Bo LIU, Yuan-rui ZHANG, Xue-lian CAO, Yu LIU, Bin GU, Tie-xin WANG, 2020.
A survey of model-driven techniques and tools for cyber-physical systems.
Frontiers of Information Technology & Electronic Engineering, 21(11):1567-1590.
https://doi.org/10.1631/FITEE.2000311

A survey of model-driven techniques and tools for cyber-physical systems

Key words: Cyber-physical systems; Model-driven approach; System modeling; Software engineering

Contact: Bo LIU
E-mail: liubocq@swu.edu.cn
ORCID: https://orcid.org/0000-0002-9026-2543

Motivation

- The construction of a cyber-physical system (CPS) is a hard and complex engineering process due to the nature of integrating a large number of heterogeneous subsystems.
- Trial-and-error approach and ad-hoc processes are not able to deal with the increasing complexity of CPS construction.
- Model-driven approaches are regarded to be systematic and effective in engineering CPS.
- Extensive research has been conducted on model-driven development (MDD) of CPS; however, there are few surveys that perform systematic analysis and discussion about the MDD methods, techniques, and tools in particular.

Main idea

1. We aim to present a survey on the state-of-the-art model-driven approaches from a system engineering perspective, which
   - summarizes key technical challenges related to CPS construction,
   - introduces the most representative MDD techniques and tools, and
   - discusses their features and solutions to the challenges they support
2. Based on the above, we try to provide an overview of MDD techniques and tools for CPS researchers, and a guidance for CPS builders on how to apply these techniques and tools.

CPS challenges

The following CPS challenges are especially concerned:

1. C-I: challenges in modeling

   Complex heterogeneity is the major source of the following challenges in CPS modeling

2. C-II: challenges in verification

   CPS is in general mission-critical and has safety requirements for realtime, concurrency, etc.

3. C-III: challenges in system evolution

   CPS requires continuous evolution, while dynamically integrating various kinds of systems into the CPS is difficult.

4. C-IV: challenges in meeting diverse QoS requirements

   Diverse QoS attributes and constraints should be well captured and modeled for developing diverse CPS applications.

MDD tools

We survey 10 representative tools:

1. Architecture Analysis and Design Language (AADL)
2. UML/SysML/MARTE
3. Ptolemy
4. SCADE
5. Simulink/Stateflow
6. Modelica
7. Business Process Model and Notation (BPMN)
8. KeY/KeYmaera X
9. Event-B/Hybrid Event-B
10. Hybrid system related tools

Analysis of tools for CPS

1. Comparing modeling languages for CPS construction

Table 6 Comparison of modeling languages

Tool	Extension(s)	Modeling language	Executable semantics	Graphical notation	Target system
AADL	BAND-AiDe, AADL+	–	No	Yes	Real-time embedded systems & architecture of systems/BAND-AiDe: body area networks and devices; AADL+: cyber-physical systems
UML	SysML, MARTE	–	No	Yes	General systems & architecture of systems/ SysML: parametric systems; MARTE: real-time embedded systems
Ptolemy	CyPhySim	Dataflow & process network & discrete event model & ...	Yes	Yes	Hierarchical and heterogeneous systems & real-time systems & real-time embedded systems
SCADE		LUSTRE	Yes	Yes	Real-time embedded systems
Simulink/ Stateflow	PVS-Simulink	–	Yes	Yes	Hybrid & real-time systems & hierarchical systems
Modelica	OpenModelica	–	Yes	Yes	Complex and heterogeneous systems & hybrid & real-time systems
BPMN	BPMN4CPS	–	No	Yes	Business process of general systems/ BPMN4CPS: business process of CPS
KeY	KeYmaera X	Dynamic logic	Yes	No	Java programs/KeYmaera X: hybrid systems
Event-B	Hybrid Event-B	–	Yes	Yes	Highly concurrent systems & discrete event systems/hybrid systems & CPS
HCSP/HHL		–	Yes	Yes	Hierarchical and real-time systems & hybrid systems & CPS

– indicates that these tools provide self-defined modeling languages. The term after "/" shows the more specific types of systems that the modeling languages of the tool extensions are more suitable for modeling

Analysis of tools for CPS (Cont'd)

2. Comparing functionalities for CPS construction

Table 7 Comparison of functionalities

Tool/Extension	Model transformation	Physical modeling	Verification	Program synthesis	Simulation	System scale
AADL	No	No	No	C, Ada	No	Large-scale
AADL/BAND-AiDe	No	Yes	No	C, Ada	No	Large-scale
AADL/AADL+	No	Yes	No	C, Ada	Yes	Large-scale
AADL/BLESS	No	Yes	Yes	C, Ada	No	Large-scale
UML	No	No	No	*	No	Large-scale
UML/SysML	No	Yes	No	*	No	Large-scale
UML/MARTE	No	No	No	*	No	Large-scale
Ptolemy	Yes	Yes	No	C, Java, VHDL	Yes	Moderate
Ptolemy/CyPhySim	Yes	Yes	No	C, Java, VHDL	Yes	Moderate
SCADE	Yes	No	Model checking	C, Ada	Yes	Large-scale
Simulink/Stateflow	Yes	Yes	Model checking	C/C++	Yes	Moderate
Simulink/Stateflow/ PVS-Simulink	Yes	Yes	Model checking & theorem proving	C/C++	Yes	Moderate
Modelica	No	Yes	No	No	Yes	Moderate
Modelica/OpenModelica	No	Yes	No	No	Yes	Large-scale
BPMN	Yes	No	No	No	No	Large-scale
BPMN/BPMN4CPS	Yes	Yes	No	No	Yes	Large-scale
KeY	No	No	Theorem proving	No	No	Small
KeY/KeYmaera X	No	Yes	Theorem proving	No	No	Small
Event-B/Hybrid Event-B	No	Yes	Theorem proving	No	Yes	Large-scale
HCSP	Yes	Yes	Theorem proving	System C	Yes	Large-scale

* A wide range of languages

Analysis of tools for CPS (Cont'd)

3. Support for coping with CPS challenges

Table 8 Support for coping with CPS challenges

Tool	C-I	C-II		C-III	C-IV
	Modeling (heterogeneity)	Time	Concurrency	System evolution	Meeting diverse QoS
AADL	Yes	Yes	Yes	Partly supported	Well supported
AADL/BAND-AiDe	Yes	Yes	Yes	Partly supported	Well supported
AADL/AADL+	Yes	Yes	Yes	Partly supported	Well supported
UML	Yes	No	No	Partly supported	Partly supported
UML/SysML	Yes	No	Yes	Partly supported	Partly supported
UML/MARTE	Yes	Yes	Yes	Partly supported	Partly supported
Ptolemy	Yes	Yes	Yes	Well supported	Well supported
Ptolemy/CyPhySim	Yes	Yes	Yes	Well supported	Well supported
SCADE	Yes	Yes	Yes	Partly supported	Well supported
Simulink/Stateflow	Yes	Yes	Yes	Partly supported	Well supported
Simulink/Stateflow/ PVS-Simulink	Yes	Yes	Yes	Partly supported	Well supported
Modelica	Yes	Yes	Yes	Partly supported	Well supported
Modelica/OpenModelica	Yes	Yes	Yes	Partly supported	Well supported
BPMN	No	No	No	Not well addressed	Partly supported
BPMN/BPMN4CPS	No	Yes	No	Not well addressed	Partly supported
KeY	No	No	No	Not well addressed	Partly supported
KeY/KeYmaera X	No	Yes	No	Not well addressed	Partly supported
Event-B	Yes	No	Yes	Partly supported	Well supported
Event-B/Hybrid Event-B	Yes	Yes	Yes	Partly supported	Well supported
HCSP	Yes	Yes	Yes	Partly supported	Well supported

Major results

1. Although CPS is still in its infancy, the realms of embedded systems and hybrid systems are strongly CPS-related. Through analysis in this survey, we found that many model-driven techniques and tools that have been working well in the above realms, still work in CPS construction.
2. CPSs are typically multidisciplinary and highly complex, such that we have to carefully choose a set of mature techniques and tools. These tools are heterogeneous themselves, which puts forward the urgent demand for a unified modeling platform.

Conclusions

1. We present a survey of research on model-driven development of CPSs from a system engineering perspective, in which:
   - the widely used techniques and tools are introduced
   - comparative analyses are presented from various perspectives, including their modeling languages, functionalities, and the challenges which they address in CPS design.
2. We believe that model-driven approaches are an inevitable choice in building CPSs, and further research effort is needed in the development of model-driven theories, techniques, and tools.
3. We argue that a unified modeling platform is needed. Such a platform would benefit research in the academic community and practical development in industry, and improve the collaboration between these two communities.

刘波(1981-)，男，四川广安人，2012年于重庆大学获工学博士学位，现为西南大学计算机与信息科学学院讲师，西南大学软件研究与创新中心主任助理。主要研究领域为人机物融合系统、服务与区块链系统。发表论文20余篇，参与或主持国家及省部级项目10余项。

顾斌(1968-)，男，博士，中国空间技术研究院研究员，博士生导师。长期从事航天器控制系统、计算机控制、可信软件等的研制工作。

王铁鑫(1987-)，男，辽宁抚顺人，2015年于法国IMT Mines Albi获博士学位，现为南京航空航天大学讲师，CCF会员。主要研究领域为模型驱动工程、协同合作管理等。发表论文20余篇，主持或承担过国家基础研发子项目及青年创新项目等。