grammars for conceptual modeling

Sampath Kumar Veera Ragavan [a,*], Madhavan Shanmugavel [a], Velappa Ganapathy [b], Bijan Shirinzadeh [c]

[a] School of Engineering, Monash University Malaysia Campus, Malaysia
[b] Faculty of Engineering & Technology, SRM University, India
[c] Monash University, Clayton Campus, Australia

HIGHLIGHTS

- We analyze limitations of MDA/UML based frameworks for designing Cyber Physical Systems.
- We propose BG-UMF as a novel, practical and viable alternative based on Bond Graph Grammars.
- Design Automation (Concept to Code) is achieved by propagating constraints and causality.
- Output is an executable model, transformable to platform independent code using available tools.
- Validation is done using an example—concept design and development of a rotorcraft.

ARTICLE INFO

Article history:
Received 15 August 2013
Received in revised form
20 April 2015
Accepted 4 May 2015
Available online xxxx

Keywords:
Mechatronic system design
Design synthesis
Unified modeling framework
Executable models
Design automation
Model development
Cyber-Physical Systems (CPS)
Unified Modeling Language (UML)
Model Driven Development (MDD)
Model Based Design (MBD)

ABSTRACT

Existing techniques for developing large scale complex engineering systems are predominantly software based and use Unified Modeling Language (UML). This leads to difficulties in model transformation, analysis, validation, verification and automatic code generation. Currently no general frameworks are available to bridge the *concept-code* gap rampant in design and development of complex, software-intensive mechatronic systems called cyber-physical systems. To fill this gap and provide an alternative approach to Object Management Group's UML/SysML/OCL combination, we propose: **B**ond **G**raph based **U**nified Meta-**M**odeling **F**ramework (**BG-UMF**). BG-UMF is a practical and viable alternative and uses a novel hybrid approach based on model unification and integration. The focus is on conceptual design and development of executable models for large systems. The viability of the framework is demonstrated through an application scenario: conceptual design and development of a navigation and control system for a rotor-craft UAV.

1. Introduction

Model Driven Architecture (MDA) transformed software design. MDA's paradigm conceptual shift from *"everything is an object"* to *"everything is a model"* and automatic model transformations aims to increase levels of design automation. MDA promised automatic *Model Transformation* through 3 stages based on levels of platforms independence: *Platform Specific Model (PSM)* at the lowest level of model hierarchy, *Platform Independent Model (PIM)* in the middle and *Computation Independent Model (CIM)* at the highest level of model hierarchy [1]. However, in reality though MDA has helped automate transformation from PIM to PSM to Code, CIM to PIM transformation is done manually.

CIM frameworks and contents of CIM corresponding to the notion of "computationally independence" are rarely discussed [2]. The major focus and contributions of this work are *meta-modeling* of the more elusive CIM to PIM transformation (Fig. 3),

* Corresponding author.
E-mail addresses: veera.ragavan@monash.edu (S. Veera Ragavan), madhavan.shanmugavel@monash.edu (M. Shanmugavel), ganapathy.v@ktr.srmuniv.ac.in (V. Ganapathy), bijan.shirinzadeh@monash.edu (B. Shirinzadeh).

http://dx.doi.org/10.1016/j.robot.2015.05.003
0921-8890/

Nomenclature

AO	Actor Oriented
BG	Bond Graph
BG-UMF	Bond Graph based Unified Modeling Framework
C2C	Concept to Code
CIM	Computation Independent Model
CPS	Cyber Physical Systems
DOF	Degrees of Freedom
FBS	Function-Behavior-Structure
FM	Functional Modeling
IPM	Ideal Physical Model
MBD	Model Based Design
MDA	Model Driven Architecture
MDD	Model Driven Development
MOC	Model of Computation
MT	Model Transformation
OCL	Object Constraint Language
OMG	Object Management Group
OO	Object Oriented
PIM	Platform Independent Model
PSM	Platform Specific Model
RUAV	Rotor-craft UAV
SCAP	Sequential Causal Assignment Procedure
UMF	Unified Modeling Framework
UML	Unified Modeling Language
WBG	Word Bond Graph

to bridge the CIM–PSM gap and showcasing Bond Graphs (BG) as a viable, formal alternative meta-language for domain-independent modeling (Fig. 2) of engineering systems.

Unified modeling language (UML) is the de facto language of the software development community for MDA. Through UML, the object management group (OMG), aims to define a standard language for specifying, visualizing, constructing and documenting all the artifacts of a software system. Though MDA is promising, it also has many pitfalls [3] and lacks pragmatics [4]. *Model Transformation* in MDA is defined as *the automatic generation of a target model from a source model* [1]. If model transformations are the "heart and soul" of MDA [5], it is also currently the missing link [6]. Moreover UML cannot be used as a simulation modeling language as it lacks semantics for the modeling notations used and hence requires formal languages such as Object Constraint Language (OCL) for constraint propagation. Compatibility issues in UML models make formal validation and verification difficult. The lack of formal UML semantics is one of the main obstacles for engineering real CPS applications with UML; which we seek to offset using graph grammars. Moreover based on our experiences, UML and SysML are not the best options for engineering coupled dynamic systems such as *Cyber-Physical Systems (CPS)*.

CPS is a new breed of complex, software-intensive mechatronic system and calls for a close coupling between the physical and computing domains. Hence as shown in this paper, a different methodological focus is required: Firstly on getting the physics right; Secondly on maintaining absolute models in the meta-physical level as functional and behavioral models as long as possible in order to retain model semantics and for added flexibility. The rest is mathematics and automatic model transformation. This paper discusses the experiences of using a *Model Based Design* approach and BG-UMF to bridge the concept-implementation gap in CPS.

C2C or *Concept to Code* implementations are nontrivial. C2C implementations with focus on software development has been attempted at an elementary level, but not for complex engineering systems. Very few concept development tools exist [7]. Two major issues are: Automatic generation of executable code and maintenance/troubleshooting of the generated code. Development of C2C framework is an enormous and challenging task especially due to the dynamic nature and lack of support by Object-Oriented frameworks [8]. Locating the relevant feature in the code is difficult, even with formal concept analysis [9] and none of concept location methods are perfect [10,11], and far from supporting "*Round-Trip*" engineering.

> *None of the existing approaches support a full path from requirements …to system prototype where all transitions are performed by model transformations.* [12]

We propose BG-UMF for C2C design implementations. BG-UMF provides as output, the much needed CIM model in a ready form suitable for model transformations. Supported by automatic CIM to PIM/PSM model transformations, as shown in Fig. 3(c) complete C2C design automation becomes a reality.

Main contribution and novelty of this work lies in using Bond Graph (BG) as a middleware for meta-modeling[1] thereby eliminating the need for textual languages like OCL. We use BG elements for visual representation, BG causality for formal constraint propagation and BG graph grammars for formal verification of the dynamic evolution of the network structure. Unlike other methods, the main advantage in BG causality is that it allows constraint propagation in both directions, irrespective of the direction of power or information flow.

Secondly, the additional novelty here is to extend traditional modeling framework using *Word Bond Graph (WBG)* to include the notion of Model of Computation (MOC) similar to Ptolemy and Modelica and thereby extending the object-oriented (OO) to actor-oriented (AO) framework [14–16] for handling concurrency and timeliness critical to *CPS*.

Thirdly, we employ the immensely appealing idea of Bond Graph (BG) causality for constraint propagation and a small set of 9 simple BG elements to completely represent physical subsystems across multiple domains and compose these systems as *Unified Physical Systems* to generate differential algebraic equations automatically. For this, we propose and detail a systematic methodology using BG grammar.

Last but not least, we developed a novel and unified approach based on BG-UMF; a BG graph grammar based meta-modeling framework, as a practical and viable alternative to OMG's UML/SysML/OCL combination for meta-modeling CPS. The developed framework is collaborative and hierarchical, encompassing all phases of design from concept to code (executable), leveraging on existing automatic code generation tools. In a collaborative development environment, it provides the much needed support to explore the conceptual design space and analyze the effects of individual model decisions on the overall design. This is important, as conceptional development tools are scarce [7].

The rest of the paper is organized as follows. Section 2 provides a survey on the state of art, the motivation and rationale behind using graph grammars and importantly the need for our alternate framework—BG-UMF. In Section 3 we propose the Unified Modeling Framework (UMF) we developed as a systematic design methodology for CPS. Here we present the fundamentals for a systematic design methodology for conceptual design using BG grammar and unified meta-modeling framework (BG-UMF) for design and validation of dynamic systems. Benefits of using BG-UMF is discussed in Section 3.3. Section 4 shows step by step

[1] *Meta-model is the embodiment of a modeling paradigm* [13]. BG can be used for Meta-model composition, as it contains and governs *the set of axioms, notions, idioms, abstractions, and techniques that govern how systems within the domain are to be modeled.*

Please cite this article in press as: S. Veera Ragavan, M. Shanmugavel, V. Ganapathy, B. Shirinzadeh, Unified meta-modeling framework using bond graph grammars for conceptual modeling, Robotics and Autonomous Systems (2015), http://dx.doi.org/10.1016/j.robot.2015.05.003