



T.R.  
EGE UNIVERSITY  
Graduate School of Applied and Natural  
Science



# **Model-driven Engineering of Cyber-Physical Systems Software: A Systematic Literature Review**

**MSc THESIS**

Mustafa Abshir MOHAMED

International Computer

İzmir  
2020



T.R.  
EGE UNIVERSITY  
Graduate School of Applied and Natural Science

# **Model-driven Engineering of Cyber-Physical Systems Software: A Systematic Literature Review**



Mustafa Abshir Mohamed

Supervisor: Assoc. Prof. Dr. Geylani Kardaş

Co-Supervisor: Dr. Moharram Challenger

International Computer  
Information Technologies, Second Cycle Programme.

İzmir  
2020



Mustafa Abshir Mohamed tarafından Yüksek Lisans tezi olarak sunulan “Model-driven Engineering of Cyber-Physical Systems Software: A Systematic Literature Review” başlıklı bu çalışma EÜ Lisansüstü Eğitim ve Öğretim Yönetmeliği ile EÜ Fen Bilimleri Enstitüsü Eğitim ve Öğretim Yönergesi'nin ilgili hükümleri uyarınca tarafımızdan değerlendirilerek savunmaya değer bulunmuş ve 03/02/2020 tarihinde yapılan tez savunma sınavında aday oybirliği/oyçokluğu ile başarılı bulunmuştur.

**Jüri Üyeleri:**

**Jüri Başkanı** : Doç. Dr. Geylani KARDAŞ  
**Raportör Üye** : Dr. Öğr. Üyesi İlker Kocabaş  
**Üye** : Doç. Dr. Tolga Ayav

**İmza**

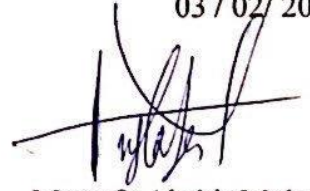
  
.....  
  
.....  
  
.....



## EGE ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ ETİK KURALLARA UYGUNLUK BEYANI

EÜ Lisansüstü Eğitim ve Öğretim Yönetmeliğinin ilgili hükümleri uyarınca Yüksek Lisans Tezi olarak sunduğum “Model-driven Engineering of Cyber-Physical Systems Software: A Systematic Literature Review” başlıklı bu tezin kendi çalışmam olduğunu, sunduğum tüm sonuç, doküman, bilgi ve belgeleri bizzat ve bu tez çalışması kapsamında elde ettiğimi, bu tez çalışmasıyla elde edilmeyen bütün bilgi ve yorumlara atıf yaptığımı ve bunları kaynaklar listesinde usulüne uygun olarak verdiğimi, tez çalışması ve yazımı sırasında patent ve telif haklarını ihlal edici bir davranışımın olmadığını, bu tezin herhangi bir bölümünü bu üniversite veya diğer bir üniversitede başka bir tez çalışması içinde sunmadığımı, bu tezin planlanmasından yazımına kadar bütün safhalarda bilimsel etik kurallarına uygun olarak davrandığımı ve aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul edeceğimi beyan ederim.

03 / 02 / 2020



Mustafa Abshir Mohamed





**ÖZET**  
**Siber Fiziksel Sistem Yazılımlarının Model-güdümlü Mühendisliği: Bir**  
**Sistemik Literatür İncelemesi**

MOHAMED, Mustafa Abshir

Yüksek Lisans Tezi, Uluslararası Bilgisayar Anabilim Dalı

Tez Danışmanı: Doç. Dr. Geylani Kardaş

İkinci Danışmanı: Dr. Moharram Challenger

Şubat 2020, 130 sayfa

Günümüzde, Siber-Fiziksel Sistemlerin (CPS) geliştirilmesi, esasen sanayi ve toplumun hemen her alanında sundukları fırsatlar nedeniyle, gün geçtikçe hem araştırmacıların hem de endüstrideki uygulayıcıların daha fazla ilgisini çekmektedir. Bununla birlikte, CPS'lerin geliştirilmesi ve yönetilmesi, sahip olduğu heterojen ve karmaşık özellikleri nedeniyle zorlu bir iştir. CPS'lerin geliştirilmesi ile ilgili olarak, alanın karmaşıklığı nedeniyle standart bir metodoloji henüz yoktur. Bu sistemler için geliştirme karmaşıklığını azaltmaya yönelik uygulanan önemli yaklaşımlardan biri birçok alanda yazılım geliştirmede sıklıkla kullanılan ve platformdan soyutlama seviyesini arttıran Model-güdümlü Mühendislik (MDE)'tir.

Bu tez, MDE paradigması aracılığıyla CPS geliştirmeyi içeren son araştırmaların belirlenmesine ve sınıflandırılmasına odaklanan sistemik bir literatür taraması (SLR) çalışmasını sunmaktadır. Çalışma kapsamında 2010-2018 yılları arasında yayınlanan 140 araştırma makalesi ele alınmış ve incelenmiştir. Buna göre, çeşitli MDE yaklaşımlarının kapsamlı bir analizi ve araçları sunulmuştur. Çalışma ayrıca literatürdeki boşlukları ve doldurulması gereken alanları da belirlemiştir. Böylece çalışma sonuçlarının araştırmacıların ve geliştiricilerin alandaki yönlerini bulmalarına ve daha ilerideki araştırmaları için mevcut zorlukları genel olarak anlamalarına yardımcı olacağına inanılmaktadır.

**Anahtar Kelimeler:** Siber Fiziksel Sistemler, CPS, Model-güdümlü Mühendislik, MDE, Sistemik Literatür İncelemesi, SLR, İkincil çalışma.



**ABSTRACT**  
**Model-driven Engineering of Cyber-Physical Systems Software: A  
Systematic Literature Review**

MOHAMED, Mustafa Abshir

MSc Thesis, International Computer

Supervisor: Assoc. Prof. Dr. Geylani Kardaş

Co-Supervisor: Dr. Moharram Challenger

February 2020, 130 pages

Nowadays, the development of Cyber-Physical Systems (CPS) draws more interest from both researchers and practitioners considering the opportunities they offer in almost all areas of industry. However, development and management of CPS are challenging tasks due to their inherent heterogeneity and complexity characteristics. Regarding the development of CPS, there currently exists no standard methodology owing to the complexity of the domain. One of the key approaches to reduce the development complexity for CPS is Model-driven Engineering (MDE), which is frequently used in many business domains for software development to increase the level of platform abstraction.

This thesis presents a Systematic Literature Review (SLR) study that focuses on identifying and classifying the recent research practices pertaining to CPS development by applying MDE approaches. The study evaluates 140 research papers published during years 2010–2018. Accordingly, a comprehensive analysis of various MDE approaches and tools used in the development life-cycle of CPS is provided. Furthermore, the study identifies the research gaps and areas which need more investigation. Conducted study may help researchers and practitioners to get an overall understanding of the research trends and existing challenges for further research and development.

**Keywords:** Cyber-physical Systems (CPS), Model-driven Engineering (MDE), Systematic Literature Review (SLR), Secondary study.



## FOREWORD

Engineers come across vital challenges when developing complex systems like Cyber-physical systems (CPS) due to their heterogeneous nature, e.g. the need for knowledge and skills from multiple academic disciplines, the integration of the artifacts of those various disciplines, and the difficulty of the maintenance activities of such heterogeneous artifacts. The development of CPS needs a unified methodology that permits efficient raise of the abstraction level to overcome issues of heterogeneity induced by the multidisciplinary nature of the system. Model-driven engineering (MDE) is believed to be an alternative solution to overcome the challenges faced while developing CPS.

Reporting on what previous researches achieved as well as the current research efforts and open challenges related to this field may guide researchers and developers for their further work on the design and implementation of CPS. Therefore, this thesis presents a systematic literature review on employing MDE paradigm in CPS, mainly finding out the followed approaches when applying MDE for CPS, used tools/languages and addressed CPS application domains.



## TABLE OF CONTENTS

	<u>Page</u>
ÖZET .....	V
ABSTRACT .....	VII
FOREWORD .....	IX
TABLE OF CONTENTS.....	XI
LIST OF FIGURES .....	XIII
LIST OF TABLES.....	XV
ACRONYMS .....	XVII
1. INTRODUCTION.....	1
1.1 CYBER-PHYSICAL SYSTEMS.....	1
1.2 MODEL DRIVEN ENGINEERING.....	2
1.3 METHODOLOGIES FOR SECONDARY STUDIES.....	3
1.4 PROBLEM STATEMENT .....	4
1.5 CONTRIBUTIONS.....	5
2. RELATED WORK.....	6
3. METHODOLOGY .....	9
3.1. PROCESS .....	9
3.2. TEAM.....	10
3.3. RESEARCH QUESTIONS .....	11
3.4. SEARCH AND SELECTION STRATEGY .....	13
3.5 INCLUSION & EXCLUSION CRITERIA (SELECTION CRITERIA).....	16
3.5.1 Inclusion criteria .....	17
3.5.2 Exclusion criteria .....	17
3.6 STUDY QUALITY ASSESSMENT AND SELF-ASSESSMENT .....	18
3.7 DATA EXTRACTION.....	18
4. PERFORMING SYSTEMATIC REVIEW .....	20
4.1 PERFORMING AUTOMATIC SEARCH.....	20
4.2 REMOVING DUPLICATES.....	21
4.3 SELECTING PRIMARY STUDIES .....	22

**TABLE OF CONTENTS (continued)**

	Page
4.4 FORWARD SNOWBALLING .....	23
4.5 REALIZING DATA EXTRACTION.....	23
4.6 DATA ANALYSIS .....	24
5. RESULTS.....	25
5.1 BIBLIOMETRICS & DEMOGRAPHICS ANALYSIS .....	25
5.1.1 Publication trend per year .....	25
5.1.2 Citation analysis and top-cited studies .....	25
5.1.3 Active researchers in the domain .....	26
5.1.4 Countries contributing to the field (based on author affiliations) .....	27
5.1.5 Publication venues .....	28
5.2 QUALITY ASSESSMENT .....	30
5.3 RESEARCH QUESTIONS ANALYSIS .....	31
5.3.1 Modeling approaches employed for applying MDE in CPS .....	31
5.3.2 Developed/used MDE tools for CPS.....	55
5.3.3 Addressed CPS components .....	59
5.3.4 Targeted CPS application domains .....	61
5.3.5 Conducted evaluations for the proposed solutions.....	65
5.3.6 Addressed CPS challenges .....	66
6. DISCUSSION .....	71
6.1 THREATS TO VALIDITY .....	74
7. CONCLUSION.....	77
REFERENCES .....	78
ACKNOWLEDGEMENT.....	81
RESUME.....	82
APPENDIXES .....	



## LIST OF FIGURES

<u>Figure</u>	<u>Page</u>
Figure 1.1: CPS architecture .....	1
Figure 3.1: Overview of the systematic literature review process .....	10
Figure 3.2: Search and selection strategy .....	14
Figure 4.1: Search and selection process .....	23
Figure 5.1: Publication trend per year. ....	25
Figure 5.2: (a) Number of citation per year (b) Median number of citations .....	26
Figure 5.3: Authors with at least three papers.....	27
Figure 5.4: Countries contributing to the field (based on author affiliations). ....	28
Figure 5.5: Reported modeling approaches. ....	32
Figure 5.6: Distribution of the reported modeling approaches over the years .....	33
Figure 5.7: Reported modeling approaches vs. Authors' countries. ....	34
Figure 5.8: Reported activities/purposes for which modeling approaches used. ...	36
Figure 5.9: Modeling approaches distribution over modeling activities. ....	38
Figure 5.10: Reported MDE phases/activities.....	40
Figure 5.11: Most used tools/languages which are at least reported by 2 studies. ....	56
Figure 5.12: Frequency and type of the developed tool/language.....	57
Figure 5.13: Used languages/frameworks for developing DSMLs, Metamodels, or tools.....	58
Figure 5.14: Distribution of languages/frameworks used for developing DSMLs, metamodels, tools over publication years. ....	58
Figure 5.15: Distribution of languages/frameworks used for developing DSMLs, metamodels, tools over authors' affiliation country. ....	59
Figure 5.16: Reported CPS component(s) .....	60
Figure 5.17: Reported CPS application domains targeted by the studies. ....	62
Figure 5.18: Distribution of the conducted evaluations' categories over the evaluation type.....	65
Figure 5.19: Categorized CPS challenges reported by the studies. ....	66







## LIST OF TABLES

<u>Table</u>	<u>Page</u>
Table 2.1: Summarized Related work.....	6
Table 3.1: PICOC criteria definition. ....	11
Table 3.2: Digital Libraries .....	14
Table 3.3: Keywords definition based on PICOC criteria .....	15
Table 3.4: Quality assessment and self-assessment.....	18
Table 3.5: Data extraction form .....	19
Table 4.1: Search strings.....	21
Table 5.1: Most cited papers .....	26
Table 5.2: Venues with at least two papers.....	29
Table 5.3: Quality assessment results analysis.....	30
Table 5.4: Developed DSLs, Meta-models, tools, and extensions.....	42
Table 5.5: Reported simulation tools and languages .....	44
Table 5.6: Reported model transformations, and used transformation tools and languages.....	46
Table 5.7: Reported V&V and the used tools .....	49
Table 5.8: Reported tools/languages used for modeling and the studies used them. .....	51
Table 5.9: Reported code generation languages/tools and studies used them .....	53
Table 5.10: Reported System analysis languages/tools .....	54
Table 5.11: Addressed CPS components and the corresponding studies. ....	61
Table 5.12: CPS application domain correlated with the evaluations presented by the studies.....	64
Table 5.13: CPS challenges and their corresponding studies. ....	68
Table 5.14: Correlation analysis between CPS application domains and its challenges.....	70









**ACRONYMS**

<u>Symbol</u>	<u>Definition</u>
CPS	Cyber-Physical System(s)
DSL	Domain-specific Language
DSML	Domain-specific Modeling Language
EMF	Eclipse Modeling Framework
GME	Generic Modeling Environment
MBE	Model-based Engineering
MBSE	Model-based System Engineering
MDA	Model-driven Architecture
MDE	Model-driven Development
PICOC	Population Intervention Comparison Outcome Context
SLR	Systematic Literature Review
SM	Systematic Mapping
SoP	State-of-Practice
UML	Unified Modeling Language
WebGME	Web Generic Modeling Environment







## 1. Introduction

This thesis presents a Systematic Literature Review (SLR) study that focuses on identifying and classifying the recent research practices pertaining to Cyber-Physical System development by applying model-driven engineering approaches. In the following subsections, cyber-physical systems and model-driven engineering are briefly introduced first. Then the motivation behind the conducted study is discussed with the problem statement in addition to the contributions.

### 1.1 Cyber-physical systems

The first emerge for the term ‘‘Cyber-Physical system’’ (CPS) was in 2006 at the National Science Foundation (Lee, 2015). CPS is a system that its computation and communication components control and monitor physical phenomena (Lee, 2008). In CPS, sensors monitor and measure certain physical phenomena, like pressure, temperature, light, touch, etc. The measured data are transferred to the controllers/software through communication elements (i.e. wired/wireless network). The controllers/software make decisions/actions based on the received data from the sensors and send them through communication elements to actuators which in return make changes to the physical phenomena (Sanislav et al., 2017). The overall architecture of a CPS is depicted in (Figure 1.1).

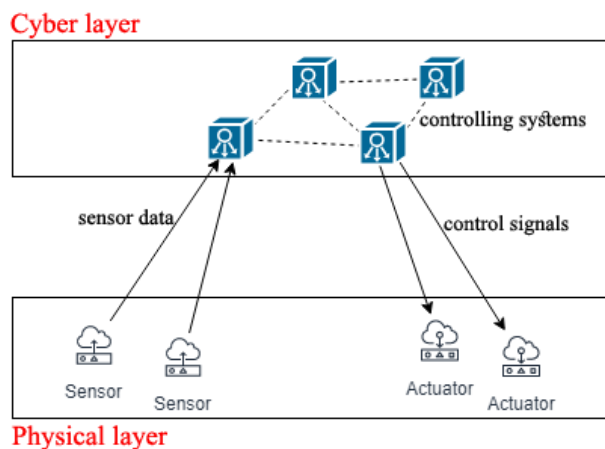


Figure 1.1: CPS architecture

Applications of CPS include, but are not limited to, monitoring complex real-world phenomena, smart manufacturing (i.e. industry 4.0), smart building, critical infrastructures, like chemical and power plants, smart grids, natural gas distribution systems, transportation systems, etc. (Lee, 2008).

Significant challenges come across developers of CPS due to its heterogeneous nature, such as the need for knowledge and skills from multiple academic disciplines, the integration of the artifacts of those various disciplines, and the difficulty of the maintenance activities of such heterogeneous artifacts. As a practical example, developing a CPS requires a group of developers from different disciplines like software engineering, electrical engineering, electronics engineering, industrial engineering, and so on. This induces communication challenges among these engineers due to the different tools and abstractions used in each discipline. Another challenge can be the time consumed to comprehend and integrate the codes written by these different developers. The maintenance of such a code with various levels of abstractions is executive or almost impossible. In order to eliminate these challenges and reduce the complexity of CPS development, one of the key approaches is Model-driven Engineering (MDE), which is frequently used in many business domains for software development (France and Rumpe, 2007).

## **1.2 Model driven engineering**

In general, models have two features. Reduction feature where the models focus on the main properties of a system and neglect the details to keep the representation of the system relatively easier, and Mapping feature whereby models are generalized from a prototype of the original system. Models could be used for different purposes such as sketches, blueprints, or programs. There is an increasing need for the use of such models in software development for the following reasons (Brambilla et al., 2017);

- the increasing demand for accelerating the development process.
- the increasing complexity of software artifacts relative to the software functionality demanding abstraction level to be raised to facilitate the maintenance process or future upgrading.

- the need for a medium language between the non-developers (e.g., customers, managers, business stakeholders, etc.) and the software developers.

The use of models as the basic building blocks for the development of software artifacts is called Model-driven Engineering (MDE). MDE paradigm raises the abstraction level of software/system development from low-level artifacts to a higher-level of models. MDE bridges the gap between problem identification and software implementation phases. This can be done by thoroughly/partially generating either software implementations (C++, Java, and C#) or deployment artifacts (XML-based configuration) from models that describe the system at multiple levels of abstraction, and from a variety of perspectives (France and Rumpe, 2007).

### **1.3 Methodologies for secondary studies**

Generally, to conduct a secondary study, there are several methodologies that can be followed. Survey research is one of them. As it is proposed by (Pinsonneault and Kraemer, 1993), there is a significant difference between survey research and survey. A survey is a quantitative method that aims to collect data about features, behaviors or opinions of a specific group of people as a representative for a target population. On the other hand, survey research is a methodology that is specified to conduct surveys for advanced scientific knowledge or research purposes.

Systematic Mapping (SM) is a method of collecting previously written research papers, articles, conference papers, book chapters etc. in a certain area of study based on a set of questions made by the researchers. Collected documents can be later reviewed, analyzed and structured in different categories to provide a wider view of the research area. Thereby, it could be much easier to determine those areas which need more research studies to be performed. In this regard, it serves as a valuable basis for future researches (Petersen et al., 2008).

Systematic Literature Review (SLR), also known as a systematic review, is a form of a secondary review that aims to identify, evaluate and interpret all the available researches, relevant to particular research questions, topic area, or

phenomenon of interest. It collects and critically scrutinizes data from the studies included in the review (aka. primary studies) (Kitchenham and Charters, 2007).

The procedure used in conducting SM and SLR is nearly identical. However, they are different in terms of their goals, processes, and results. Firstly, SM focuses on categorizing the conducted studies based on a thematic analysis. Whereas, SLR focuses on empirical evidence in its categorization. Secondly, SM studies show the research gaps in a specific area of study. In addition to these features of SM, SLR makes an in-depth comparison between tools, techniques, and approaches proposed by the primary studies. In terms of the process, in the inclusion and exclusion phase of the SM, thematic analysis (e.g., reading the title, abstract, introduction, and conclusion sections of the publication) is used. Meanwhile in SLR, the studies shall be read in-depth (e.g. by additionally reading the methodology and other required sections in the publication).

In conclusion, SLR can be an intensive and more-in-depth version of SM. In this study, SLR is adopted as the research methodology. This study will be achieved by following the process proposed by (de Almeida Biolchini et al., 2007; Siddaway, 2014) and using guidelines defined by (Kitchenham and Charters, 2007).

#### **1.4 Problem statement**

A unified development methodology for CPS has not been standardized yet. The abundance of different hardware platforms available makes the development of such systems very complex. There is a need for a unified methodology that permits efficient raise of the abstraction level to overcome issues of heterogeneity induced by the multidisciplinary nature of the system. Towards this goal, many researchers believe that MDE is a better alternative solution to overcome challenges such as development complexity, heterogeneity, adaptability, and reuse and they propose various applications of MDE for CPS development. However, no secondary study highlighting 1- previous researches 2- current research efforts 3- open challenges related to applying MDE approaches for CPS has been done yet. This overview would be helpful to both researchers and practitioners for discovering the pros and cons for applying MDE in CPS and for identifying



interesting research directions. Without secondary study, it is cumbersome to determine what was proposed, what has been successfully completed and what rather has failed.

## **1.5 Contributions**

The aim of this thesis is conducting an SLR of the studies used MDE techniques such as Domain-specific modeling, Metamodeling, Validation & Verification, Model Transformation, and Artifact Generation for CPS. Evaluation of research questions and analysis of the proposed approaches and toolchain in the primary studies is performed as the result of the systematic review in the scope of this study. Furthermore, in this work, trends, bibliometrics and demographics are provided to help collecting important information such as; the active authors/researchers in this domain, the publication per year for each country and other information.

The results of this study may help the researchers to easily reach the desired class of studies and related publications considering the tools, technologies, approaches, and best practices used. This study also enables researchers avoid unnecessary duplications of trial and error. Finally, it identifies research gaps and areas need more investigations and determine best practices, tools, techniques, and languages which can be used.

The remainder of the thesis is organized as follows: Section 2 gives the related work. Section 3 describes the research methodology and protocol definition to carry out the SLR, then, the procedure of conducting the SLR is discussed in Section 4. Section 5 shows the results, where the discussion of the results and the conclusions are presented in sections 6 and 7 respectively.

## 2. Related Work

Since the scope of this thesis is to present an SLR study on the state-of-the-art of MDE for CPS, the secondary studies (surveys, SMs, SRLs, Tertiary Studies) are addressed in this section as the related work. Although, there is no any secondary study on this topic, the following studies can be found relevant. Table 2.1 presents a summary of these related work.

Table 2.1: Summarized Related work

Study	Methodology	Domain	Paradigm	# of Papers	Studies b/w years
Barišić et al., 2018	SLR	CPS	MPM	265	2006-2017
Rashid et al., 2015	SLR	Embedded Systems	MBSE	61	2008-2014
Queiroz et al., 2014	SLR	safety-critical embedded systems	PLE+MDE	19	-
Casalaro et al., 2015	SM	Mobile robot systems	MDE	69	After 2000
Liebel et al., 2018	Survey	embedded systems	MBE	-	-
Agner et al., 2013	Survey	embedded software	UML+MDA	-	-
The work in this thesis	SLR	CPS	MDE	140	2010-2018

In (Barišić et al., 2018), an SLR on multi-paradigm modelling for cyber-physical systems is presented where authors focused on studies promoting multi-modeling, multi-view and multi-formalism approaches for the development of CPS. The study reported the most used approaches and tools in the primary studies for multi-paradigm modeling as well as indicating the type of formalism presented, and which language/tool is used for implementing it. Furthermore, they reported the actors and stakeholders involved in the modeling process and their background knowledge.

Rashid et al. (2015) conducted SLR of the development of embedded systems using model-based system engineering (MBSE) approach. The study reviewed 61 research papers published in one of the four renowned scientific databases IEEE, SPRINGER, ELSEVIER, and ACM during the years 2008-2014. Subsequently, primary studies are grouped into six categories according to their

relevance to the corresponding model-based system engineering activity namely general category, modeling category, model transformation category, model verification category, simulation category, and property specification category. As the result, the study presented 28 tools which support modeling, model transformation, validation, and verification activities. The study examined the utilization of UML and SysML/MARTE profiles, and it also analyzed the application of both model-to-model and model-to-text transformations.

Another SLR is presented in (Queiroz et al., 2014) in which authors investigated studies combining product line engineering (PLE) and MDE for the development of safety-critical embedded systems. This study further examined whether there are empirical studies applied the aforementioned techniques in the development process of safety-critical embedded systems. The study exposed that in recent years, use of MDE combined with PLE techniques to build safety-critical embedded systems is gradually growing. The study also found out that the proposed approaches in the primary studies were not compared with any other related studies, besides, these approaches do not explicitly differentiate between the software and hardware variabilities.

An SM study is presented in (Casalaro et al., 2015), this study investigates the implementations of MDE in the field of mobile robot systems (MRS). In this study, 69 research papers were selected, and as a result, the authors found out that many domain-specific modeling languages (DSMLs) are supported with tools which are mostly built ad-hoc. Also, they reported that the solutions based on UML and using Eclipse-based tools were less preferred.

A survey is presented in (Liebel et al., 2018) in which the quantitative data from 113 subjects were collected to provide the current state-of-practice (SoP) and challenges faced by the domain of embedded systems due to weaknesses in model-based engineering (MBE). The survey has two research questions, the first question is related to capturing the state of MBE practice in the embedded systems domain, how much activities concern MBE compared to non-MBE, and understanding the pros and cons of adopting and deploying MBE. The second question is about estimating whether there are important variations in the SoP between different groups in the embedded systems domain. As a result, the study provided information about the used methods and tools, purposes of models,

effects of using it, and weaknesses of MBE. Furthermore, answers to the survey show that most of the participants think that the positive outcomes distinctly exceeded the negative outcomes of MBE. Nonetheless, survey participants mentioned weaknesses such as the interoperability challenges amongst existing tools, and MBE causes high efforts to train the developers.

Another survey is presented in (Agner et al., 2013). The study introduces statistical findings about the use of UML modeling and model-driven approaches for the design of embedded software in Brazil. The goal of this study is to identify gaps in the knowledge of how exactly UML and MDE or Model-driven Architecture (MDA) are used in industry, and to provide an understanding of how social and organizational factors impact the use of UML and MDE/MDA.

Unlike the work presented in (Rashid et al., 2015; Queiroz et al., 2014), this thesis work focuses on conducting SLR on studies concerning the development of CPS using the MDE paradigm. The work herein and the SLR given in (Barišić et al., 2018) both consider the development of CPS. However, this thesis differentiates from (Barišić et al., 2018) in terms of the results since this work finds MDE approaches and tools used for the development of CPS, the most addressed MDE phase, developed tools and languages in this regard, and also presenting reported CPS challenges by the primary studies. These were not taken into account in (Barišić et al., 2018).

### 3. Methodology

In this section, the applied methodology for the conducted SLR is discussed. In the following sections, initially, the process to be followed during this work is described, followed by defining research questions, search and studies selection strategy, then specifying inclusion and exclusion criteria and quality assessment and self-assessment criteria, and finally determining the procedure to follow for data extraction.

#### 3.1. Process

The procedure of the systematic review was developed by following the guidelines defined in (Kitchenham and Charters, 2007). Figure 3.1 shows an overview of the followed process. SLR composes three main phases; review planning, review execution, and review reporting (de Almeida Biolchini et al., 2007).

##### 1- Planning phase

- a. Protocol development: Research scope and review protocol are developed; the protocol is subject to improvement in later stages through an iterative manner.
- b. Determining research questions: Research questions are defined following the PICOC guidelines (Kitchenham and Charters, 2007).

##### 2- Execution phase

- a. Collecting studies: Search keywords are formalized, and then, studies are collected.
- b. Determining proper studies: Publications are included or excluded based on the inclusion and exclusion criteria defined in the protocol.
- c. Extracting data from studies: Information is extracted from the studies according to the research questions.
- d. Data analysis: Data extracted from primary studies are analyzed to answer the research questions.

##### 3- Reporting phase: This phase involves the systematic discussion and reports the outcomes of the analysis.

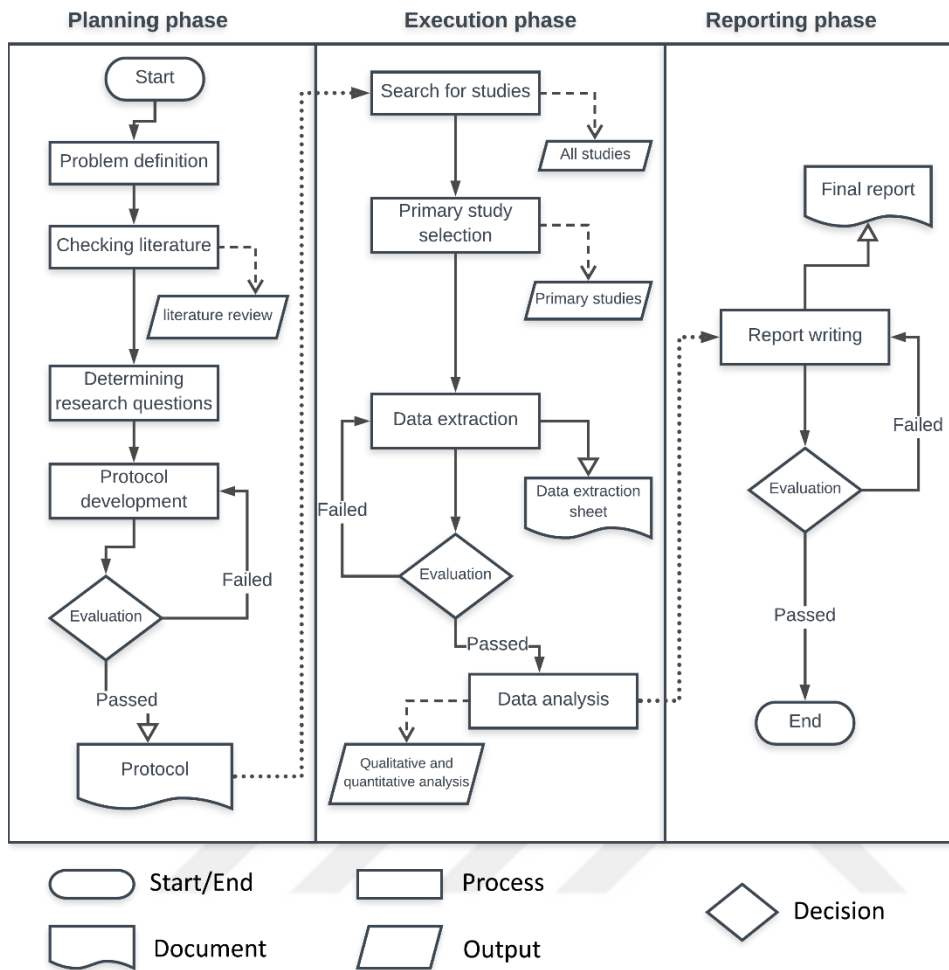


Figure 3.1: Overview of the systematic literature review process

### 3.2. Team

The team in this study consists of the following 3 people:

- **Primary researcher:** Mustafa Abshir Mohamed, M.Sc. student in Information Technologies at Ege University, Turkey. With knowledge of CPS and MDE, he conducted the SLR and carried out most of the tasks of the SLR study.
- **Secondary researcher:** Dr. Moharram Challenger, a postdoctoral researcher at University of Antwerp, Belgium. He is active in the fields of CPS and MDE. He also has knowledge in conducting SLR and SM studies. He regularly supported and reviewed the work performed by the primary researcher since he is also the co-supervisor of this thesis work.

- **Supervisor:** Dr. Geylani Kardaş, an associate professor at Ege University, Turkey. He is a senior researcher with many years of expertise in MDE. He supported the primary and the secondary researchers during protocol definition, analysis of the findings, and report writing. He also helped resolving the conflicts between the findings of the primary and the secondary researchers.

### 3.3. Research questions

In this study, the state-of-the-art MDE techniques in CPS are taken into consideration. To address them, research questions were identified by following PICOC criteria (Kitchenham and Charters, 2007) (see Table 3.1).

Table 3.1: PICOC criteria definition.

Population	CPS
Intervention	MDE techniques for CPS
Comparison	not applicable
Outcome	report on the current state-of-the-art approaches, languages, tools, and challenges of MDE for CPS.
Context	peer-reviewed publications.

Following the defined PICOC mentioned above, the research questions of this study are determined as below:

**RQ1:** Are any of MDE approaches or techniques used in/for the development of the studied CPS?

- **Objective:** With answering this question, the existing MDE approaches for CPS, modeling purpose, and the MDE phase addressed are reported.

**RQ1.1:** What is the modeling approach presented/used in the study?

**RQ1.1.1:** What is the purpose for which the models were used?

**RQ1.2:** Which phase(s) of the system development is/are addressed in the study (using MDE)?

**RQ2:** Is/Are there any tool(s) used to apply MDE in/for CPS in the study?

- **Objective:** With answering this question, the used languages and tools, also, developed tool(s) by the study, its availability and the used language to develop the tool are reported.

**RQ2.1:** Which language/tool(s) is/are presented/used in each phase of the system development?

**RQ2.2:** If any tool(s) is/are developed in the study, is/are this/these tool(s) reported?

**RQ2.3:** Is/are the developed tool(s) available and/or accessible?

**RQ2.4:** What is/are the framework(s) or programming language(s) for the development of this/these tools?

**RQ3:** What is/are the CPS component(s) addressed in the study?

- **Objective:** This question is aimed to report on the CPS component such as sensor, cyber component, physical component, actuator etc. that is modeled.

**RQ4:** Does the study present any application domain?

- **Objective:** It is aimed to report the CPS domain like critical infrastructure, Smart Buildings, Industry 4.0 etc. which the primary studies are targeting.

**RQ4.1:** What is the application domain?

**RQ4.2:** What is the use case?

**RQ5:** Is there any evaluation presented in the study?

- **Objective:** Reporting on the evaluation method followed by these primary studies such as case study, use case, example and empirical study.

**RQ5.1:** What is the evaluation approach?

**RQ5.2:** If the evaluation is based on a case study, what is the case study?

**RQ6:** Does the study address any challenge(s)?

- **Objective:** Reporting on the CPS challenges which primary studies are addressing, also, challenges addressed during tool development/usage by the primary studies.



**RQ6.1:** Which CPS challenge(s) does the paper address?

**RQ6.2:** Does the study report challenges addressed during developing the approach/tool?

### **3.4. Search and selection strategy**

This stage could be considered as one of the most important and critical stages when conducting a secondary study (i.e. in this case, SLR). Therefore, it should be carefully defined since the search of primary studies should ensure the comprehensive coverage of the topic under consideration. For a search strategy to be optimal, it needs to simultaneously include utmost relevant primary studies (i.e. recall) and exclude irrelevant ones (i.e. precision). One can deduce that an optimal search strategy must have 100% recall and/or 100% precision. Nevertheless, it is unpromising that a search strategy gives 100% in both/either recall and/or precision. Accordingly, one should come up with a gratifying trade-off search strategy (i.e. good enough), that results in not many relevant studies missed, and a manageable quantity of irrelevant studies included (Skoglund and Runeson, 2009).

The search strategy, developed in this thesis, composes four stages. Firstly, an automatic search over the most relevant scientific digital libraries was performed. Secondly, all duplicate papers were removed. Thirdly, following predetermined criteria of inclusion, only papers related to the topic were considered. Eventually, further studies were searched by forward snowballing. The composition of the search and selection strategy followed in this work is shown in Figure 3.2.

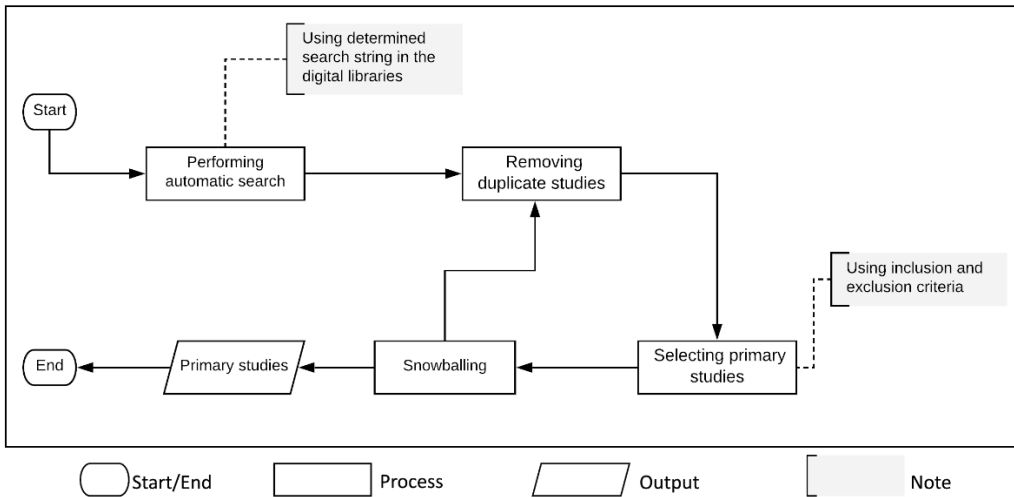


Figure 3.2: Search and selection strategy

### Stage 1: Performing automatic search

To get as many related primary studies as possible, an automatic search on the digital libraries shown in Table 3.2 was performed.

Table 3.2: Digital Libraries

Digital Library	URL	Access Date
ACM	<a href="https://dl.acm.org/">https://dl.acm.org/</a>	Oct/2018
Dblp	<a href="https://dblp.uni-trier.de/">https://dblp.uni-trier.de/</a>	Oct/2018
IEEE Xplore	<a href="https://ieeexplore.ieee.org/">https://ieeexplore.ieee.org/</a>	Oct/2018
ScienceDirect	<a href="https://www.sciencedirect.com/">https://www.sciencedirect.com/</a>	Oct/2018
Scopus	<a href="https://www.scopus.com/">https://www.scopus.com/</a>	Dec/2018
Web of Science	<a href="https://www.webofknowledge.com/">https://www.webofknowledge.com/</a>	Oct/2018

PICOC criteria (Kitchenham and Charters, 2007) were used to define the keywords shown in Table 3.3, which leads to form “good enough” search strings. These search strings are used when performing automatic search in the aforementioned digital libraries.

Table 3.3: Keywords definition based on PICOC criteria

Population	"cyber-physical system*" OR "cyber physical system*" OR "smart system*" OR "cyberphysical systems" OR "cps"
Intervention	MDE OR MDD OR MDA OR "model-driven *" OR "model driven *" OR "code generation" OR "generative approach*" OR "model-based approach*" OR "domain specific model*" OR metamodel* OR "meta-model*" OR "meta model*" OR "modeling approach"
Comparison	Not applicable
Outcome	Report on the current state-of-the-art approaches, languages, tools and challenges of MDE for CPS.
Context	Peer-reviewed publications.

The overall search string is as follows:

("model-driven development" OR "model-driven engineering" OR "model-driven architecture" OR "code generation" OR "generative approach" OR "model-based approach" OR "model-driven approach" OR "domain specific model\*" OR metamodel OR "meta-model" OR "meta model" OR "modeling approach")

**AND**

("cyber-physical system\*" OR "cyber physical system\*" OR "smart system\*" OR "cyberphysical systems" OR "cps")

Due to the different syntax of each digital library, a specific search string for each of these libraries was created. This is to ensure including as much relevant primary studies as possible.

## **Stage 2: Removing duplicate studies**

Initially, pool of primary studies was kept in Mendeley<sup>1</sup> reference manager. Also, this repository was used to facilitate the process of determining duplicate studies. Two papers are considered as duplicate if:

- their title, author(s), publication date and venue are the same. In case of different versions of the same paper, the most recent is kept.

<sup>1</sup> <https://www.mendeley.com>

- the same paper is published in different venues, one of them is selected (the most recent).
- the same study has both journal and conference publications, the journal publication is considered as it contains the extended study and provides more information.

### **Stage 3: Selecting primary studies**

In this stage, primary studies are selected following predefined inclusion and exclusion criteria (see Section 3.5). Only those studies matching the criteria are included in the final pool of the research. The criteria were applied considering the reading of TITLE, ABSTRACT, KEYWORDS, INTRODUCTION sections, however, if it is not enough for reaching a decision, other parts like METHODOLOGY and CONCLUSION are considered.

### **Stage 4: Forward Snowballing**

To ensure no left potential primary studies, papers which might not have been reached by automatic searching or published in the predefined digital libraries were also searched. According to (Wohlin, 2014), forward snowballing process is realized by identifying other publications citing any of the primary studies. Google Scholar is used to find those papers. Consequently, newly found and selected papers are added to the final pool.

## **3.5 Inclusion & Exclusion Criteria (Selection Criteria)**

Once all potentially relevant papers are gathered, their relevance must be assessed. Selection criteria are intended for the purpose of identifying those papers (primary studies) directly related to the research questions as suggested in (Kitchenham and Charters, 2007). The inclusion and exclusion criteria must be based on the research questions. These criteria are applied when selecting the primary studies and when performing forward snowballing. To reduce the potentiality of a bias to occur, these criteria should be documented in the protocol definition stage. The selection criteria might be revised during the search process. Inclusion and exclusion criteria are applied to a paper/study by reading sections like title, abstract, introduction, and conclusion.

- a paper is included in the primary studies pool only if it meets all the inclusion criteria and none of the exclusion criteria.

### **3.5.1 Inclusion criteria**

**IC1:** Study must propose at least one of the MDE approaches or techniques for CPS.

**IC2:** Study must target CPS or at least one of its application domains.

**IC3:** Study must be peer-reviewed journal papers, workshop papers or conference papers.

**IC4:** Models presented by the study must not be used only for documentation and design purposes.

**IC5:** Paper publication period must be between 2010 and 2018.

**IC6:** Study must be available in full-text and published in a renowned digital library.

### **3.5.2 Exclusion criteria**

**EC1:** Study is a secondary study (survey, systematic mapping, systematic review, etc.).

**EC2:** Study is irrelevant to CPS or any of its application domains and the field of software engineering.

**EC3:** The study is a summarized version of a complete work already in the SLR pool.

**EC4:** Study is a kind of educational, editorial, tutorial, or other material (i.e., not a scientific paper).

**EC5:** Study was written in other languages than English.

### 3.6 Study quality assessment and self-assessment

Quality assessment (QA) and self-assessment (SA) questions similar to (Barišić et al., 2018) are formed. The QA questions (see Table 3.4), are defined to assess the quality of the studies. **QA-1** measures the degree of clarity to which the primary studies define the problem they are addressing. **QA-2** answers whether the primary studies reported the contributions of their study clearly. **QA-3** reports whether the study presents any future works or not. The results of the quality assessment questions are presented in Section 5.

SA question **SA-1**, see Table 3.4, is used to validate the understanding of the primary reviewer regarding the reviewed paper. If the primary reviewer gives a self-assessment score below 50% (i.e. not very confident about the paper) for a study, the secondary reviewer revises the extracted data from the study.

Table 3.4: Quality assessment and self-assessment

Quality Assessment		
	QA-1	What is the level of fineness in which the problem of the study defined?
	QA-2	To which extent the promises of the study are explicit?
	QA-3	Does the study present any future work(s)?
Self-assessment		
	SA-1	What is the reader's trust level regarding the paper?

### 3.7 Data extraction

Initially, the final pool of the primary studies is stored in Mendeley. Next, Google sheet is used for the data extraction stage. In the sheet, research questions are represented in columns, whereas, primary studies are presented in rows. The process of data extraction in this study goes through 3 phases. Data extraction form is shown in Table 3.5.

**Phase 1:** The primary reviewer starts extracting data from the primary studies (answering research questions). Extracted data for each study is represented in a row where each row has a key that refers to the study in

Mendeley. Data extraction of each paper is followed by answering quality and self-assessment questions.

**Phase 2:** The secondary reviewer starts reviewing primary studies with self-assessment score below 50%. After evaluating the study, if the secondary reviewer agrees with the answers given by the primary reviewer, the study is marked as agreed on, else, it goes through phase 3.

**Phase 3:** In this phase, primary and secondary reviewers discuss the paper disagreed upon in an effort to reach a common ground.

Table 3.5: Data extraction form

#	Study data	Description	RQ
1	Study ID	unique identifier for the study	-
2	Bibliometric & demographics	Authors' name, Title of the study, Year of publication, Authors affiliated country, number of citations	-
3	Source	IEEE Xplore, ACM, Scopus, Science Direct etc.	-
4	Article type	Conference, Journal, Workshop etc.	-
5	Modeling approach	used modeling approach(s) by the study	RQ 1.1
6	Modeling purpose	The purpose for which the study used models	RQ 1.2
7	MDE phase	The MDE phase (i.e. system design, simulation, transformation, V&V) the study addressed	RQ 1.3
8	Tools/Languages	used or developed tools/languages by the study	RQ 2
9	CPS component	The CPS component (i.e. physical component, cyber component, sensor, actuator) the study addressed	RQ 3
10	CPS application domain	The CPS application domain the study targeted	RQ 4
11	Type of evaluation	The type of evaluation (i.e. case study, use case, empirical study) the study presented	RQ 5
12	CPS challenges	The type of CPS challenge(s) the study addressed	RQ 6
13	Quality assessment	problem statement, contribution, future work	QA 1, QA 2, QA 3
14	Self-assessment	Reviewer's level of understanding of the reviewed paper	SA 1

## **4. Performing systematic review**

In this section, the process followed to conduct this systematic review is explained. Related process for performing the execution phase was applied by following the protocol defined in Section 3.

### **4.1 Performing automatic search**

Initially, digital libraries, indicated in Section 3, were manually searched. However, a massive number of studies (in average, above 5000 results from each search engine) were achieved. Therefore, it is decided to perform an automatic search as it is advised during conducting SLRs in (Kitchenham and Charters, 2007).

In order to perform an automatic search, search strings are to be developed. These search strings must fit the syntax of the targeted search engine. They should be “good-enough” to include as many relevant studies as possible, and concurrently, exclude irrelevant ones. Keyword and overall search string are defined using PICOC criteria as discussed in Section 3. Table 4.1 shows searched digital libraries and the corresponding search string(s) used. After concluding the automatic search, 646 studies were obtained.

Many challenges were encountered while using the digital libraries, one main challenge of using these digital libraries is the lack of guidelines explaining how to use the advanced search feature of these digital libraries. Also, the number of allowed terms of the search string is limited in digital libraries like ScienceDirect and DBLP, which lead to splitting the search string into multiple search strings. Also, wild cards are not supported in ScienceDirect. Another challenge is digital libraries like ACM and IEEE do not provide the capability to restrict the search to more than one specific area at once like title, abstract, and keywords combined.



Table 4.1: Search strings

Digital Library	Results	Search Query
IEEE	164	("model-driven development" OR "model-driven engineering" OR "model-driven architecture" OR "code generation" OR "generative approach" OR "model-based approach" OR "model-driven approach" OR "domain specific model*" OR "metamodel" OR "meta-model" OR "meta model" OR "modeling approach") AND ("cyber-physical system*" OR "cyber physical system*" OR "smart system*" OR "cyberphysical systems" OR "cps")
ACM	55	
Web of Science	16	
Scopus	363	See the online repository <sup>2</sup>
ScienceDirect	23	("code generation" OR "generative approach" OR "domain specific modelling" OR "modelling approach") AND ("cyber-physical systems" OR "cyber physical systems" OR "smart systems" OR cps OR "cyberphysical systems")
	12	("model-driven development" OR "model-driven engineering" OR "model-driven architecture" OR "model-based approach" OR "model-driven approach") AND ("cyber-physical systems" OR "cyber physical systems" OR "smart systems" OR "cyberphysical systems")
	9	(metamodel OR "meta-model" OR "meta model") AND ("cyber-physical systems" OR "cyber physical systems" OR "smart systems" OR "cyberphysical systems")
dblp	4	(metamodel   "meta-model"   "meta model") ("cyber-physical systems"   "cyber physical systems"   "smart systems"   "cyberphysical systems")
	0	("model-driven development"   "model-driven engineering"   "model-driven architecture"   "model-based approach"   "model-driven approach") ("cyber-physical systems"   "cyber physical systems"   "smart systems"   "cyberphysical systems")
	0	("code generation"   "generative approach"   "domain specific modelling"   "modelling approach") ("cyber-physical systems"   "cyber physical systems"   "smart systems"   cps   "cyberphysical systems")

#### 4.2 Removing duplicates

All 646 studies were stored in Mendeley. When Mendeley detects duplicate studies, one of them was manually removed. The process of duplication-

<sup>2</sup> <https://docs.google.com/spreadsheets/d/1YAX2mRoZE7Zchv1FCw2o9fCQMItFxPg2Jw2-4Vh1DiA/edit#gid=730724762>

checking goes until further stages (i.e., forward snowballing). The eliminated duplicate papers were 113 studies.

### **4.3 Selecting primary studies**

Selection of studies was based on the inclusion and exclusion criteria defined in Section 3. The process of selecting primary studies is shown in Figure 4.1. 533 studies were covered in this stage; the inclusion or exclusion of studies are performed in several iterations:

- Iteration 1: Primary reviewer went through each study reading its title, abstract, and checking the general content (figure, models, tables, etc.). Studies which meet the inclusion and exclusion criteria passed to the next iteration (278 studies were removed in this iteration).
- Iteration 2: All the studies which passed iteration 1 were read in more detail by further reading the related paper's introduction and conclusion sections and if necessary other sections (e.g. methodology and case study). This iteration resulted in including 88 papers and excluding 82 papers. 85 papers left undecided "to be reviewed".
- Iteration 3: the 85 undecided papers in iteration 2 were again reviewed with a secondary reviewer. In this stage, both reviewers agreed on either including or excluding the paper. As a result, 34 papers were later included, whereas 51 papers were later excluded.

To sum up, 88 papers were included from iteration 2 and 34 papers were included from iteration 3, forming a pool of 122 primary studies.

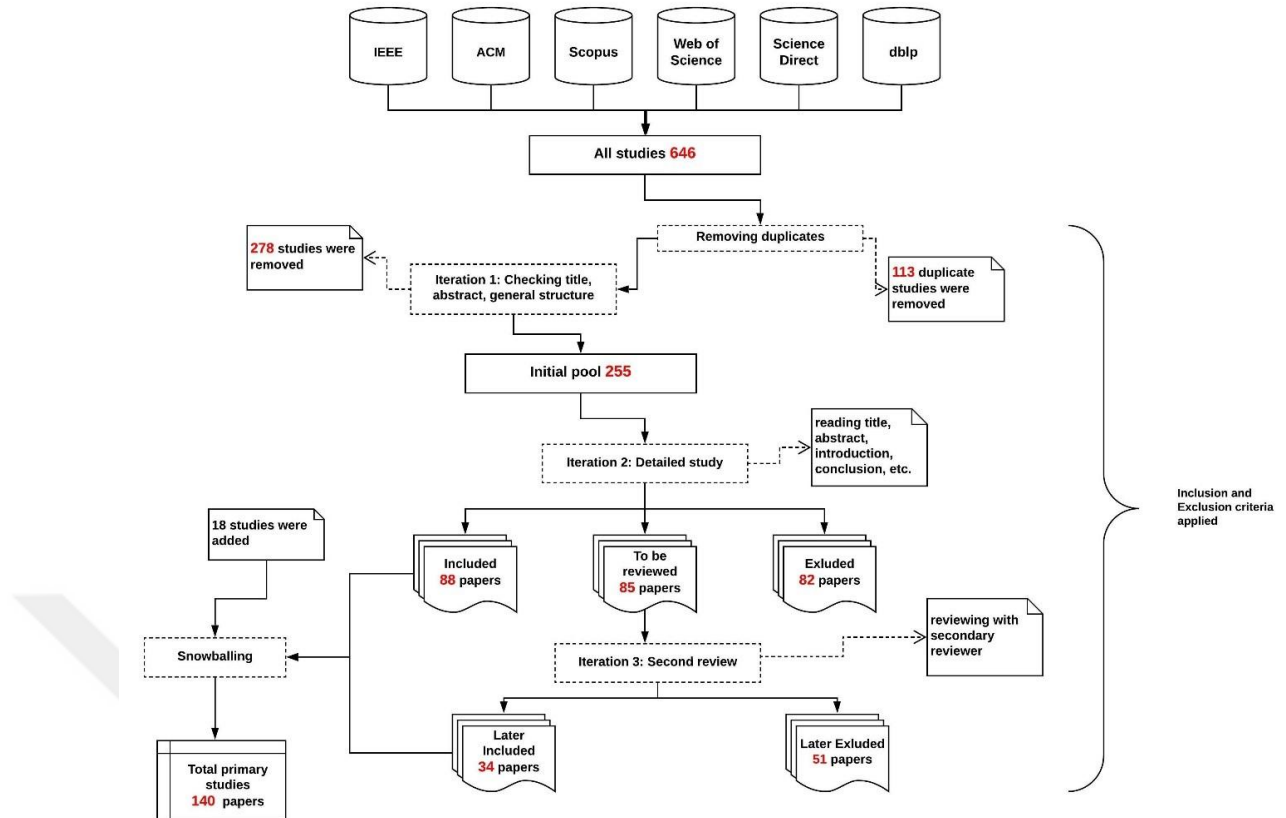


Figure 4.1: Search and selection process

#### 4.4 Forward Snowballing

After defining the primary studies, forward snowballing was performed during data extraction phase. The process of conducting forward snowballing is stated in Section 3. As a result, 18 papers were included to the pool of the primary studies, making a total of 140 papers considered in this study.

#### 4.5 Realizing Data extraction

In this stage, Google spreadsheets<sup>3</sup> were used for extracting data from the primary studies. Each study is given a unique key in order to match it with the original paper in Mendeley. The full text of each paper was read to answer the research questions and quality and self-assessment questions as well. A detailed qualitative and quantitative analysis was derived from the outcome of this section.

<sup>3</sup> <https://docs.google.com/spreadsheets/d/1K1vpUr5UKDz0KPKI-65TbP413XZpD6cKRcbDCJn-cXk/edit?usp=sharing>

Unlike in the selection stage, papers were read in a meticulous manner according to the protocol defined in this thesis.

#### **4.6 Data analysis**

Analyzing the extracted data from the primary studies was the last stage in the execution phase. As stated in section 3, data analysis will encompass both quantitative and qualitative analysis. The results of the analysis are presented in Section 5.



## 5. Results

In this section, the results and the findings of the SLR are presented. The section starts with bibliometrics and demographics analysis, followed by quality assessment analysis, and finally, analysis of research questions.

### 5.1 Bibliometrics & Demographics Analysis

#### 5.1.1 Publication trend per year

Basically, Figure 5.1 depicts the increase in the number of research papers on this topic. Between the years 2010-2018, researchers' interest in the domain of applying MDE for CPS had grown continuously for the period under observation.

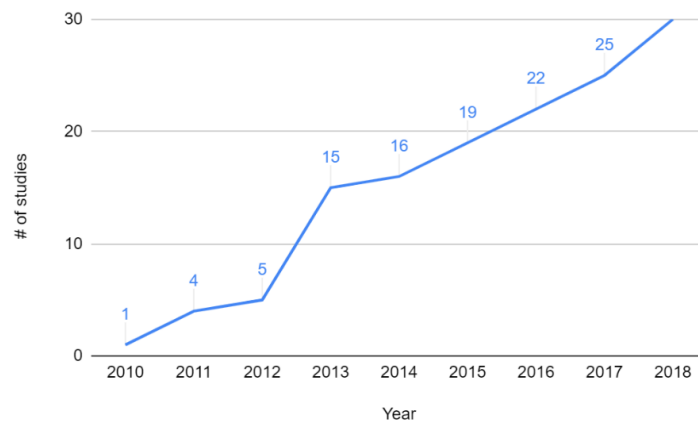


Figure 5.1: Publication trend per year.

#### 5.1.2 Citation analysis and top-cited studies

In this section, results related to the citation distribution over the year of publication is presented. The number of citations was obtained using Google Scholar. Figure 5.2.(a) shows distribution of citations over publication years, where Figure 5.2.(b) shows the median number of all papers' citations published in a given year. Only 15% of the primary studies are never cited. The 3 most cited papers are listed in (Table 5.1).

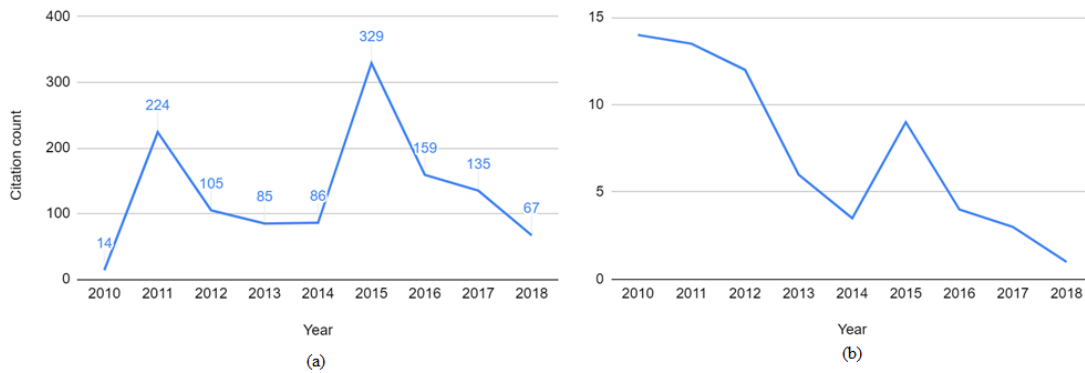


Figure 5.2: (a) Number of citation per year (b) Median number of citations

Table 5.1: Most cited papers

Study	Title	Year	Citations #
[26] <sup>4</sup>	Petri Net Modeling of Cyber-Physical Attacks on Smart Grid	2011	196
[130]	Cyber-Physical Modeling and Cyber-Contingency Assessment of Hierarchical Control Systems	2015	84
[111]	Modelling complex and flexible processes for smart cyber-physical environments	2015	67

### 5.1.3 Active researchers in the domain

To get an overview of the most active researchers in this domain, the number of papers published by each author are counted. To keep the brevity of the ranking results, Figure 5.3 shows only researchers who published at least three papers in the pool of primary studies. The authors “Lichen Zhang” and “Janos Sztipanovits” have the greatest number of publications, each with 6 papers. Followed by “Dehui Du” and “Jonathan Sprinkle” with 4 papers each. The complete list of authors can be found in the accompanying online repository<sup>5</sup>.

<sup>4</sup> The primary studies, included in this SLR, are referenced with these numbers in brackets throughout the thesis. The list of these primary studies is found in the Appendixes.

<sup>5</sup>[https://docs.google.com/spreadsheets/d/1VJhjimNVXU7\\_zxCri1Lf9WuhBD5phNWJYflNbzRtOGM/edit?usp=sharing](https://docs.google.com/spreadsheets/d/1VJhjimNVXU7_zxCri1Lf9WuhBD5phNWJYflNbzRtOGM/edit?usp=sharing)

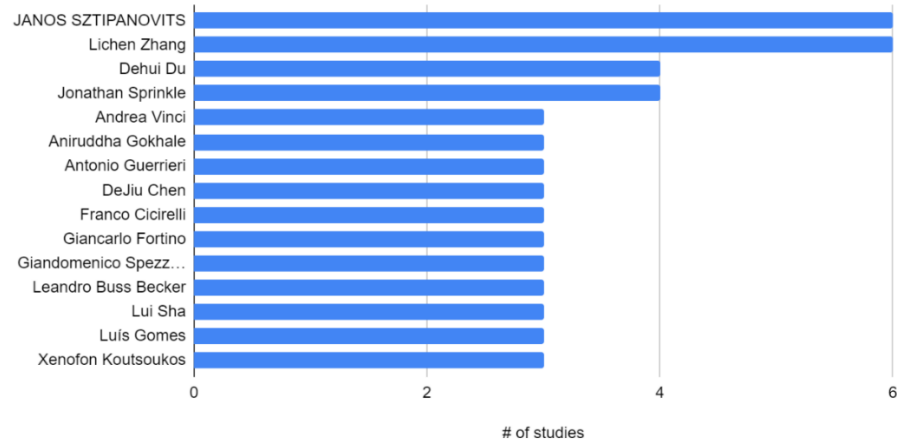


Figure 5.3: Authors with at least three papers.

#### 5.1.4 Countries contributing to the field (based on author affiliations)

Similar to (Garousi et al., 2013) that presents bibliometric studies in software engineering, most active countries are listed based on authors' affiliation, that is authors who have published papers in the field of applying MDE for CPS. If a researcher moved between two or more countries, each of his/her papers are assigned to the exact affiliation information on top of each paper. If a paper was written by researchers from more than one country, the counters for each of those countries were incremented by one.

Figure 5.4 shows the ranking of countries with at least two publications. The top 5 countries are; USA with 39 publications (25.16%), China with 23 publications (14.84%), Germany with 16 publications (10.32%), Italy with 13 publications (8.39%), and France with 12 publications (7.74%). According to the analysis, 112 (80%) of the papers were written by the author(s) affiliated to one country, while 28 papers (20%) were jointly written by authors from more than one country. In terms of internationally authored papers and the collaborating nations, the collaboration between China and USA is the highest [48, 60, 130], followed by Sweden and Italy [5, 110], and Tunisia and France [44, 45].

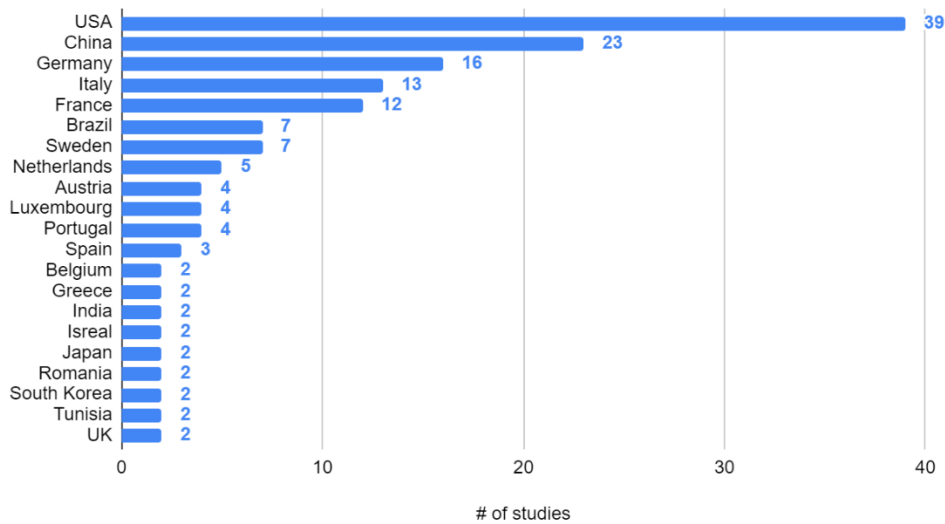


Figure 5.4: Countries contributing to the field (based on author affiliations).

### 5.1.5 Publication venues

90 of the studies (64.75%) were conference papers, while 33 (23.74%) and 14 (10.07%) studies were journals and workshop papers respectively. Table 5.2 shows the ranking of the top venues with at least two studies. The complete list of publication fora can be found in the online repository<sup>6</sup>. There are 16 venues in Table 5.2: 10 conferences/symposia, 4 journals, and 2 workshop. Interestingly, one can see that journals are at the bottom of the list with 2 publications each. That is, researchers in this field seem most likely preferring conferences than journals.

<sup>6</sup>[https://docs.google.com/spreadsheets/d/1VJhjimNVXU7\\_zxCri1Lf9WuhBD5phNWjYflNbzRtOGM/edit?usp=sharing](https://docs.google.com/spreadsheets/d/1VJhjimNVXU7_zxCri1Lf9WuhBD5phNWjYflNbzRtOGM/edit?usp=sharing)



Table 5.2: Venues with at least two papers

<b>Venue type</b>	<b>publication venue</b>	<b># of studies</b>
Conference	International Conference on Emerging Technologies and Factory Automation (ETFFA)	6
Workshop	Workshop on Domain-specific modeling	6
Conference	ACM/IEEE International Conference on Cyber-Physical Systems (ACM/IEE ICCPS)	4
Conference	Industrial Cyber-Physical Systems (ICPS)	3
Conference	International Conference on Engineering of Complex Computer Systems (ICECCS)	3
Conference	International Conference on Industrial Informatics (INDIN)	3
Conference	Annual Computer Software and Applications Conference (IEEE COMPSAC)	2
Conference	Brazilian Symposium on Computing Systems Engineering (SBESC)	2
Conference	International Conference on Networking, Sensing and Control (ICNSC)	2
Conference	International Systems Conference (SysCon)	2
Conference	ACM Symposium on Applied Computing (ACM SAC)	2
Journal	Advanced Engineering Informatics	2
Journal	IEEE Transactions on Smart Grid	2
Journal	IFAC Symposium on Information Control Problems in Manufacturing INCOM	2
Journal	International Journal of Critical Infrastructure Protection	2
Workshop	IFAC Workshop on Intelligent Manufacturing Systems	2

## 5.2 Quality assessment

When reading a research paper, the reader must be able to easily identify sections like: 1- the problem the paper address, 2- proposed contribution by the study, and 3- possible future work, as the clarity of these sections increase the quality and the overall understanding of the study. Therefore, it is meaningful to present statistics regarding how the primary studies stated these three sections. The results are given in Table 5.3.

In terms of the clarity of stating the problem which the study addresses, 74 studies (52.86%) clearly and precisely described the problem, while the problem description of the other 66 studies (47.14%) is partially obscure. Regarding stating the contribution, no contribution is stated by 26 studies (18.57%), 55 studies (39.29%) clearly stated their contribution, while the other 59 studies (42.14%) vaguely stated their contribution. Concerning the future work, 112 studies (80%) reported the possible future work, while the remaining 28 studies (20%) do not report any future work.

Table 5.3: Quality assessment results analysis

<b>Quality assessment</b>		
<b>Quality</b>	<b>%</b>	<b># of studies</b>
<b>QA-1: Problem statement</b>		
Not stated	0	0%
Partially stated	47.14%	66
Fully stated	52.86%	74
<b>QA-2: Contribution</b>		
Not presented	18.57%	26
Presented and Clear	39.29%	55
Presented but not clear	42.14%	59
<b>QA-3: Future work</b>		
No	20.00%	28
Yes	80.00%	112

### 5.3 Research questions Analysis

It is worth mentioning that some of the studies fit more than one group, that is, some papers reported more than one modeling approach, the purpose of modeling (modeling activity) and targeted MDE technique/phase. Therefore, in this work, each study is assured not to be limited to only one group, and instead assign it to every possible group reported.

#### 5.3.1 Modeling approaches employed for applying MDE in CPS

In this section, the results and findings for "**RQ1: Are any of MDE approaches or techniques used in/for the development of the studied cyber-physical system?**" and its sub-questions are presented.

**RQ1.1:** What is the modeling approach presented/used in the study?

As shown in (Figure 5.5), the most used approach is metamodeling. 15.86% of the primary studies (23 papers) reported metamodeling as the approach used in their studies. This is followed by model-based approach with 20 papers (13.79%), DSL with 18 papers (12.41%) and component-based approach with 15 papers (10.34%). Other used approaches include; State Machine based modeling, Model Driven Development, Signal-based Modeling, Models@run time, Agent-oriented modeling, Dynamic Constraint Feedback (DCF), Properties Modeling, Stochastic Occurrence Hybrid Automata (SOHA)-based modeling, Model-Integrated-Computing (MIC), Microservice-based development and Theory-based (e.g. modeling theory based on fuzzy logic).

Integrated approaches category comprises studies which promote either the integration of multiple approaches or multi-domain modeling approach. Studies employing integrated approaches are [42, 126]. On the other hand, studies which used multi-modeling approaches are [18, 65, 94, 135, 136, 138].

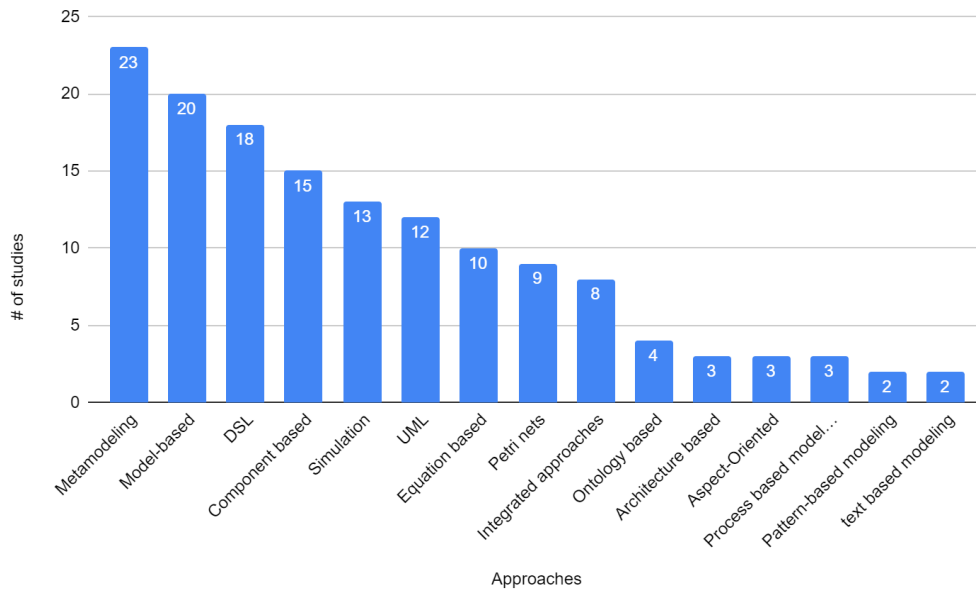


Figure 5.5: Reported modeling approaches.

Figure 5.6 shows the distribution of modeling approaches over the years. For better comprehension of the chart, the most used approaches reported by more than 5 studies are given only.

The most consistently used approach within the period of the study (2010-2018) was DSL except for 2010. This approach was at least reported by one paper between years 2011-2018. However, its growth fluctuates. Metamodeling and Model-based approach also showed a consistent presence between 2012-2018, while UML and Component-based approach were present continuously between 2013-2018. Although Metamodeling approach had minor reduction in its usage between the years 2012 and 2016, it always increased. For the years between 2015 and 2018, it is clearly observed that the Metamodeling approach was always amongst the top-most used 3 approaches.

For further understanding of the modeling approaches, the distribution of the most reported approaches over the countries is shown in Figure 5.7, which leads to find the followings:

- **Metamodeling:** is the most reported modeling approach with a total of 23 studies. 18 of those studies were written by authors affiliated to Europe, 2 by authors affiliated to Korea, 1 study by researchers affiliated to USA, 1

study was jointly written by authors from USA and Singapore, and 1 study was jointly written by authors from Malaysia, India, Europe(Austria).

- **Model-based approach:** reported by 20 studies, 6 studies are affiliated to USA, 1 study was jointly written by the authors from USA and Jordan, 7 studies are affiliated to Europe, and the rest of the studies are distributed amongst; Brazil (2 studies), China (1 study), Israel (1 study), Taiwan (1 study), and 1 study written jointly by New Zealand, Europe(Finland), and China.

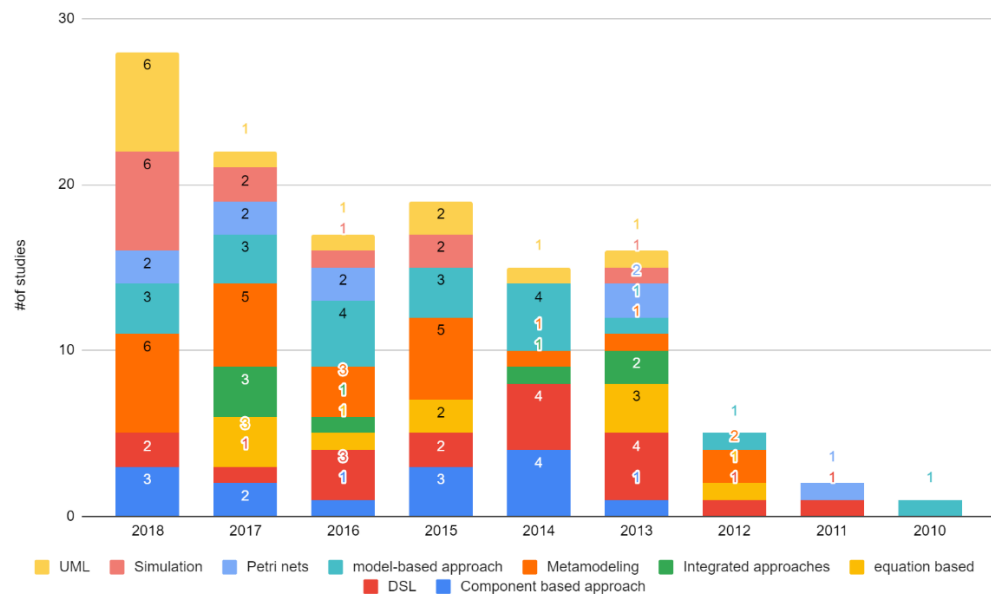


Figure 5.6: Distribution of the reported modeling approaches over the years

- **DSL:** Reported by 18 studies in total, 10 papers written by USA affiliated researchers, 1 study jointly written by USA and Pakistan researchers, 5 studies written by authors affiliated to Europe. Others include 1 study by KSA, and 1 study jointly written by China and Europe (France) affiliated researchers.
- **Component-based approach:** Among 15 studies following this approach, 9 studies were from Europe and 2 studies were from China. Others: 1 study by Brazil, 1 study by India, and 1 study jointly prepared by the researcher from Morocco and Europe (Latvia, Spain, Czech Republic, Netherlands, Italy, Romania).

- **Simulation:** This approach was reported by 13 studies in which 6 studies were written by authors affiliated to Europe, 3 from USA affiliated researchers and 1 study was written jointly by USA and China affiliated authors. Others: 1 study by Brazil, 1 study by Israel, and 1 study jointly written by authors affiliated to Israel and Japan.
- **UML:** Reported by 12 studies in total; 7 studies were written by authors affiliated to Europe, 3 studies by China affiliated researchers, 1 study by USA, and 1 study by Korea.
- **Equation-based modeling:** 10 studies used this approach. 3 studies were written by USA affiliated authors, 2 studies had joint authorship, one by USA and Pakistan and the other one by USA and China affiliated researchers. 2 studies are from Europe, 1 study from China, 1 study from Brazil, and 1 study from Iran.
- **Petri nets:** Used by 9 studies. 3 studies are from Europe affiliated researchers, 3 studies were written by researchers affiliated to China, 1 study was jointly written by China and USA affiliated authors. Others: 1 study by USA, and 1 study jointly written by USA and Europe (Spain, UK).

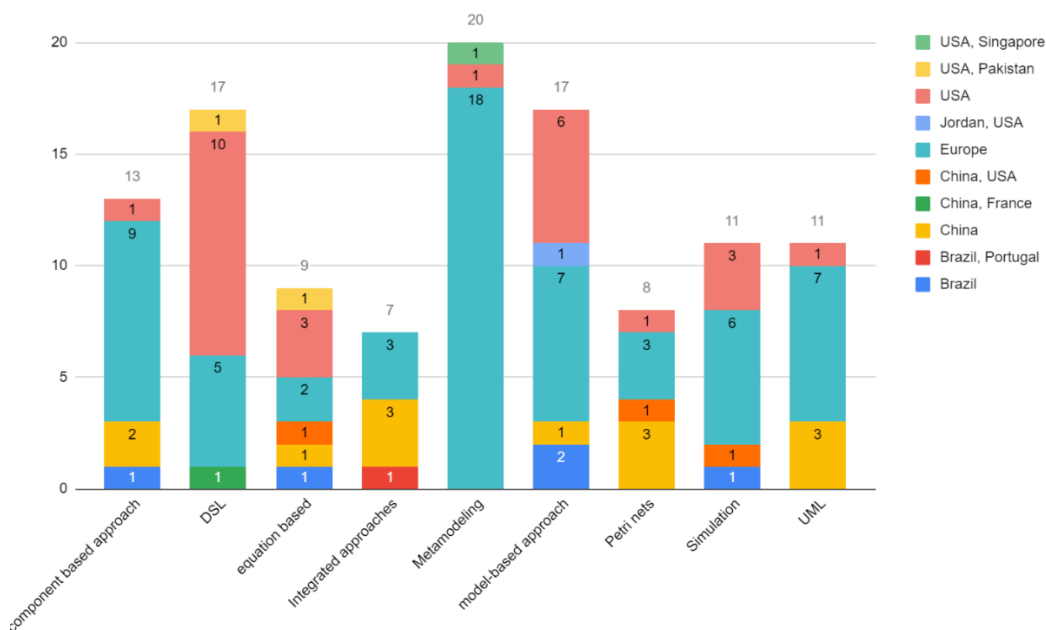


Figure 5.7: Reported modeling approaches vs. Authors' countries.

- **Integrated approaches:** This approach was reported by 8 studies. 3 studies are from Europe, 3 studies were written by authors affiliated to China, 1 paper jointly written by Brazil and Europe (Portugal) affiliated authors, and 1 study jointly written by authors affiliated to Japan and Thailand.

In summary, it can be seen that the Metamodeling and Model-based approaches are mostly used in Europe. On the other hand, DSL approach is mostly used in the USA and its usage surpasses all the European countries combined.

Further, it is important to mention that although equation-based approach is reported by 10 studies, it was used jointly with other approaches in 5 out of the 10 studies. [119] used equation-based modeling with DSL where they developed DSML for the performance analysis purpose. [114] also used DSL with equation-based approach to develop a DSML for simulation. [112] along with equation-based modeling used a simulation-based approach and used the Ptolemy II modeling tool and Simulink Design Verifier (SLDV) for Model-based Testing and formal verification. [105] used equation-based modeling with Petri nets based modeling approach. The study used discrete/continuous Petri nets for scheduling the analysis. [81] used Metamodeling based approach with equation-based modeling for the development of meta-models using Visual Environment for Cyber-Physical Modelling (VE-CPM). The remaining studies, which used equation-based modeling as their only approach, did not present any tool/language except [74] that presented a tool HA-SPIRAL for code generation. To this end, the equation-based modeling approach is somewhat useful as a supporting approach rather than as an independent approach in this field – applying MDE for CPS.

**RQ1.1.1:** What is the purpose for which the models were used?

Out of the 140 studies, 136 of them report their purpose for using the models, while 4 do not state their purpose. From the 136 studies, 111 papers reported only one activity, while 22 reported two activities, 2 reported three activities, and the last paper reported four activities. Figure 5.8 shows activities reported by at least 2 studies for better comprehension of the chart. Figure 5.9 represents the

distribution of modeling approaches over the modeling activities. All modeling activities are shown in Appendix 1 together with the approaches used and the studies reported them. Reported activities are as follows:

- **Development:** 37 papers (22.42%) are grouped under this category. These studies can be put into two categories: firstly, papers that developed DSL, Metamodel, tool, or language, secondly, studies that aim at automating the development process of a system, and perform tasks like transformation, code generation, building libraries, design process, and others. The most used approaches for this activity are Metamodeling and DSL, 8 papers used each of the two approaches. Model-based approach was used by 7 studies, while 3 papers reported Component-based approach. Further, Equation based approach, Integrated approaches, and Architecture-based approach reported by 2 studies each, while the rest of the approaches were reported by 1 study each.

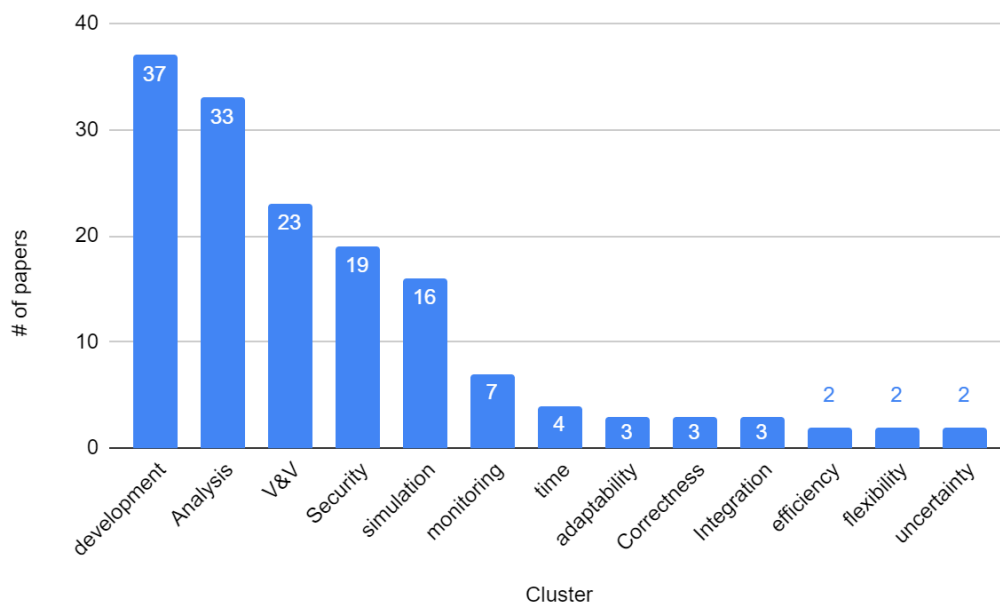


Figure 5.8: Reported activities/purposes for which modeling approaches used.

- **Analysis:** Reported by 33 studies (20%). Here, the aim of the studies is mainly focused on analyzing an existing system (DSL, metamodel, tool) for various activities. The most reported ones include: safety analysis, performance analysis, requirement analysis, security analysis, cost and



energy consumption analysis, dependability analysis, and so on. Metamodeling and Model-based approaches are the most reported approaches for this activity with 6 studies for each, followed by Petri nets with 5 studies, Integrated approaches and UML each reported by 4 studies, 3 studies each for DSL and equation based approach, 2 studies reported Simulation and Component based approach for each, while the rest of the approaches were reported by 1 study for each.

- **Validation and Verification:** 23 studies (13.94%), studies in this group conducted V&V activities regarding DSML validation, metamodel verification, behavior verification, verification of correctness, safety properties verification, model-based testing, formal verification and so on. Approaches used for this activity are distributed as follows: 5 studies reported DSL and model-based approach for each, followed by Simulation based approach with 3 studies. Equation-based, Component-based and Ontology-based approaches reported by 2 papers each. The rest of the approaches were reported by 1 study each.
- **Security:** 19 studies (11.52%) are concerned about the security of the system from different aspects. Studies reported about safety are also grouped in this set. Activities conducted by this group includes threat modeling, attack modeling, analyzing cyber-attacks, security evaluation and experimentation, safety guarantees of the generated code, and safe reconfiguration. The most used approaches for this activity are Metamodeling and Simulation reported by 3 studies each, followed by Model based approach, Component based approach, UML, Petri nets, Pattern-based modeling reported by 2 studies, while the rest of the approaches were reported by 1 study each.

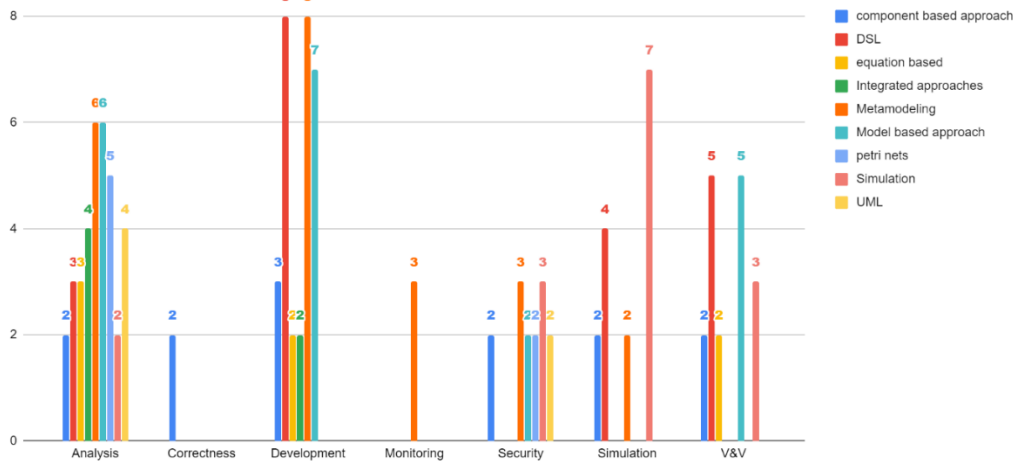


Figure 5.9: Modeling approaches distribution over modeling activities.

- **Simulation:** The aim of the studies in this group (16 studies (9.70%)) is the use of simulations for various purpose like using simulations for verification reasons or accompanying it with DSML, while other studies used it for analysis purpose. Mostly, studies reported simulation along with other activities like V&V, Analysis and Development. Obviously, the most used approach for this activity is Simulation based approach which is reported by 7 studies. It is followed by 4 studies reporting DSML, 2 studies for Metamodeling and 2 studies for Component based approach.
- **Monitoring:** 7 studies (4.24%) reported about system monitoring or management activities, such as performance monitoring, runtime behavior monitoring, process monitoring, monitoring simulation activities and results. The most reported approach in this group is Metamodeling with 3 studies. Other existing approaches were reported by 1 study each.
- **Time:** 4 studies (2.42%) seek to improve the time aspect of the system to increase productivity.
- **Adaptability:** 3 studies (1.82%) support the implementation of self-adaption aspect of the system.
- **Correctness:** 3 studies (1.82%) support the correctness of the system (DSML, metamodel, tool), often in terms of the correctness of operations or the generated code.

- **Integration:** 3 studies (1.82%) seek to combine different aspects of CPS and support their integration.

The rest of the activities are reported only by one study and it did not fit any of the aforementioned activities (as presented in Appendix 1).

For a deeper understanding of how studies addressed modeling approaches and the activities for which they were used, studies can be grouped into three categories:

- Studies which presented one modeling approach and used it for one modeling purpose, e.g. [39, 89, 140].
- Studies which presented one modeling approach and used it for more than one modeling purpose. Studies using the same approach for two different modeling purposes are [5, 24, 32] while studies using the same modeling approach for more than two modeling purposes are [60, 22].
- Studies which presented more than one modeling approach and used it for one modeling purpose/reason. For instance, [81] presented metamodeling and equation-based modeling approaches for the development purposes. [105] used petri nets and equation-based approach for the analysis reasons, [119] also used DSL and equation-based approaches for analysis activities, and finally [84] presented UML and pattern-based modeling approaches for security purposes.

**RQ1.2:** Which phase(s) of the system development is/are addressed in the study (using MDE)?

In this section, in addition to answering RQ1.2, a correlation analysis of RQ1.2 with RQ2 was carried out to find out the used or developed tools/languages in each of the MDE phases.

Figure 5.10 shows the reported MDE phases and their use frequencies. While it is possible to design and/or define MDE phases in numerous ways, we adopted the MDE phases defined in (Barišić et al., 2018) in our work since it also relates with the modeling of CPS. As can be seen in Figure 5.10, the studies differ in the number of the MDE phases they addressed. 72 studies addressed 1 phase, 46 studies reported 2 phases, 17 studies reported about 3 phases, and 5 studies

reported 4 phases. Discussion on all these MDE phases is given in the following where they are sorted from most reported to less reported in these studies.

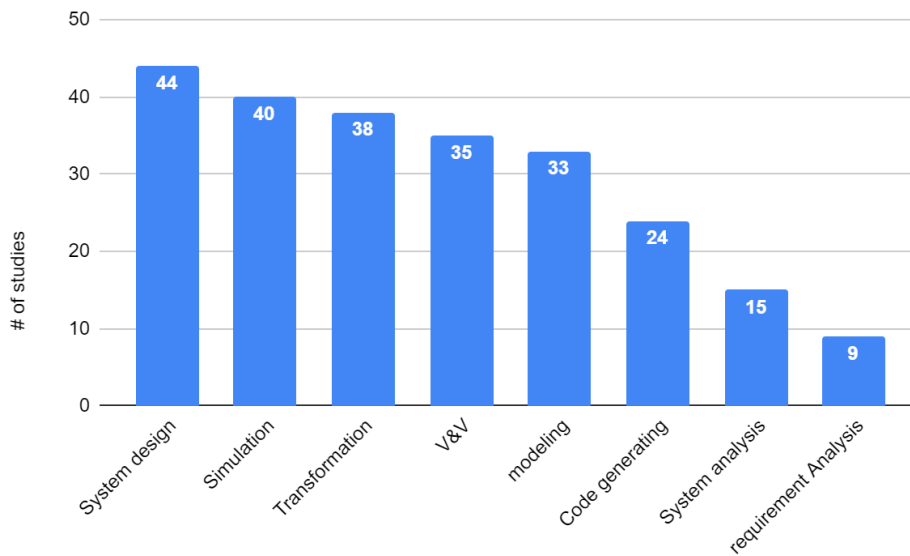


Figure 5.10: Reported MDE phases/activities.

#### - **System design**

System design is the most reported MDE phase, it is reported by 44 studies (18.41%). 15 studies presented DSL, 14 studies developed metamodel, 4 studies developed tools, 4 studies developed extensions for other tools/language, and the other 7 studies either develop a new modeling approach [43, 65], or combine MDE with existing approaches for CPS development [92]. Table 5.4 summarizes developed tools/languages and the corresponding papers.

Studies which developed DSLs are given as follows: [5] proposed a DSML called CyPhEF that supports the development and validation of self-adaptive CPS. [7] developed a simple graphical DSML for CPS while a DSML for irrigation networks was developed in [119]. In [86], authors developed a DSL that helps in quick construction of co-simulations for CPS, the grammar of the DSL was implemented in Xtext, while the code generation implementation was defined in Xtend. A framework called Advanced Vessel Simulation (AVS) was developed in [68] which supports design and evaluation of racing sailboat simulations. The AVS metamodel was developed in EMF, and Sirius was used for

developing the graphical editor. A textual DSL named CHARIOT was created with using Xtext in [104].

A DSL for managing different sensor configurations for a self-driving mini vehicle was developed in [77], the domain knowledge, static semantics, and the abstract syntax of a sensor management DSL were defined with the Eclipse Modeling Framework (EMF). [63] developed a DSML for the design of networked control systems (NCS) using passivity for separating the NCS control design from uncertainties (i.e. time delays and packet loss). In [118], authors used an ecore-based meta-model to define the abstract syntax of the proposed DSML, and the concrete syntax was implemented as an extension of Simulink standard blocks.

[2] developed two meta-models for representing and sharing incident knowledge of CPS. Meta-models were developed as Eclipse plugins. A metamodel for a systematic analysis of CPS threat modeling was developed in [78] using MetaGME, while [76] developed a metamodel using ADOxx and UML and they used it for the description of an end-to-end communication use case. A meta-model for the development of a smart cyber-physical environment was presented in [31].

[15] developed a meta-model for flexibility and dynamic reconfiguration of the automated production systems by using Eclipse Modeling Framework (EMF). [120] used UML profile to develop a meta-model for modeling cyber-physical assembly systems. Also, in [32], UML class-diagrams were used to develop a meta-model called Smart Environment Metamodel (SEM) to design smart cyber-physical environments. In [56], the authors extended some metamodels of SysML/MARTE for capturing the characteristics of CPS like continuous behavior and stochastic behavior. The approach was implemented in GEMOC. They defined the abstract syntax using EMF, graphical concrete syntax in Sirius, and the textual concrete syntax using Xtext. Meta-models conforming to ISA95 and ISA88 standards were developed in [57] for monitoring the process of an oil production industry.

In [126], they used ADOxx to develop the modeling tool Cyber-Physical Systems for Industry (CPS4I) for the connection of CPS and conceptualizations of

industrial applications in integrated models. A tool, called FTOS, based on openArchitectureWare8 (oAW) was developed in [19], which provides code generation for designing fault-tolerant automation systems. In [44], the authors presented BPMN4CPS which is an extension of BPMN 2.0 for handling CPS features. New extensions for MechatronicUML were developed in [53].

Table 5.4: Developed DSLs, Meta-models, tools, and extensions

<b>System Design</b>			
<b>Type</b>	<b>Used framework/language</b>	<b>Relevant studies</b>	
DSML	EMF	[5, 77, 68, 118]	
	GME	[119, 132, 63, 10, 133, 4]	
	Xtext	[86, 104]	
	Papyrus tool	[7]	
	Xtend	[86]	
	GReAT	[10, 133]	
	SIMULINK	[118]	
	WebGME	[20]	
	GEMOC	[47]	
	Sirius	[47, 68]	
	Kermeta	[47]	
	Meta-model	EMF	[15, 56, 111]
		UML	[32, 120, 30, 76, 31]
GEMOC		[56]	
Xtext		[56]	
Sirius		[56]	
Kevoree Modeling Framework (KMF)		[51]	
GME (MetaGME)		[78]	
ADOxx		[76]	
SysML		[11]	
ISA95 and ISA88 based		[57]	
-		[2]	
-	[66]		
Tool	MetaEdit+	[23]	
	ADOxx	[126]	
	openArchitectureWare (oAW)	[19]	

	-	[59]
Extension	-	[14, 43, 44, 53]

### - **Simulation**

40 of the studies (16.74%) reported simulation. 11 studies addressed exactly the simulation process. They can be summarized as follows: Only 1 study [122] developed a simulator. 2 studies developed meta-models [29,80], and 8 studies [12, 50, 54, 79, 82, 89, 106, 124] used existing tools for modeling and simulation. Remaining 29 studies incorporating simulation addressed the other phases (i.e. system design, transformation, V&V, etc.).

Table 5.5 shows the tools and languages used for the simulation activities. Studies presented different reasons for using simulation, e.g. [5] presented simulation as a feature of the developed DSL and used it for efficiency and time analysis via MECSYCO co-simulation engine. Also [119] presented the simulation as a feature of the developed DSL and used it for performance analysis via MATLAB and EPANET. In [86], authors developed a DSL for constructing HLA-based co-simulations. [63] used Simulink for time and network delay analysis. [10] benefited from Robocode simulator to simulate a reconfigurable conveyor system's behavior and run it in the background (used it as a background simulation) to output time information and the coordinate for the generation of Java animation. Similar to [63], [35] used Simulink for time performance analysis. [37] is another example for utilizing simulations for analysis purposes in which CPS Safety Analysis and simulation Platform (CP-SAP) was developed. Simulations were also used for security experimentation purposes like in [50, 106, 131].

Table 5.5: Reported simulation tools and languages

<b>Simulation</b>		
<b>Existing tool</b>	<b>Developed tool</b>	<b>Relevant studies</b>
Simulink/Stateflow	-	[63, 79, 35, 95, 60, 4, 125, 124, 131, 96, 106]
MATLAB	-	[119]
Modelica	-	[114]
ModelicaML	-	[137]
C2WT		[12, 89]
Ptolemy-II	-	[97, 130, 37]
EPANET	-	[119, 54]
-	Meta-model	[80]
MECSYCO	-	[5]
POOSL	-	[86]
Robocode	-	[10]
Embedded Systems Modeling Language (ESMoL)	-	[90]
IOPT-Flow simulator	-	[99]
SystemC	-	[127]
CPGAME	-	[82]
Verilog-AMS	-	[7]
-	Smart Grid Simulator (SGS)	[122]
-	Meta-model	[29]
CPAL	-	[88]
DEVS-Suite Simulator	-	[3]
JSBSim	-	[50]
ScicosLab	-	[50]
CIF	-	[107]



## - Transformation

38 studies (15.90%) presented transformations (listed in Table 5.6). 30 studies covered one transformation type (either M2M or M2T), 3 studies considered two transformation types, while 1 study [40] showed 3 different transformation types namely M2M, M2T, T2M. the transformation types presented by the other 4 studies was not clarified. Therefore, 39 transformations were presented in total and they are as follow: 28 M2M transformations, 10 M2T transformations, and one T2M transformation. Studies implemented M2M transformations can be categorized into two:

First category covers the studies using existing tools and languages. [49] used Y2U tool to transform Statechart models to UPPAAL timed automata model. [41] presented M2M transformation by transforming Simulink simulation models to AADL architectural models using Assisted Transformation of Models engine. AADL and Modelica were used in [136, 138], where both Modelica and AADL were transformed to each other. In [36], authors used Critical Infrastructure Protection - Vulnerability Analysis and Modeling (CIP VAM) UML profile to transform UML models to Bayesian Network (BN) models. In [92], the authors transformed UML models to Distributed Embedded Real-time Compact Specification (DERCS) models with using GenERTiCA. In [8], the authors transformed a SysML model to a graph by employing GraphML. Other studies include: [35, 40, 84] used QVT, [34] implemented M2M transformation using Xtext, [10, 129, 133] used Graph Rewriting and Transformation (GReAT) for M2M transformation, while EXTEND is used for the M2M transformation in [19].

Studies which developed metamodel, tool, or language for the M2M, M2T, and T2M transformations, constitute the second category. In [116], they proposed a transformation method that transforms Simulink model to ECML model by designing metamodels for both Simulink and ECML. [95] developed the model translation tool UPP2SF that transforms UPPAAL timed automata models to Simulink/Stateflow. On the contrary, [60] developed a tool named STU that

translates Simulink/Stateflow model into UPPAAL timed automata model. In [42], the authors developed a tool named ECPS Verifier that was used for the transformation of AADL models to UPPAL timed automata. In [98], they presented a tool named Simulink/AADL Translator Tool (AS2T) that automates the transformation of the simulation models of Simulink to AADL models. [28] presented a framework called Modena that helps transforming SysML and MARTE models into Reactive Modules Language (RML) and Modelica models.

The studies bellow presented the implementations of M2T transformations using existing tools like Acceleo [35, 77], Xtend [34, 40], IOPT tools [17], and GenERTiCA [92], except for one study [140] that presented meta-models of HybridUML and Quantified Hybrid Program (QHP) and then they used ATLAS Transformation Language (ATL) for defining the transformation rules. [40] was the only study presenting T2M transformation using Xtext.

Table 5.6: Reported model transformations, and used transformation tools and languages

<b>Transformation tools</b>			
<b>Transformation Type</b>	<b>Existing tool</b>	<b>Developed tool</b>	<b>Relevant studies</b>
M2M	-	metamodel	[116]
	Y2U tool	-	[49]
	AST Engine	-	[41]
	QVT	-	[40, 35, 84]
	AADL,Modelica	-	[136, 138]
	CIP VAM UML	-	[36]
	EXTEND	-	[19]
	GreAT	-	[10, 133, 129]
	UML4IoT	-	[120]
	-	UPP2SF	[95]
	ME+ tool	-	[24]
	-	STU	[60]
	GenERTiCA	-	[92]
	-	ECPS Verifier	[42]
	Xtext	-	[34]
	GraphML	-	[8]

	-	AS2T	[98]
	-	Modana	[28]
	Other		[91, 72, 118, 58, 25]
M2T	Acceleo	-	[77, 35]
	Xtend	-	[40, 34]
		metamodel	[140]
	IOPT tools	-	[17]
	GenERTiCA	-	[92]
	Other		[15, 127, 43]
T2M	Xtext	-	[40]
Other			[123, 3, 45]

#### - Validation and Verification (V&V)

35 studies (14.64%) reported V&V activity. Only 2 studies developed a tool [45, 127], and 1 study developed an ontology [75]. Table 5.7 presents studies which implemented V&V as part of their work.

Studies using UPPAAL for verification include the followings: [49] used UPPAAL to formally verify the safety properties of medical guideline. A Domain-specific model checking (DSMC) for MECHATRONICUML using UPPAAL model checker was presented in [40]. A pacemaker was modeled and verified using UPPAAL in [95]. In [60], the authors used the UPPAAL tool for the verification of SIMULINK/STATEFLOW models after being transformed to UPPAAL timed automata. In [42], UPPAAL was used for the formal verification (i.e. model checking) of AADL models.

Other tools and languages used for model checking for verification include; Simple Promela Interpreter (SPIN) model checker was used in [133] to verify the Promela code. Also, in [52], SPIN was used as a model checker to verify the PrT net models after translating it to a Promela code. In [94], they used a probabilistic model checker called PRISM. The authors in [103] verified their protocols via timed model checking MECHATRONICUML.

Simulink/Stateflow was used in [125] to verify supervisory controllers for hierarchical systems. Simulink Design Verifier (SLDV) was used for the

verification of the simulation models in [112]. In [85], they used SLDV to verify the behavioral models developed in Simulink in order to meet the requirements modeled. Furthermore, Object Constraint Language (OCL) was used for the verification of the static semantics of a meta-model presented in [77]. Also, in [5], OCL was used for defining and validating metamodel constraints. In [140], a verification of KeYmaera-QHP code in KeYmaera, a hybrid verification tool, was presented. In [90], the authors used FORMULA for metamodel analysis and verification. Frama-C was used in [33] to prove and verify a developed C code library. In [85], Assume Guarantee REasoning Environment (AGREE) tool was utilized to verify that the AADL architectural models satisfy the system requirements.

Studies, presenting validation, are summarized as follows: [126] developed a modeling tool (CPS4I) and a modeling method (SeRoIn) then validated them using Open Models Laboratory (OMiLAB). CHECK validation language was used in [19] to formulate tests for the detection of semantic design errors in the developed models. A generated code in [9] was tested and analyzed using Frama-C. In [107], the authors implemented simulation-based validation.

A tool, named Simulation and Verification of Hierarchical Embedded Systems (SHARC), was developed in [127] for the verification of the behavior of automotive safety-critical systems. In [75], the authors developed an ontology and used it as the validation mechanism.

Table 5.7: Reported V&amp;V and the used tools

<b>V&amp;V</b>		
<b>Existing Tool/Language</b>	<b>Developed Tool/Language</b>	<b>Relevant studies</b>
UPPAAL	-	[49, 40, 48, 95, 60, 42]
Simulink Design Verifier (SLDV)	-	[112, 128, 85]
SPIN (Simple Promela Interpreter)	-	[133, 52]
Object Constraint Language (OCL)	-	[77, 5]
Clock Constraint Specification Language (CCSL)	-	[100]
Open Models Laboratory (OMiLAB)	-	[126]
KeYmaera	-	[140]
CHECK	-	[19]
-	Ontology	[75]
Linux Driver Verification tool (LDV)	-	[62]
FORMULA	-	[90]
EAST-ADL	-	[23]
-	SHARC	[127]
Web Generic Modeling Environment (WebGME)	-	[20]
Protégé	-	[109]
Simulink	-	[125]
PIPE+	-	[52]
Frama-C	-	[33]
PRISM	-	[94]
MECHATRONICUML	-	[103]
AGREE	-	[85]
-	BPMN4CPS Tool	[45]

## - **Modeling**

33 studies (13.81%) reported about modeling. This category encompasses studies which used existing languages/tools for modeling, wherein studies which developed a language/tool for modeling were included in the group of system design. Only 4 studies [27, 62, 93, 130] did not report any tool, instead, they either proposed an approach for modeling CPSs or used equational models. Table 5.8 shows various languages/tools used for modeling by the studies.

In [112, 113], authors used Ptolemy II to model Medical CPS, while in [11] the behavioral model of production nominal resource was modeled using Ptolemy II. In [97], authors modeled a Holter Monitor. They used Ptolemy II to model the device's functionality and UPPAAL for modeling system's state space and the transitions between them. A modeling approach called time-constrained aspect-oriented Petri net was presented in [105]. The approach combines discrete/continuous Petri nets and aspect-orientation for modeling CPS. In the work presented in [72], colored Petri nets were extended to probabilistic colored Petri nets for modeling and analyzing CPS attacks. Petri nets were also used in [26] for modeling smart grid threats.

View oriented approach was adopted in [136] for the description of different aspects of an aerospace CPS. Modelica was used for modeling the overall architecture of a lunar rover robot and the lunar rover robot's body structure model while AADL was used for modeling the navigation system of the lunar rover. Similarly, authors in [135] integrated AADL, UML and Modelica to model the requirements of a vehicular ad-hoc network. Further, in [137], AADL and Modelicaml were integrated to model big data-driven CPS. In [85], Simulink/Stateflow was adopted to model a generic patient-controlled analgesia infusion pump system for analyzing logical requirements and behaviors, while AADL was used for developing the architectural model of the system.

Yakindu statechart tools was adopted in [49] to model and simulate a stroke statechart model. Likewise, in [48], they used Yakindu statecharts for the modeling of a simplified cardiac arrest. UML statechart model was modeled in [39] using Yakindu statechart. In [84], Papyrus tool was used for creating the

UML models. The authors in [61] presented a methodology for knowledge representation of CPS using the modeling tool Papyrus. [109] used UML for defining the dependability analysis models. Implementations of modeling CPS using HybridUML was presented in [140].

Finite state machine (FSM) was adopted in [16] to model the behavior of automation components. [88] used FSM to describe the logic of a servo tester. GME was used in [70] to build Lathe CNC System models and export models' data as an XML file. ASLan++ was used in [108] for modeling water treatment plant and attack model. [27] presented a new formalism named Stochastic Occurrence Hybrid Automata and a modeling approach to model the stochastic behavior in CPSs.

Table 5.8: Reported tools/languages used for modeling and the studies used them.

<b>Modeling</b>	
<b>Tool/Language</b>	<b>Relevant studies</b>
Ptolemy II	[112, 97, 11, 113]
Petri nets	[52, 26, 105, 72]
AADL	[136, 135, 137, 85]
YAKINDU Statechart Tools	[49, 48, 39]
Papyrus tool	[35, 61, 84]
UML	[135, 109]
Modelica	[135, 137]
Simulink	[128, 85]
Finite State Machine (FSM)	[16, 88]
SysML	[35, 56]
HybridUML	[140]
MARTE	[56]
UPPAAL	[97]
GME	[70]
MetaEdit+	[24]
web ontology language (OWL)	[101]
ADVISE Meta tool	[21]

### - **Code generation**

24 studies (10.04%) reported about language/tools used or developed for code generation purposes. Table 5.9 lists used and developed languages/tools for code generation. These studies are categorized here into studies which used existing tools for code generation, and studies which developed tools for code generation.

In [95], they used Simulink Real-Time Workshop Embedded Coder (RTWEC) to generate C code from a pacemaker Stateflow chart. Likewise, C code and VHDL code were generated in [60, 85] from the Stateflow models using Simulink coder. Moreover, a tool named GeneAuto was presented in [33] that generates C or ADA code from Simulink models. In [90], built-in code generator for Embedded Systems Modeling Language (ESMoL) was used to generate functional C code. IOPT-Flow tool framework was used in [99] to generate C and Javascript code or VHDL hardware descriptions.

Programmable Logic Controllers (PLCs) Code Generation was presented in [46] using Scenario Modeling Language (SML). In the same manner, implementation of PLC code generation was presented in [107] using Compositional Interchange Format (CIF). Clock Constraint Specification Language (CCSL) constraints were utilized in [100] for code generation purposes. The authors in [86] used Xtend for the support of code generation for OpenRTI. Kevoree Modeling Framework (KMF) was used in [51] for the generation of Java API in order to create and manipulate the runtime models.

In [6], they developed a tool named I2C4IOPT for automatic code generation of globally asynchronous and locally synchronous systems (GALS) - supported by Arduino boards. An ISA88 editor was implemented in EMF in [15] to generate a programmable logic controllers (PLC) control code. A code generator was developed in [63] for the generation of Simulink models and network-scripts. In [9], a model-based code generator for medical CPS was presented. An interpreter was developed in [133] to translate finite-state machine (FSM) models and constraints into Promela code.



Table 5.9: Reported code generation languages/tools and studies used them

<b>Code generation</b>		
<b>Existing Tool/Language</b>	<b>Developed Tool/Language</b>	<b>Relevant studies</b>
Simulink coder	-	[95, 60, 85]
-	I2C4IOPT	[6]
Acceleo	-	[68]
Clock Constraint Specification Language (CCSL)	-	[100]
Xtend	-	[86]
-	code generator	[63, 9, 70]
XPand	-	[19]
Scenario Modeling Language (SML)	-	[46]
-	ISA88 editor	[15]
Embedded Systems Modeling Language (ESMoL)	-	[90]
IOPT-Flow tool framework	-	[99]
-	interpreter	[133]
Generation of Embedded Real-Time Code based on Aspects (GenERTiCA)	-	[92]
KMF	-	[51]
Rhapsody code generator	-	[96]
HA-SPIRAL	-	[74]
Compositional Interchange Format (CIF)	-	[107]
GeneAuto	-	[33]

### - System Analysis

15 studies (6.28%) reported language/tools which used or was developed for system analysis reasons. Table 5.10 lists the used and developed languages/tools for system analysis. Studies can be categorized into ones directly using existing tools for system analysis, and others developing new tools for system analysis purposes.

In [139], meta-models for operational analysis and system analysis were developed. They also used TTool for safety analysis. A knowledge-based approach using Failure Models, Effects and Criticality Analysis (FMECA) techniques was presented in [1]. The authors first modeled FMECA using UML

class diagram, then the FMECA metamodel was expressed in Protégé, which was then used to build an ontology-based KB. A metamodel was developed to enable the management of application requirements and business constraints for CPS in [110]. CPS meta-model for knowledge formalization was presented in [69], where they also implemented formal concept analysis in their work. CPS Safety Analysis and Simulation Platform (CP-SAP) was developed in [37] for Human-machine interaction (HMI) safety analysis of CPS. A framework called Modena was presented in [28] that aims to model and analyze the non-functional aspect (i.e. time, energy, etc) of Energy-Aware CPS.

In [105], a modeling approach based on discrete/continuous Petri nets was proposed for schedulability analysis. Also, a modeling approach was presented in [72] that supports both qualitative and quantitative analysis of CPS attacks using probabilistic colored Petri nets. CPS dependability analysis was presented in [55] using Stochastic Petri Net (SPN). An approach for specification and analysis of automotive CPS was presented in [135] where Modelica was used for analyzing engine model and AADL was used for End to End Delay Analysis from brake-pedal to throttle actuator. Security analysis tool CL-AtSe was used in [108] for analyzing and discovering potential attacks on Industrial Control Systems.

Table 5.10: Reported System analysis languages/tools

<b>System Analysis</b>		
<b>Existing Tool/Language</b>	<b>Developed Tool/Language</b>	<b>Relevant studies</b>
EAST-ADL	-	[22, 24]
Petri nets	-	[105, 72, 55]
AADL	-	[135]
Modelica	-	[135]
-	Metamodel	[110, 83, 139, 69, 1]
-	CP-SAP	[37]
-	Modana	[28]
CL-AtSe	-	[108]
TTool	-	[139]
HiP-HOPS tool	-	[24]

### 5.3.2 Developed/used MDE tools for CPS

In this section, the results and findings of "**RQ2: Is/Are there any tool(s) used/developed to apply MDE in/for cyber-physical systems in the study?**" and its sub-questions are presented.

Out of the 140 studies, 13 studies did not present or develop any tool/language. The other 127 studies are as follows:

- 68 studies used existing tools/languages for modeling CPS.
- 59 studies developed DSLs, Metamodels, tools in addition to using existing tool/languages.

**RQ2.1:** Which tool(s) is/are presented/used in each phase of the system development?

Figure 5.11 shows tools and languages used by the primary studies. For better understanding of the tools/languages, and to give the reader a clear idea about the MDE phase/activity the tool/language was used for, a correlation analysis between RQ1.2 and RQ2 is presented in section 5.3.1. Therefore, in this section, the most used tool/languages are briefly discussed.

The study found that Simulink is the most used tool. Majority of the reviewed studies used Simulink for simulation purposes, listed in (Table 5.5). Simulink was also used for modeling [85, 128]. Simulink coder is used for code generation purposes in [60, 85, 95]. Simulink Design Verifier (SLDV) was adopted for the verification of models [85, 112, 128]. AADL follows Simulink as the most used tool. It was used for modeling the cyber component of the system [136], developing architectural models [85], or for system analysis [135].

UML is used by various studies for building metamodels, listed in Table 5.4, it was also used for modeling activities like defining dependability analysis models [109]. The vast majority of the studies used UPPAAL for verification, see Table 5.7. For instance, [97] used UPPAAL for modeling system's state space and the transitions between them.

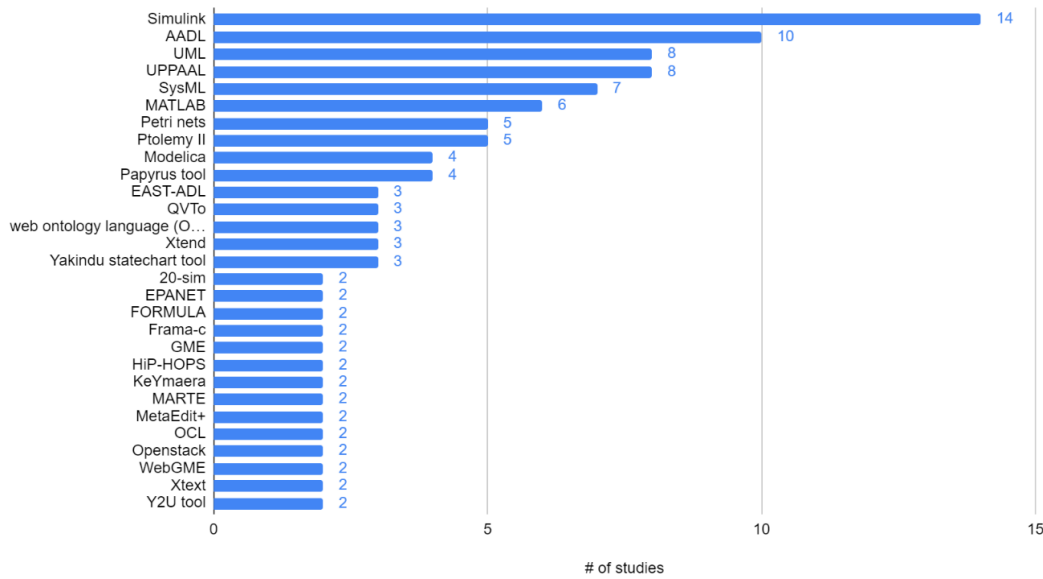


Figure 5.11: Most used tools/languages which are at least reported by 2 studies.

**RQ2.2:** If any tool(s) is/are developed in the study, is/are this/these tool(s) reported?

As mentioned earlier, 59 studies developed DSL/DSML, metamodel, tool, or extension for other tools/languages. 22 of these studies developed a metamodel, 15 studies developed DSL/DSML, 18 studies developed a tool (including 2 frameworks and 1 platform), and 4 studies presented extensions for other tools/languages as shown in Figure 5.12. These developed tools/languages were addressed in a detailed way in the correlation analysis done between RQ1.2 and RQ2 which is presented in section 5.3.1.

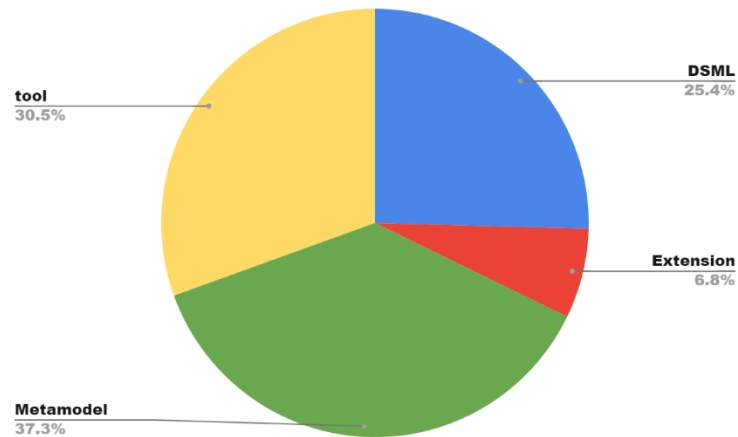


Figure 5.12: Frequency and type of the developed tool/language.

**RQ2.2.1:** Is/are the developed tool(s) available and/or accessible?

From the abovementioned 59 studies which consider the development of a tool/language, only 10 studies [2, 5, 6, 41, 42, 45, 56, 60, 127, 133] provided public access (mostly with a web link) to the developed tool/language.

**RQ2.2.2:** What is/are the framework(s) or programming language(s) for the development of this/these tools?

As indicated in Figure 5.13, UML is the most used language, followed by EMF and GME. Figure 5.13 shows the correlation between RQ2.2 and RQ2.2.2. It is clear that UML is mostly used for building metamodels, where EMF is used for building both metamodels and DSLs, and GME is mostly used for building DSLs. Other presented tools are; GreAT that is used alongside with GME [10, 133], Sirius used for building the graphical concrete syntax [47, 56, 68] and finally, Xtext used for developing DSL grammar in [86], and for building the textual concrete syntax as in [56]. These tools/languages are shown in Appendix 2 and discussed in detail in section 5.3.1.

GME seems to be the third most used language for building DSLs and metamodels. However, it is worth mentioning that the results of distributing RQ2.2.2 over the publication years (see Figure 5.14) show that GME is not used for the last 2 years (2017 and 2018) of the examined period by any of the primary studies. Further, the results of RQ2.2.2 were distributed over authors' country of affiliation as depicted in Figure 5.15. The study found out that GME and its tool

GReAT were only used by authors/researchers affiliated to the USA, where on the other hand, UML and EMF were mostly used by authors/researchers affiliated to Europe.

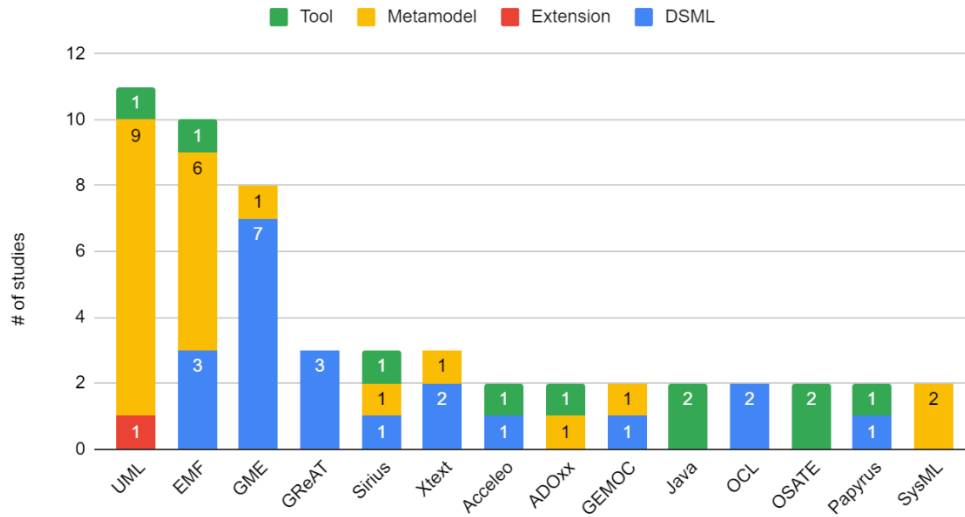


Figure 5.13: Used languages/frameworks for developing DSMLs, Metamodels, or tools

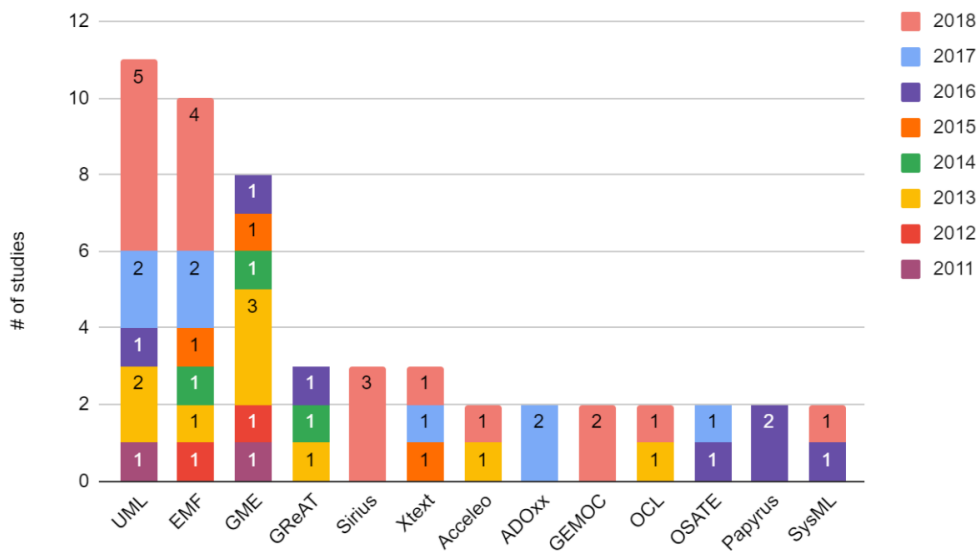


Figure 5.14: Distribution of languages/frameworks used for developing DSMLs, metamodels, tools over publication years.

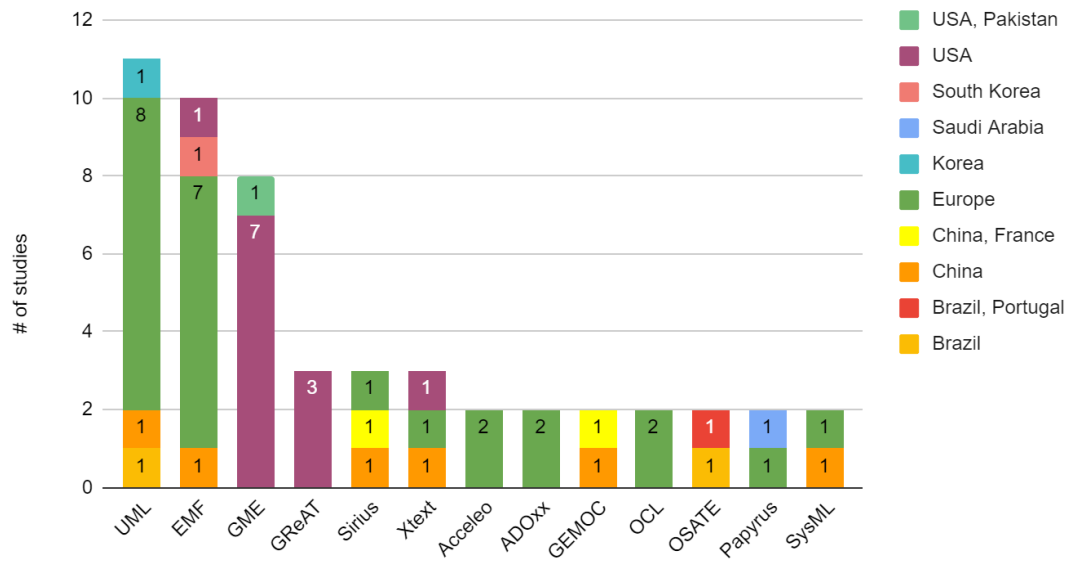


Figure 5.15: Distribution of languages/frameworks used for developing DSMLs, metamodels, tools over authors' affiliation country.

### 5.3.3 Addressed CPS components

In this section, the results and findings for **“RQ3: What is/are the CPS component(s) addressed in the study?”** are presented.

According to (Gunes et al., 2014), a CPS mainly consists of 5 components, which are, Physical components, Cyber components, Sensors, Actuators, and Network. Amongst the 140 primary studies, only 6 papers were left undetermined (the addressed CPS component by the 6 papers could not be determined) and 9 studies addressed more than 1 CPS component. Figure 5.16 shows the categories of CPS components. The full list of studies and their supporting CPS components is given in Table 5.11.

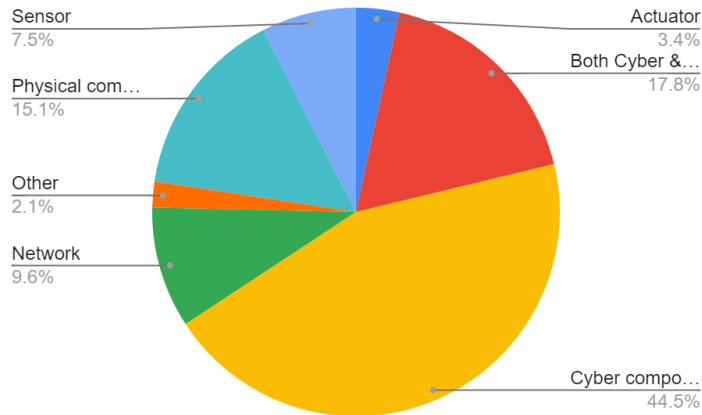


Figure 5.16: Reported CPS component(s)

- **Cyber Components:** 65 studies (44.5%) addressed this component, i.e. the software aspect of the system. Examples are Controllers (e.g. [6, 7]), Development Artifacts related with transformation [86], simulation [116], System verification [140]. System behavior covering timing behavior [49], System safety properties [100], and system requirements [55, 134].
- **Physical Components:** Reported by 22 studies (15.1%). These studies addressed the physical and hardware components of the system, e.g. Physical Dynamics (environment behavior) [27, 30], Power plant [64, 72, 121], Hardware [24, 137].
- **Both Cyber & Physical components:** Reported by 26 studies (17.8%). This category contains the studies discussing modeling both cyber and physical aspects of the system. [18] reported about modeling a controller (cyber component), and a plant (physical component). Another example is [136] where the authors modeled a lunar rover robot's body (physical component) and its navigation system (cyber component).
- **Network:** Reported by 14 studies (9.6%). Studies in this group addressed issues like; sensor network [133], network security [76], physical attacks [21], security requirements and attacks [84].
- **Sensors:** 11 studies (7.5%) reported this component. Studies in this group addressed the different operations of sensors, like sensor designing [41], sensor management [77], sensor data analysis [9], and sensor failures [87].



- **Actuators:** Reported by only 5 studies (3.4%). This component is less addressed one compared to the other CPS components. For instance, [41] covered actuator modeling and design, while [109] discussed actuator failure.
- **Other:** Studies, which do not fit any of the above categories, are grouped under this category. They consider Business processes [14], workflows (process) [111] and data [102].

Table 5.11: Addressed CPS components and the corresponding studies.

Category	# of studies	Relevant studies
Cyber component	65	[7, 6, 116, 49, 100, 39,126, 91, 86, 67, 40, 140,132, 59, 19, 10, 114, 22,54, 75, 79, 46, 17, 62, 90,138, 60, 99, 81, 92, 68,42, 118, 70, 129, 34, 11,58, 128, 93, 101, 124,16, 66, 33, 117, 12, 88, 8,115, 3, 106, 74, 134, 107,103, 98, 113, 55, 104, 37,25, 85, 45, 47]
Both Cyber & Physical components	26	[78, 136, 112, 110, 15,135, 35, 56, 51, 71, 4, 52,123, 26, 130, 65, 94, 89,61, 18, 83, 50, 69, 28, 44,1]
Physical component	22	[5, 120, 72, 48, 95, 24,137, 97, 23, 127, 20, 73,125, 29, 38, 27, 57, 121,64, 108, 13, 30]
Network	14	[2, 119, 76, 31, 63, 36, 54,133, 32, 21, 82, 84, 122,131]
Sensor	11	[119, 41, 77, 31, 132, 133,32, 9, 109, 34, 87]
Actuator	5	[41, 31, 32, 109, 34]
Other	3	[14, 111, 102]

### 5.3.4 Targeted CPS application domains

In this section, the results and findings for “**RQ4: Does the study present any application domain?**” and its sub-questions are presented.

Figure 5.17 depicts the reported CPS application domains. Here, the study is reporting about the CPS domains targeted by the primary studies. There are various CPS domains, such as Critical Infrastructure, Smart Manufacturing, Air Transportation, Emergency Response, Intelligent Transportation, Health Care and Medicine (Gunes et al., 2014). 63 studies out of total 140 studies (about 45%),

addressed a specific CPS domain, while the rest of them addressed CPS in general. This implies that the modeling activities presented in those studies can be applied to any domain of CPS. CPS application domains are correlated with the evaluation methods presented by the examined studies. Results of this correlation are presented in Table 5.12.

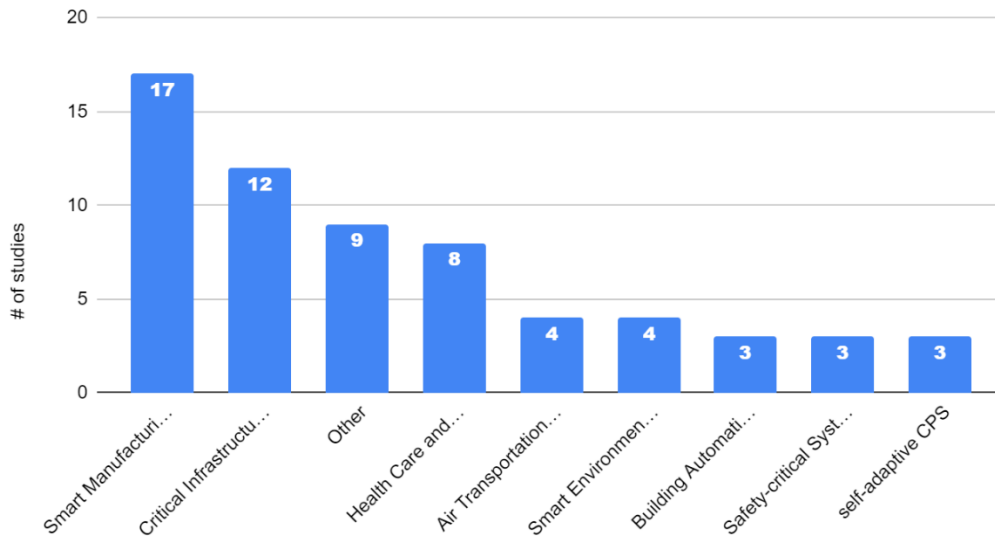


Figure 5.17: Reported CPS application domains targeted by the studies.

- **Smart Manufacturing:** Addressed by 17 out of total 63 studies (26.98%). Studies under this category aim at optimizing productivity in factories (smart factories). Applications included in these studies take into account Industry 4.0/CPSP [14, 15, 16, 29, 57, 69, 76, 121], industrial applications [108, 115, 126], automation systems [19, 135], evolvable production systems [23, 24], and assembly systems (ASs) [120].
- **Critical Infrastructure:** 12 studies (19.05%) reported under this category. It refers to the public infrastructures and valuable properties. Applications grouped under this category cover smart grids [12, 13, 26, 51, 64, 84, 122, 130], irrigation networks [119], railway networks [36, 65], water distribution systems [54].
- **Health Care and Medicine (HC&M):** 8 studies (12.70%) presented under this category. Included sub-categories are Medical Cyber-Physical Systems (MCPS) [9, 85, 112, 113], medical best practice guidelines [48, 49] and smart medical devices [97, 128].

- **Smart Environments:** Addressed by 4 studies (6.35%). The smart environment is a physical environment in which sensing, actuating, network, and computation capabilities are enriched. The followings are the studies grouped under this category: [30, 31, 32, 111].
- **Air Transportation:** 4 studies (6.35%) reported under this category. Applications are; Unmanned Aerial Vehicles [42, 43], Air Traffic Control (ATC) [82] and Aerospace CPS [136].
- **Safety-critical Systems:** Reported by 3 studies (4.76%). Safety-critical systems are systems whose failure or malfunction can have a severe loss, in terms of human or economic consequences. Studies of this cluster include [33, 127, 139].
- **Building Automation:** Reported by 3 studies (4.76%). Studies in this category aim at providing optimum automation and control to buildings' heating, air conditioning, lighting, etc. by deploying sensors, actuators, and control systems. Studies classified under this group are [2, 28, 86].
- **Self-adaptive Systems:** 3 studies (4.76%) presented under this category. Self-adaptive systems are systems that modifies their own behavior during the runtime using feedback due to the constant changes in the system. The followings are the studies grouped under this cluster: [5, 77, 132].
- **Other:** Studies which did not fit any of the aforementioned categories are grouped under this category. They are as follows: Distributed cyber-physical systems [6], smart contracts [39], networked control systems [63], racing sailboats [68], intelligent transportation [20], smart systems [73], material handling applications [4], cloud-based CPS [34], complex systems [25].

Table 5.12: CPS application domain correlated with the evaluations presented by the studies.

Domain	Evaluation type	Description
Smart Manufacturing	Case study	IKEA Gregor office chair [120], assembly production system [24], assembly system [23], Petroamazonas EP Oil Company [57], liqueur plant [121], industrial water process system [115], enterprise production line [69]
	Empirical study	OMiRob [126]
	Example	robot packaging system [80], Pick and Place Unit [15], Vehicular Ad-hoc NETwork [135], pneumatic stopper unit [16], water treatment plant SWaT [108]
	Use case	end-to-end communication use case for an Industry 4.0 application [76], White-goods production [29]
Critical Infrastructure	Case study	flood level prediction [119], SCADA system[84], secondary-voltage control system [130]
	Empirical study	Smart Grid [51], Water Distribution System[54]
	Example	Railway network [36], monitoring of smart grids [122], smart meter [26], process plant de-sign [13]
	Use case	Virtual Power Plant [64]
Health Care and Medicine	Case study	Simplified stroke [49], simplified cardiac arrest [48], Holter Monitor [97], Clinical scenario[113], Generic Patient Controlled Analgesia In-fusion Pump (GPCA) system [85].
	Empirical study	clinical scenarios [112]
	Use case	patient-controlled analgesia infusion pump [128]
Smart Environment	Case study	smart environment scenario [31], smart office[32,30]
	Example	newspaper fetching task [111]
Air Transportation	Case study	lunar rover system [136], Unmanned Aerial Vehicle [42]
	Example	VTOL Unmanned Aerial Vehicle (43)
	Use case	Malaysia Airlines Boeing 777 carrying flight MH-370 [82]
Safety-critical Systems	Case study	battery management system [127], railway signaling system [139]
	Empirical study	rocket system and its payload [33]
Building Automation	Case study	energy-aware building [28]
	Example	Smart Building [2], Room Thermostat [86]
self-adaptive CPS	Case Study	Smart Power Grid [5], self-driving miniature vehicle [77], Water Monitoring [132]

### 5.3.5 Conducted evaluations for the proposed solutions

In this section, the results and findings for “**RQ5: Is there any evaluation presented in the study?**” and its sub-questions are presented.

Out of the 140 studies considered in this thesis, 129 studies (92.1%) evaluate their proposed solution. Among these studies, 70 of them (54.3%) perform this evaluation by means of a case study, 31 of them (24%) present an example, 17 studies (13.18%) conduct an empirical study, and 11 studies (8.53%) present a use case. This SLR groups the presented these evaluations performed by the primary studies into specific clusters to find out some patterns about them. 82 studies out of the 129 studies fit into the clusters shown in (Figure 5.18), while the other 47 studies which do not fit in any of the clusters are grouped under "Other" cluster – not shown in the chart. The raw data related to this analysis is available online<sup>7</sup>.



Figure 5.18: Distribution of the conducted evaluations' categories over the evaluation type.

<sup>7</sup> [https://docs.google.com/spreadsheets/d/1fVPirG-KnWaUFIV6MXvLsxq\\_VR7bPjNPSH\\_L3xhzNDU/edit?usp=sharing](https://docs.google.com/spreadsheets/d/1fVPirG-KnWaUFIV6MXvLsxq_VR7bPjNPSH_L3xhzNDU/edit?usp=sharing)

### 5.3.6 Addressed CPS challenges

In this section, the challenges which the primary studies addressed are reported according to the “**RQ6: Does the study address any challenge(s)?**” in addition to its sub-questions.

107 studies out of 140 studies (76.43%) reported the CPS challenge(s) they faced. Reported CPS challenges are shown in (Figure 5.19). It is worth mentioning that several studies addressed more than one CPS challenge. In order to relate to the challenges presented by the studies to one another, the categorization of CPS challenges presented in (Gunes et al., 2014) was also followed in this study, with considering complexity as a separate category. Reported CPS challenges and their corresponding studies are shown in (Table 5.13).

- **Complexity:** 34 studies (22.82%) were classified under this category. It is reasonable that complexity was the most reported challenge, due to the nature of the CPS development process that requires complex engineering work. Some of the addressed complexity challenges include: complexity of design, timing behavior specification, execution complexity, co-simulation construction, architecture complexity, interaction complexity, semantics complexity, interdependency complexity, requirements complexity.

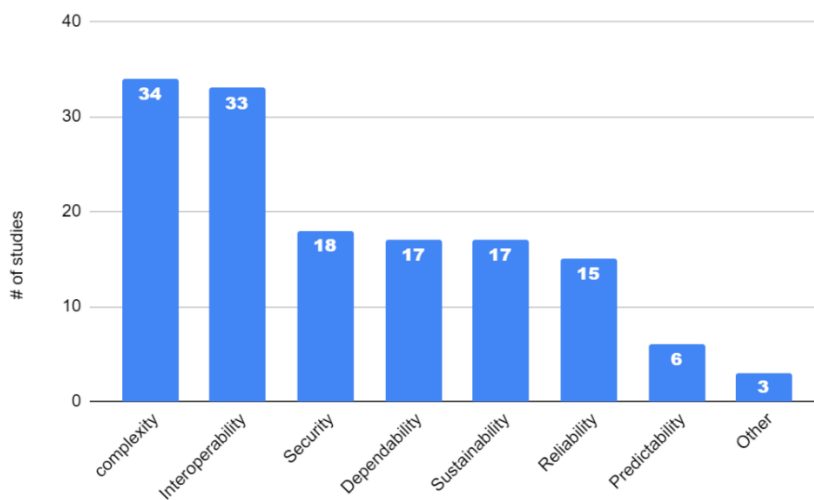


Figure 5.19: Categorized CPS challenges reported by the studies.

- **Interoperability:** also means Heterogeneity. 33 studies (22.15%) classified under this category. To develop a CPS, the collaboration of different disciplines is a must. Thus, CPS combines different components (i.e. hardware, software, sensors, network, etc.), hence, managing and coordinating all these disciplines and operations are challenging. Scalability and composability are two important types of interoperability challenge. Scalability is challenging since the system ought to keep functioning adequately when new features are added. To provide the composability, CPS development should consider combining several components within a system and managing their interrelationships.
- **Security:** Reported by 18 studies (12.08%). Studies in this category are concerned about the 3 security aspects of the CPS. Firstly, integrity needs to be supplied to protect the correctness of information from being manipulated or modified. An example for the CPS integrity problem would be compromising a sensor/actuator and injecting false data. Second aspect is confidentiality, that refers to allowing only authorized individuals to get access to the data. Third aspect is availability which means keeping the CPS components on service, e.g. preventing cyber- attacks (like denial of service) that may limit or block the availability of the system.
- **Dependability:** The ability of the system to keep functioning as required. 17 studies (11.41%) were covered under this category. It encompasses aspects like safety, and maintainability. The system must be maintainable simply when a failure occurs.
- **Sustainability:** 17 studies (11.41%) were covered under this category. It refers to challenges like adaptability, efficiency in using resources, reconfigurability, uncertainty, performance measurement, and optimization.
- **Reliability:** 15 studies (10.07%) were covered under this category. Reliability means that the CPS should function correctly not only in closed and fixed environments but also in open and uncertain environments. Challenges to address are; fault tolerance, robustness, timing uncertainty etc.

- **Predictability:** 6 studies (4.03%) were in this group. Predictability refers to the degree to which the system's behavior/functionality and outcomes are predictable and they satisfy the system requirements. For instance; predicting system's stochastic behavior and accuracy, that is, the degree to which the system's measured outcomes need to be accurate.
- **Other:** This category contains other challenges which are concurrency, latency and remote monitoring.

Table 5.13: CPS challenges and their corresponding studies.

CPS challenge	# of studies	Relevant studies
Complexity	34	[7, 100, 91, 86, 41, 40,132, 63, 59, 19, 114, 110,75, 120, 48, 46, 35, 90,68, 42, 71, 125, 70, 52,111, 11, 101, 65, 64, 134,107, 103, 53, 85]
Interoperability	33	[7, 126, 91, 80, 136, 63,59, 114, 135, 79, 46, 62,90, 56, 51, 123, 122, 29,34, 11, 43, 124, 16, 121,12, 115, 61, 3, 83, 103,104, 139, 47]
Security	18	[2, 49, 78, 76, 36, 54, 72,21, 84, 26, 93, 131, 117,8, 89, 108, 106, 50]
Dependability	17	[77, 136, 19, 22, 60, 20,71, 109, 4, 84, 125, 87,88, 74, 55, 37, 85]
Sustainability	17	[119, 39, 76, 140, 132,112, 10, 120, 92, 4, 14,122, 58, 38, 130, 102, 69]
Reliability	15	[40, 140, 136, 36, 19, 112,10, 24, 127, 20, 51, 9, 14,117, 64]
Predictability	6	[5, 112, 62, 9, 27, 3]
Other	3	[67, 82, 57]

Further, in this SLR, a correlation analysis of the CPS domains and its challenges is scrutinized so as to provide an understanding of the challenges addressed in each CPS application domain, (see Table 5.14). Despite the fact that the correlation analysis cannot indicate the CPS domain wholly, for instance, one can see that in the smart manufacturing application domain, most research works converged on interoperability and sustainability challenges. Similarly, in the critical infrastructure application domain, most research works concentrated on security, sustainability, and interoperability challenges. However, it is interesting to note that the latency and the predictability challenges of both domains were not addressed by any of the examined papers.



**RQ6.2:** Did the study report challenges addressed while developing the approach/tool?

Only 15%, that is, 21 studies out of the 140 studies reported about the limitations they faced. These studies are [2, 78, 80, 63, 112, 133, 79, 60, 20, 51, 129, 93, 57, 131, 64, 3, 108, 83, 103, 85, 45].



Table 5.14: Correlation analysis between CPS application domains and its challenges.

	complexity	Dependability	Interoperability	Latency	Predictability	Reliability	Remote monitoring	Security	Sustainability
<b>Air Transportation (AT)</b>	[42]	[136]	[136, 43]	[82]		[136]			
<b>Building Automation (BA)</b>	[86]							[2]	
<b>Critical Infrastructure (CI)</b>	[65, 64]	[84]	[51,122, 12]			[36, 51,64]		[36, 54,84, 26]	[119, 122, 130]
<b>Health Care and Medicine (HC&amp;M)</b>	[48, 85]	[85]			[112, 9]	[112, 9]		[49]	[112]
<b>Safety-critical Systems</b>			[139]			[127]			
<b>self-adaptive CPS</b>	[132]	[77]			[5]				[132]
<b>Smart Environment (SE)</b>	[111]								
<b>Smart Manufacturing (SM)</b>	[19,120]	[19]	[126,80, 135,29, 16,121,115]			[19, 24,14]	[57]	[76,108]	[76,120, 14,69]

## 6. Discussion

In this section, discussion of the findings achieved as the result of the applied research workflow of this SLR study is given along with its implications. Threats to the validity of the study is also discussed in this section.

At first, the quantitative analysis revealed that the number of published research papers in this field continues to increase year after year. USA affiliated researchers are the most interested researchers in this field (39 studies), followed by China (23 studies). Moreover, most preferred publication venues are conferences (64.75%, 90 studies) by far.

RQ 1.1 revealed that the metamodeling was the most used approach by the researchers. Model-based and DSL approaches follow the metamodeling. Also, modeling approaches were correlated with the authors' affiliation country in an attempt to determine which of the modeling approaches are mostly used in different countries. The study found out that, metamodeling and model-based approaches are mostly adopted by researchers affiliated to Europe, while DSL based approach was adopted mostly by USA affiliated researchers.

Although, in terms of the number of studies, metamodeling is the most adopted modeling approach, yet component-based approach is the most reported modeling approach in terms of the number of activities it is used for, which covered 9 activities namely: Adaptability, Analysis, Correctness, Development, Efficiency, Flexibility, Security, Simulation, and V&V.

As far as the purpose of modeling is concerned (**RQ1.1.1**), the most-reported purpose of modeling was development, that is, developing either DSL, metamodel, tool or automating the development process of a system. Other reported modeling purposes were Analysis (like safety analysis, performance analysis, requirement analysis, etc.), V&V (DSML validation, metamodel verification, behavior verification, etc.), and security (threat modeling, attack modeling, analyzing cyber-attacks, etc.).

Regarding model-driven engineering activity/phase addressed with **RQ 1.2**, the most considered MDE activity was system design. Researchers developed DSLs (15 studies), metamodels (22 studies) and tools (18 studies). Since the total

number of DSL studies is quite low (10.71%, only 15 studies out of 140) in a complex domain like CPS, this underpins the necessity for conducting more research to design DSLs to address different aspects of CPS development lifecycle. DSLs can provide a higher level of abstraction for complex systems such as CPS which may lead to increase the performance and to decrease the time and the cost of CPS development. Simulation was the second most reported MDE activity (40 studies, 16.81%). Apart from 1 study that developed a tool [122] and 2 studies that developed metamodels [29, 80] for simulation purposes, the rest of the studies (37 studies) used existing simulation tools/languages. Therefore, one can observe that there is a research gap in developing domain-specific simulation tools/languages for CPS.

Furthermore, **RQ 1.2** revealed that M2M transformation gains more attention in terms of the existing / developed tools, and languages in comparison with the other transformation types M2T and T2M. In addition, it is observed that languages like GenERTiCA and Xtext were used for the implementation of more than one transformation type. Also, it is worth mentioning that tools like UPP2SF and STU can be used as a complementary tool for M2M transformation. Other complementary languages for modeling CPS are Modelica and AADL, where Modelica is used for modeling the physical world and AADL for modeling the cyber part, and the transformation between these two languages do not require any third-party language or tool [136, 138]. V&V was reported by 35 studies (14.71%,). However, apart from 2 studies [45, 127] which developed a tool and one study that developed an ontology [75], the rest of the studies used only existing tools.

Results of **R.Q 2.1** showed that the top 10 most used languages/tools in the field of applying MDE paradigm on CPS are; Simulink, AADL, UML, UPPAAL, SysML, MATLAB, Petri nets, Ptolemy II, Modelica and Papyrus. **RQ 2.2** revealed that 59 studies out of 140 developed DSL, metamodel or tool. However, only 10 of the 59 studies reported the availability of these developed tools/languages (i.e. can be downloaded in their paper) according to the results for **RQ 2.2.1**. Therefore, this is also another alarming fact to consider by the CPS community that is interested in applying MDE for CPS; particularly, if they intend to have an impact on the industry. The results of **RQ2.2.2** revealed that UML,

EMF, and GME were the most used tools/environments which these 59 studies used while developing their DSLs, metamodels, and tools. However, GME was not present for the last 2 years (2017 and 2018) in any study. Findings also revealed that GME was mostly used by the researchers affiliated to the USA, while UML and EMF were mostly used by the researchers affiliated to Europe.

Regarding the addressed CPS components, most of the papers focused on the cyber and physical components of CPS (**R.Q 3**). There is limited work on the other components (sensor, network, actuator), Especially, the actuator is the component that received the least attention by the researchers on this topic. Results for **R.Q 4** showed that 63 studies out of 140 (45%) addressed a specific CPS domain. Smart manufacturing is the most addressed CPS domain by the researchers (26.98%, 17 studies out of 63). The other domains followed are Critical Infrastructure, Health Care and Medicine, Air Transportation, Smart Environment, Building Automation, Safety-critical Systems, and self-adaptive CPS respectively.

For the evaluation method, **R.Q 5** results revealed that the majority of the studies (54.26%, 70 studies) presented case study(s) as the major evaluation method for their proposed solution. On the other hand, only 17 studies (13.18%) presented an empirical evaluation for their proposed solution. That is, conducting empirical evaluations for this topic is an area which still requires much attention.

Results for **R.Q 6** showed that a variety of CPS challenges were addressed. However, the most addressed challenges were complexity and interoperability. The much focus for these two challenges can be related to the heterogeneous nature of CPS. CPS combines different components and requires the interaction of different researchers from different backgrounds. Thus, it informs why researchers interested in this field should pay more attention to reducing the complexity and interoperability aspects of CPS. Other challenges addressed were Security, Dependability, Sustainability, Reliability, Flexibility and Predictability.

Finally, the results of the study also showed several research gaps that the researchers in the community may take into account. First of all, the development of domain-specific languages, and domain-specific simulation and verification

tools for CPS needs to be provided. Applying MDE on the different types of actuator components used inside CPS also needs further investigation since their role on building many CPS is vital and they are less addressed in the current MDE studies compared to the other CPS components. Moreover, conducting empirical evaluations in this field is missing which is critical on the assessment of the proposed modeling approaches especially on their usability for both the construction and the execution of CPS.

## 6.1 Threats to validity

Threats to validity for this SLR study are classified according to categories proposed in (Wohlin et al., 2012), and hence they include four types, namely construct, internal, external and conclusion validity threats.

- Construct validity

It represents how the SLR study truly reflects the intent of the researchers, and what is asked by the research questions. To define the research questions, it is important to stress that the process proposed by (de Almeida Biolchini et al., 2007; Siddaway, 2014) and using guidelines defined by (Kitchenham and Charters, 2007) were followed in this study.

Furthermore, another aspect of construct validity is to assure that all relevant studies on the selected topic are found adequately. The possibility of missing primary studies is a common threat to the validity of any SLR. Both the terms MDE and CPS are well-established concepts, and thus, the terms are sufficiently good enough to be used as keywords. Therefore, to mitigate this risk, a good-enough search string through several iterations was formed, and adequate coverage of literature was achieved. General publication databases, which index most well-reputed publication venues, were extensively searched in this study as well. The list of publication venues shown in the online repository<sup>8</sup> indicates that the coverage of the search is enough. Also, to improve the results, the forward snowballing sampling method was used, and it has proved to be effective.

---

<sup>8</sup>[https://docs.google.com/spreadsheets/d/1VJhjimNVXU7\\_zxCri1Lf9WuhBD5phNWjYflNbzRtOGM/edit?usp=sharing](https://docs.google.com/spreadsheets/d/1VJhjimNVXU7_zxCri1Lf9WuhBD5phNWjYflNbzRtOGM/edit?usp=sharing)

- Internal validity

This relates to the degree to which the design and the conduct of the SLR study are likely to prevent systematic errors. Internal validity is a prerequisite for external validity (Kitchenham and Charters, 2007). Therefore, both qualitative and quantitative analysis were used to minimize threats. The use of a rigorous protocol and data extraction form mitigates this kind of threats to validity. Moreover, threats originating from personal bias or lack of understanding of the study were reduced by conducting data extraction phase iteratively. For this purpose, one researcher extracted data from the primary studies and answered quality and self-assessment questions. The other two researchers (expert in CPS and MDE) reviewed the extracted data from studies with low self-assessment rates below 50%.

- External validity

According to (Wohlin et al., 2012), external threats concern the generalizability of the SLR results, that is, the degree to which the primary studies is representative of the reviewed topic. In this thesis, the set of primary studies may not be representative of the entire set of existing studies on the topic, MDE for CPS. However, this threat was mitigated as follows; Firstly, the search strategy consisted of manual and automatic search, then followed by the forward snowballing. The forward snowballing enabled finding studies which were not captured by the search strings in the digital libraries. Secondly, the inclusion and exclusion criteria of the protocol created in this thesis support refining the set of primary studies which leads to include only studies which meet the topic. Only studies in English were included. Papers written in other languages concerning the same topic may exist. However, this threat is considered as having minimal effect.

- Conclusion validity

All relevant primary studies cannot be identified (Kitchenham and Charters, 2007). To alleviate this threat, the research protocol of this study was designed and validated carefully to minimize the risk of excluding relevant studies. Search strings were formed in a way that only a very small number of relevant studies could be missed, and a manageable quantity of irrelevant studies could be included. Besides the automatic search, a manual search and a forward

snowballing were performed. The protocol was rigorously defined to be reusable by other researchers for reproducing the same study, i.e. the data extraction form is available online<sup>9</sup>.



---

<sup>9</sup> <https://docs.google.com/spreadsheets/d/1K1vpUr5UKDz0KPKI-65TbP413XZpD6cKRcbDCJn-cXk/edit?usp=sharing>



## 7. Conclusion

CPS have proven to offer tremendous opportunities in almost all areas of industry and society. Due to its inherent heterogeneity and complexity, developing and managing such systems is known to be a challenge for the developers. Thus, numerous researches were conducted and are still being conducted in this domain.

The aim of this study was to identify the current features of the use of MDE for CPS. For this purpose, an SLR of the papers in the field, published between 2010 and 2018, was performed. The initial search retrieved 646 papers of which 140 were included in this study by following the defined selection strategy through a multi-stage process. A key feature of this SLR is that it is not restricted to a particular CPS domain. This broad scope in the search gives deeper insights into the state-of-the-art of using MDE in CPS. Furthermore, the study presented bibliometrics analysis to attain an understanding of the active researchers, publication trends per year, and publication venues in the area. Findings contribute new knowledge that can be used to improve CPS development using MDE.

The study points out that MDE for CPS is an active research area with an increasing number of publications over the years. Results showed that conferences account for the most frequently used publication venue. In terms of CPS domains, smart manufacturing is the most addressed CPS domain. Furthermore, the study showed areas that have been covered, and approaches, techniques, languages, and tools proposed. Regarding the CPS components, the effort was mostly put on the cyber and physical components, where the other components (sensors, network, actuator) did not get much attention. Study results revealed that solutions based on UML and Eclipse-based tools were mostly preferred.

Finally, the study also provided guidelines for assisting researchers to plan future work by pointing out research areas which need more attention. For instance, designing and modeling actuator used in CPS, code generation for an actuator, verification of actuator code, conducting an empirical evaluation, and developing domain-specific simulation tools required further investigation.

**REFERENCES**

- Agner, L. T. W., Soares, I. W., Stadzisz, P. C., and Simão, J. M.,** 2013, A Brazilian survey on UML and model-driven practices for embedded software development. *Journal of Systems and Software*, 86(4), 997-1005.
- Barišić, A., Savić, D., Al-Ali, R., Ruchkin, I., Blouin, D., Cicchetti, A., ... Goulão, M.,** 2018, Systematic Literature Review on Multi-Paradigm Modelling for Cyber-Physical Systems. <https://doi.org/10.5281/zenodo.2528953>
- Brambilla, M., Cabot, J., and Wimmer, M.,**2017, Model-driven software engineering in practice. *Synthesis Lectures on Software Engineering*, 3(1), 1-207.
- Casalaro, G. L., and Cattivera, G.,** 2015, Model-Driven Engineering for Mobile Robot Systems: A Systematic Mapping Study. Master thesis, Malardalen University, Västerås, Sweden.
- de Almeida Biolchini, J. C., Mian, P. G., Natali, A. C. C., Conte, T. U., and Travassos, G. H.,** 2007, Scientific research ontology to support systematic review in software engineering. *Advanced Engineering Informatics*, 21(2), 133-151.
- France, R., and Rumpe, B.,** 2007, May, Model-driven development of complex software: A research roadmap. In *2007 Future of Software Engineering* (pp. 37-54). IEEE Computer Society.
- Garousi, V., Mesbah, A., Betin-Can, A., and Mirshokraie, S.,** 2013, A systematic mapping study of web application testing. *Information and Software Technology*, 55(8), 1374-1396.
- Gunes, V., Peter, S., Givargis, T., & Vahid, F.,** 2014, A survey on concepts, applications, and challenges in cyber-physical systems. *KSII Transactions on Internet & Information Systems*, 8(12).
- Kitchenham, B., and Charters, S.,** 2007, Guidelines for performing systematic literature reviews in software engineering.
- Lee, E.,** 2015, The past, present and future of cyber-physical systems: A focus on models. *Sensors*, 15(3), 4837-4869.

**REFERENCES (continued)**

- Lee, E. A.**, 2008, May, Cyber physical systems: Design challenges. In 2008 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC) (pp. 363-369). IEEE.
- Liebel, G., Marko, N., Tichy, M., Leitner, A., and Hansson, J.**, 2018, Model-based engineering in the embedded systems domain: an industrial survey on the state-of-practice. *Software & Systems Modeling*, 17(1), 91-113.
- Petersen, K., Feldt, R., Mujtaba, S., and Mattsson, M.**, 2008, June, Systematic Mapping Studies in Software Engineering. In EASE (Vol. 8, pp. 68-77).
- Pinsonneault, A., and Kraemer, K.**, 1993, Survey research methodology in management information systems: an assessment. *Journal of management information systems*, 10(2), 75-105.
- Queiroz, P. G. G., and Braga, R. T. V.**, 2014, September, Development of critical embedded systems using model-driven and product lines techniques: A systematic review. In *Software Components, Architectures and Reuse (SBCARS)*, 2014 Eighth Brazilian Symposium on (pp. 74-83). IEEE.
- Rashid, M., Anwar, M. W., and Khan, A. M.**, 2015, Toward the tools selection in model based system engineering for embedded systems—A systematic literature review. *Journal of Systems and Software*, 106, 150-163.
- Sanislav, T., and Mois, G.**, 2017, May, A dependability analysis model in the context of cyber-physical systems. In 2017 18th International Carpathian Control Conference (ICCC) (pp. 146-150). IEEE.
- Siddaway, A.**, 2014, What is a systematic literature review and how do I do one. University of Stirling, (I), 1.
- Skoglund, M., and Runeson, P.**, 2009, April, Reference-based search strategies in systematic reviews. In EASE.
- Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B., and Wesslén, A.**, 2012, *Experimentation in software engineering*. Springer Science & Business Media.

**REFERENCES (continued)**

**Wohlin, C.**, 2014, Guidelines for snowballing in systematic literature studies and a replication in software engineering. In Proceedings of the 18th international conference on evaluation and assessment in software engineering (p. 38). ACM.



## ACKNOWLEDGEMENT

In the beginning, thanks belong to the almighty Allah, who gave me the strength, health and the ability to complete this work successfully. Also, I would like to express my profound gratitude to the Turkish government's Scholarships (YTB) program for granting me the rare opportunity to pursue this Master's degree under their flagship scholarship program.

I want to thank and express my gratitude to Associate Prof. Dr. Geylani Kardaş for his sincere comments and magnanimous guidance during the course of this dissertation. My gratitude also goes to Dr. Moharram Challenger for his unbounded contributions both academically and morally, and for being the biggest contributor towards the successful completion of my program.

I appreciate and thank my friends Oluwaseun Adeoye OYEBAMIJI, Tansu AŞICI, Burak KARADUMAN, Ruwad ALKARAM, Malath Thaer NEJRES, Husein MARAH for their contribution to this work. I would also like to acknowledge my family for their abundant support, for their patience and understanding. My special gratitude goes to my Mother Sado Iman DIRIE for having faith in me, teaching and guiding me to be who I am today.

And to every single person who spent their valuable time to teach me, even if it was a single word.

03 / 02 / 2020

Mustafa Abshir MOHAMED

## RESUME

Mustafa Abshir MOHAMED was born on 10/01/1994 in Wajale, Somaliland/Somalia. He had his primary, intermediate, and secondary education in Yemen from 1999-2011. Later, in 2012, he was admitted to Gollis University where he received a BSc in Information Communication Technology (ICT). In 2016, he was awarded a scholarship from the Turkish government under the program Türkiye Scholarships (YTB) and took a Turkish preparatory course in the academic year 2016-2017 at Ege University. In the 2017-2018 academic year, he started his graduate studies at Ege University, International Computer Institute.

### **WORK HISTORY**                      **INTERNATIONAL STUDENTS SCIENCE CONGRESS**

03-04/May/2019

**Title:** Organizing committee,

<https://sciencecon.org/committee/organizing-committee/>

### **EUROPEAN COOPERATION IN SCIENCE AND TECHNOLOGY (COST)**

10/2018 – On going, IC1404 – COST

**Title:** Systematic Literature Review on Multi-Paradigm Modelling for Cyber- Physical Systems (MPM4CPS).

### **GOLLIS UNIVERSITY, HARGEISA, SOMALIA**

2015 - 2016

Worked at Gollis university Hargeisa branch as the software developer and the database administrator. During which I Developed the Main systems for this organization.

### **SAHAN HIV/AIDS NETWORK, HARGEISA, SOMALIA**

Project 2016

Worked with SAHAN NETWORK to develop Consultant Center Management System for receiving calls from clients, record them and later store the data collected on a database to make statistical purposes.

## APPENDIXES

### List of primary studies

- [1] **Ali, N., & Hong, J.-E.**, 2018, Failure detection and prevention for cyber-physical systems using ontology-based knowledge base. *Computers*, 7 (4), 68.
- [2] **Alrimawi, F., Pasquale, L., Mehta, D., & Nuseibeh, B.**, 2018, I've seen this before: sharing cyber-physical incident knowledge. In 2018 IEEE/ACM 1st international workshop on security awareness from design to deployment (SEAD) (pp. 33–40).
- [3] **Alshareef, A., & Sarjoughian, H. S.**, 2018, Model-driven time-accurate DEVS-based approaches for cps design. In *Proceedings of the model-driven approaches for simulation engineering symposium* (p. 8).
- [4] **An, K., Trewyn, A., Gokhale, A., & Sastry, S.**, 2011, Model-driven performance analysis of reconfigurable conveyor systems used in material handling applications. In 2011 IEEE/ACM second international conference on cyber-physical systems (pp. 141–150).
- [5] **Angelo, M., Napolitano, A., & Caporuscio, M.**, 2018, Cyphef: a model-driven engineering framework for self-adaptive cyber-physical systems. In *Proceedings of the 40th international conference on software engineering: Companion proceedings* (pp. 101–104).
- [6] **Ataíde, A., Barros, J. P., Brito, I. S., & Gomes, L.**, 2017, Towards automatic code generation for distributed cyber-physical systems: A first prototype for Arduino boards. In 2017 22nd IEEE international conference on emerging technologies and factory automation (ETFA) (pp. 1–4).
- [7] **Aziz, M. W., & Rashid, M.**, 2016, Domain specific modeling language for cyber physical systems. In 2016 international conference on information systems engineering (ICISE) (pp.29-33).
- [8] **Bakirtzis, G., Carter, B. T., Elks, C. R., & Fleming, C. H.**, 2018, A model-based approach to security analysis for cyber-physical systems. In 2018 annual IEEE international systems conference (SYSCON) (pp. 1–8).

- [9] **Banerjee, A., & Gupta, S. K.**, 2014, Model based code generation for medical cyber physical systems. In Proceedings of the 1st workshop on mobile medical applications (pp. 22–27).
- [10] **Bao, S., Porter, J., & Gokhale, A.**, 2016, Reasoning for cps education using surrogate simulation models. In 2016 IEEE 40th annual computer software and applications conference (COMPSAC) (Vol. 1, pp. 764–773).
- [11] **Barbieri, G., & Fantuzzi, C.**, 2016, Design of cyber-physical systems: Definition and metamodel for reusable resources. In 2016 IEEE 21st international conference on emerging technologies and factory automation (ETFA) (pp. 1–9).
- [12] **Barve, Y., Neema, H., Rees, S., & Sztipanovits, J.**, 2018, Towards a design studio for collaborative modeling and co-simulations of mixed electrical energy systems. In 2018 IEEE international science of smart city operations and platforms engineering in partnership with global city teams challenge (SCOPE-GCTC) (pp. 24–29).
- [13] **Blackburn, M., & Denno, P.**, 2014, Virtual design and verification of cyber-physical systems: Industrial process plant design. *Procedia Computer Science*, 28, 883–890.
- [14] **Bocciarelli, P., D'Ambrogio, A., Giglio, A., & Paglia, E.**, (2017). A BPMN extension for modeling cyber-physical-production-systems in the context of industry 4.0. In 2017 IEEE 14th international conference on networking, sensing and control (ICNSC) (pp. 599–604).
- [15] **Bougouffa, S., Meßmer, K., Cha, S., Trunzer, E., & Vogel-Heuser, B.**, 2017, Industry 4.0 interface for dynamic reconfiguration of an open lab size automated production system to allow remote community experiments. In 2017 IEEE international conference on industrial engineering and engineering management (IEEM) (pp. 2058–2062).
- [16] **Brandenbourger, B., Vathoopan, M., & Zoitl, A.**, 2016, Behavior modeling of automation components using cross-domain interdependencies. In 2016 IEEE 21st international conference on emerging technologies and factory automation (ETFA) (pp. 1–4).



- [17] **Brito, I. S., Barros, J. P., & Gomes, L.**, 2016, From requirements to code (re2code)—a model-based approach for controller implementation. In 2016 IEEE 14th international conference on industrial informatics (INDIN) (pp. 1224–1230).
- [18] **Broenink, J. F., Vos, P.-J. D., Lu, Z., & Bezemer, M. M.**, 2016, A co-design approach for embedded control software of cyber-physical systems. In 2016 11th system of systems engineering conference (sose) (pp. 1–5).
- [19] **Buckl, C., Sojer, D., & Knoll, A.**, 2010, FTOS: Model-driven development of fault-tolerant automation systems. In 2010 IEEE 15th conference on emerging technologies & factory automation (ETF A 2010) (pp. 1–8).
- [20] **Bunting, M., Zeleke, Y., McKeever, K., & Sprinkle, J.**, 2016, A safe autonomous vehicle trajectory domain specific modeling language for non-expert development. In Proceedings of the international workshop on domain-specific modeling (pp. 42–48).
- [21] **Cheh, C., Keefe, K., Feddersen, B., Chen, B., Temple, W. G., & Sanders, W. H.**, 2017, Developing models for physical attacks in cyber-physical systems. In Proceedings of the 2017 workshop on cyber-physical systems security and privacy (pp. 49–55).
- [22] **Chen, D., & Lu, Z.**, 2017, A model-based approach to dynamic self-assessment for automated performance and safety awareness of cyber-physical systems. In International symposium on model-based safety and assessment (pp. 227–240).
- [23] **Chen, D., Maffei, A., Ferreirar, J., Akillioglu, H., Khabazzi, M. R., & Zhang, X.**, 2015, A virtual environment for the management and development of cyber-physical manufacturing systems. *IFAC-PapersOnLine*, 48(7), 29-36.
- [24] **Chen, D., Panfilenko, D. V., Khabbazi, M. R., & Sonntag, D.**, 2016, A model-based approach to qualified process automation for anomaly detection and treatment. In 2016 IEEE 21st international conference on emerging technologies and factory automation (ETF A) (pp. 1–8).

- [25] **Chen, R., Liu, Y., & Ye, X.**, 2018, Ontology based behavior verification for complex systems. In ASME 2018 international design engineering technical conferences and computers and information in engineering conference.
- [26] **Chen, T. M., Sanchez-Aarnoutse, J. C., & Buford, J.**, 2011, Petri net modeling of cyber-physical attacks on smart grid. IEEE Transactions on Smart Grid,2 (4), 741–749.
- [27] **Cheng, B., & Du, D.**, 2014, Towards a stochastic occurrence-based modeling approach for stochastic cps. In 2014 theoretical aspects of software engineering conference (pp. 162–169).
- [28] **Cheng, B., Wang, X., Liu, J., & Du, D.**, 2015, Modana: An integrated framework for modeling and analysis of energy-aware cps. In 2015 IEEE 39th annual computer software and applications conference (Vol. 2, pp. 127–136).
- [29] **Ciavotta, M., Bettoni, A., & Izzo, G.**, 2018, Interoperable meta model for simulation-in-the-loop. In 2018 IEEE industrial cyber-physical systems (ICPS) (pp. 702–707).
- [30] **Cicirelli, F., Fortino, G., Guerrieri, A., Mercuri, A., Spezzano, G., & Vinci, A.**, 2018, A metamodel framework for edge-based smart environments. In 2018 IEEE international conference on cloud engineering (IC2E)(pp. 286– 291).
- [31] **Cicirelli, F., Fortino, G., Guerrieri, A., Spezzano, G., & Vinci, A.**, 2016, A meta-model framework for the design and analysis of smart cyber-physical environments. In 2016 IEEE 20th international conference on computer supported cooperative work in design (CSCWD) (pp. 687–692).
- [32] **Cicirelli, F., Fortino, G., Guerrieri, A., Spezzano, G., & Vinci, A.**, 2017, Metamodeling of smart environments: from design to implementation. Advanced Engineering Informatics, 33,274–284.
- [33] **Cohen, R., Long, A.-T. B., Jobredeaux, R., & Feron, E.**, 2015, A credible autocoding application within a rocket and its payload. In 2015

IEEE/AIAA 34th digital avionics systems conference (DASC) (pp. 8C4–1).

- [34] **Dell, J., Greiner, T., & Rosenstiel, W.**, 2014, Model-based platform design and evaluation of cloud-based cyber-physical systems (ccps). In 2014 12th IEEE international conference on industrial informatics (INDIN) (pp. 376–381).
- [35] **Di Natale, M., Morelli, M., & Cremona, F.**, 2015, Matching execution architecture models with functional models to analyze the time performance of cps systems. In 2015 international conference on complex systems engineering (ICCSE) (pp. 1–6).
- [36] **Drago, A., Marrone, S., Mazzocca, N., Tedesco, A., & Vittorini, V.**, 2013, Model-driven estimation of distributed vulnerability in complex railway networks. In 2013 IEEE 10th international conference on ubiquitous intelligence and computing and 2013 IEEE 10th international conference on autonomic and trusted computing (pp. 380–387).
- [37] **Fan, C.-F., Chan, C.-C., Yu, H.-Y., & Yih, S.**, 2018, A simulation platform for human-machine interaction safety analysis of cyber-physical systems. *International journal of industrial ergonomics*, 68, 89–100.
- [38] **Feldmann, S., Rösch, S., Schütz, D., & Vogel-Heuser, B.**, 2013, Model-driven engineering and semantic technologies for the design of cyber-physical systems. *IFAC Proceedings Volumes*, 46 (7), 210–215.
- [39] **Garamvölgyi, P., Kocsis, I., Gehl, B., & Klenik, A.**, 2018, Towards model-driven engineering of smart contracts for cyber-physical systems. In 2018 48th annual IEEE/IFIP international conference on dependable systems and networks workshops (DSN-W) (pp. 134–139).
- [40] **Gerking, C., Schäfer, W., Dziwok, S., & Heinzemann, C.**, 2015, Domain-specific model checking for cyber-physical systems. In *MoDeVVA@ models* (pp. 18–27).
- [41] **Gonçalves, F. S., & Becker, L. B.**, 2016, Model driven engineering approach to design sensing and actuation subsystems. In 2016 IEEE 21st

international conference on emerging technologies and factory automation (ETFA) (pp. 1–8).

- [42] **Gonçalves, F. S., Pereira, D., Tovar, E., & Becker, L. B.**, 2017, Formal verification of AADL models using UPPAAL. In 2017 VII Brazilian Symposium on Computing Systems Engineering (SBESC) (pp. 117–124).
- [43] **Gonçalves, F. S., Raffo, G. V., & Becker, L. B.**, 2016, Managing cps complexity: Design method for unmanned aerial vehicles. *IFAC-PapersOnLine*, 49 (32), 141–146.
- [44] **Graja, I., Kallel, S., Guermouche, N., & Kacem, A. H.**, 2016, BPMN4CPS: A BPMN extension for modeling cyber-physical systems. In 2016 IEEE 25th international conference on enabling technologies: Infrastructure for collaborative enterprises (WETICE) (pp. 152–157).
- [45] **Graja, I., Kallel, S., Guermouche, N., & Kacem, A. H.**, 2017, Verification of the consistency of time-aware cyber-physical processes. In *International conference on service-oriented computing* (pp. 67–79).
- [46] **Gritzner, D., & Greenyer, J.**, 2018, Generating correct, compact, and efficient plc code from scenario-based assume-guarantee specifications. *Procedia Manufacturing* 24 (2018), 24, 153–158.
- [47] Guan, C., Ao, Y., Du, D., & Mallet, F., 2018, XSHS: An executable domain-specific modeling language for modeling stochastic and hybrid behaviors of cyber-physical systems. In 2018 25th ASIA-pacific software engineering conference (APSEC) (pp. 683–687).
- [48] **Guo, C., Fu, Z., Ren, S., Jiang, Y., Rahmaniheris, M., & Sha, L.**, 2017, Pattern-based Statechart modeling approach for medical best practice guidelines-a case study. In 2017 IEEE 30th international symposium on computer-based medical systems (CBMS) (pp. 117–122).
- [49] **Guo, C., Fu, Z., Zhang, Z., Ren, S., & Sha, L.**, 2018, Model and integrate medical resource available times and relationships in verifiably correct executable medical best practice guideline models. In *Proceedings of the 9th ACM/IEEE international conference on cyber-physical systems* (pp. 253–262).

- [50] **Hahn, A., Thomas, R. K., Lozano, I., & Cardenas, A.**, 2015, A multi-layered and kill-chain based security analysis framework for cyber-physical systems. *International Journal of Critical Infrastructure Protection*, 11, 39–50.
- [51] **Hartmann, T., Moawad, A., Fouquet, F., Nain, G., Klein, J., & Le Traon, Y.**, 2015, Stream my models: Reactive peer-to-peer distributed models@ run. time. In *2015 ACM/IEEE 18th international conference on model driven engineering languages and systems (models)* (pp. 80–89).
- [52] **He, X.**, 2018, Modeling and analyzing cyber physical systems using high level petri nets. In *2018 IEEE international conference on software quality, reliability and security companion (QRS-C)* (pp. 469–476).
- [53] **Heinzemann, C., Becker, S., & Volk, A.**, 2019, Transactional execution of hierarchical reconfigurations in cyber-physical systems. *Software & Systems Modeling*, 18 (1), 157–189.
- [54] **Housh, M., & Ohar, Z.**, 2018, Model-based approach for cyber-physical attack detection in water distribution systems. *Water research*, 139, 132–143.
- [55] **Hu, X., Liu, S., Chen, G., & Jiang, C.**, 2013, Dependability modelling and evaluation of cyber-physical systems: A model-driven perspective. In *1st international workshop on cloud computing and information security*.
- [56] **Huang, P., Jiang, K., Guan, C., & Du, D.**, 2018, Towards modeling cyber-physical systems with SYSML/MARTE/PCCSL. In *2018 IEEE 42nd annual computer software and applications conference (COMPSAC)* (Vol. 1, pp. 264–269).
- [57] **Irisarri, E., García, M. V., Pérez, F., Estévez, E., & Marcos, M.**, 2016, A model-based approach for process monitoring in oil production industry. In *2016 IEEE 21st international conference on emerging technologies and factory automation(ETFFA)*(pp.1-4).
- [58] **Jarus, N., Sarvestani, S. S., & Hurson, A. R.**, 2016, Models, metamodels, and model transformation for cyber-physical systems. In

2016 seventh international green and sustainable computing conference (IGSC) (pp. 1–8).

- [59] **Jeon, J., Chun, I., & Kim, W.**, 2012, Metamodel-based cps modeling tool. In *Embedded and multimedia computing technology and service* (pp. 285–291). Springer.
- [60] **Jiang, Y., Song, H., Yang, Y., Liu, H., Gu, M., Guan, Y., . . . Sha, L.**, 2019, Dependable model-driven development of cps: From stateflow simulation to verified implementation. *ACM Transactions on Cyber-Physical Systems*, 3 (1), 12.
- [61] **Kacem, M. A. H., Simeu-Abazi, Z., Gascard, E., Lemasson, G., & Maisonnasse, J.**, 2017, Application of a modeling approach on a cyber-physical system” robair”. *IFAC-PapersOnLine*, 50 (1), 14230–14235.
- [62] **Kothari, S., Awadhutkar, P., Tamrawi, A., & Mathews, J.**, 2017, Modeling lessons from verifying large software systems for safety and security. In *Proceedings of the 2017 winter simulation conference* (p. 109).
- [63] **Koutsoukos, X., Kottenstette, N., Hall, J., Eyisi, E., Leblanc, H., Porter, J., & Sztipanovits, J.**, 2012, A passivity approach for model-based compositional design of networked control systems. *ACM Transactions on Embedded Computing Systems (TECS)*, 11 (4), 75.
- [64] **Koutsoumpas, V.**, 2015, A model-based approach for the specification of a virtual power plant operating in open context. In *Proceedings of the first international workshop on software engineering for smart cyber-physical systems* (pp. 26–32).
- [65] **Kuesap, J., Li, B., Satarug, S., Takeda, K., Numata, I., Na-Bangchang, K., & Shibahara, S.**, 2008, Prostaglandin d2 induces heme oxygenase-1 in human retinal pigment epithelial cells. *Biochemical and biophysical research communications*, 367(2), 413–419.
- [66] **Kumar, V. R. S., Shanmugavel, M., Ganapathy, V., & Shirinzadeh, B.**, 2015, Unified meta-modeling framework using bond graph grammars for conceptual modeling. *Robotics and Autonomous*

Systems, 72, 114–130.

- [67] **Latombe, F., Crégut, X., Combemale, B., Deantoni, J., & Pantel, M.**, 2015, Weaving concurrency in executable domain-specific modeling languages. In Proceedings of the 2015 ACM SIGPLAN international conference on software language engineering (pp. 125–136).
- [68] **Lavigne, E., Guillou, G., & Babau, J.-P.**, 2018, AVS, a model-based racing sailboat simulator: application to wind integration. IFAC-PapersOnLine, 51 (10), 88–94.
- [69] **Lezoche, M., & Panetto, H.**, 2018, Cyber-physical systems, a new formal paradigm to model redundancy and resiliency. Enterprise Information Systems, 1–22.
- [70] **Li, S., Li, D., Li, F., & Zhou, N.**, 2015, CPSICGF: A code generation framework for cps integration modeling. Microprocessors and Microsystems, 39(8), 1234–1244.
- [71] **Liu, J., & Zhang, L.**, 2011, Aspect-oriented MDA development method for non-functional properties of cyber physical systems. In 2011 second international conference on networking and distributed computing (pp. 149–153).
- [72] **Liu, X., Zhang, J., & Zhu, P.**, 2017, Modeling cyber-physical attacks based on probabilistic colored petri nets and mixed-strategy game theory. International Journal of Critical Infrastructure Protection, 16, 13–25.
- [73] **Lora, M., Fraccaroli, E., & Fummi, F.**, 2017, Virtual prototyping of smart systems through automatic abstraction and mixed-signal scheduling. In 2017 22nd ASIA and south pacific design automation conference (ASP-DAC) (pp. 232–237).
- [74] **Low, T. M., & Franchetti, F.**, 2017, High assurance code generation for cyber-physical systems. In 2017 IEEE 18th international symposium on high assurance systems engineering (HASE) (pp. 104–111).
- [75] **Lynch, K., Ramsey, R., Ball, G., Schmit, M., & Collins, K.**, 2016,

Ontology-driven metamodel validation in cyber-physical systems. In *Information technology: New generations* (pp. 1255–1258). Springer.

- [76] **Maksuti, S., Bicaku, A., Tauber, M., Palkovits-Rauter, S., Haas, S., & Delsing, J.**, 2017, Towards flexible and secure end-to-end communication in industry 4.0. In *2017 IEEE 15th international conference on industrial informatics (INDIN)* (pp. 883–888).
- [77] **Mamun, M. A. A., Berger, C., & Hansson, J.**, 2013, MDE-based sensor management and verification for a self-driving miniature vehicle. In *Proceedings of the 2013 ACM workshop on domain-specific modeling* (pp. 1–6).
- [78] **Martins, G., Bhatia, S., Koutsoukos, X., Stouffer, K., Tang, C., & Candell, R.**, 2015, Towards a systematic threat modeling approach for cyber-physical systems. In *2015 resilience week (RWS)* (pp. 1–6).
- [79] **Matinnejad, R., Nejati, S., Briand, L., & Bruckmann, T.**, 2018, Test generation and test prioritization for SIMULINK models with dynamic behavior. *IEEE Transactions on Software Engineering*.
- [80] **Merschak, S., Hehenberger, P., Witters, M., & Gadeyne, K.**, 2018, A hierarchical meta-model for the design of cyber-physical production systems. In *2018 19th international conference on research and education in mechatronics (REM)* (pp. 36–41).
- [81] **Mezhuyev, V., & Samet, R.**, 2013, Geometrical meta-metamodel for cyber-physical modelling. In *2013 international conference on cyberworlds* (pp. 89–93).
- [82] **Mordecai, Y.**, 2018, Conceptual modeling of cyber-physical gaps in air traffic control. *Procedia Computer Science*, 140, 21–28.
- [83] **Morozov, D., Lezoche, M., & Panetto, H.**, 2018, Multi-paradigm modelling of cyber-physical systems. *IFAC-PapersOnLine*, 51 (11), 1385–1390.
- [84] **Motii, A., Hamid, B., Lanusse, A., & Bruel, J.-M.**, 2016, Guiding the selection of security patterns for real-time systems. In *2016 21st*



international conference on engineering of complex computer systems (ICECCS) (pp. 155–164).

- [85] **Murugesan, A., Heimdahl, M. P., Whalen, M. W., Rayadurgam, S., Komp, J., Duan, L., . . . Lee, I.**, 2014, From requirements to code: Model based development of a medical cyber physical system. In *Software engineering in health care* (pp. 96–112). Springer.
- [86] **Nägele, T., & Hooman, J.**, 2017, Rapid construction of co-simulations of cyber-physical systems in HLA using a DSL. In *2017 43rd euromicro conference on software engineering and advanced applications (SEAA)* (pp. 247–251).
- [87] **Nannapaneni, S., Mahadevan, S., Pradhan, S., & Dubey, A.**, 2016, Towards reliability-based decision making in cyber-physical systems. In *2016 IEEE international conference on smart computing (SMARTCOMP)* (pp. 1–6).
- [88] **Navet, N., & Fejoz, L.**, 2016, CPAL: High-level abstractions for safe embedded systems. In *Proceedings of the international workshop on domain-specific modeling* (pp. 35–41).
- [89] **Neema, H., Potteiger, B., Koutsoukos, X., Karsai, G., Volgyesi, P., & Sztipanovits, J.**, 2018, Integrated simulation testbed for security and resilience of cps. In *Proceedings of the 33rd annual ACM symposium on applied computing* (pp. 368–374).
- [90] **Neema, S., Simko, G., Levendovszky, T., Porter, J., Agrawal, A., & Sztipanovits, J.**, 2014, Formalization of software models for cyber-physical systems. In *Proceedings of the 2nd FME workshop on formal methods in software engineering* (pp. 45–51).
- [91] **Nikiforova, O., El Marzouki, N., Gusarovs, K., Vangheluwe, H., Bures, T., Al- Ali, R., . . . Leon, F.**, 2017, The two-hemisphere modelling approach to the composition of cyber-physical systems. In *12th international conference on software technologies, ICSOFT 2017* (pp. 286–293).
- [92] **Ollinger, L., Wehrmeister, M. A., Pereira, C. E., & Zühlke, D.**, 2013,

An integrated concept for the model-driven engineering of distributed automation architectures on embedded systems. *IFAC Proceedings Volumes*, 46(7), 222–227.

- [93] **Orojloo, H., & Azgomi, M. A.**, 2017, A game-theoretic approach to model and quantify the security of cyber-physical systems. *Computers in Industry*, 88 ,44–57.
- [94] **Pagliari, L., Mirandola, R., & Trubiani, C.**, 2017, Multi-modeling approach to performance engineering of cyber-physical systems design. In 2017 22nd international conference on engineering of complex computer systems (ICECCS) (pp.142-145).
- [95] **Pajic, M., Jiang, Z., Lee, I., Sokolsky, O., & Mangharam, R.**, 2012, From verification to implementation: A model translation tool and a pacemaker case study. In 2012 IEEE 18th real time and embedded technology and applications symposium (pp.173-184).
- [96] **Palachi, E., Cohen, C., & Takashi, S.**, 2013, Simulation of cyber physical models using SYSML and numerical solvers. In 2013 IEEE international systems conference (SYSCON) (pp. 671–675).
- [97] **Parveen, R., Pradhan, S., & Goveas, N.**, 2018, Design and feasibility study of health related devices using cots components. In 2018 IEEE industrial cyber-physical systems (ICPS) (pp. 529–533).
- [98] **Passarini, R. F., Becker, L. B., & Farines, J.-M.**, 2013, The assisted transformation of models: Supporting cyber-physical systems design by extracting architectural aspects and operating modes from SIMULINK functional models. In 2013 III Brazilian Symposium on Computing Systems Engineering (pp. 47–52).
- [99] **Pereira, F., & Gomes, L.**, 2016, The IOPT-flow framework pairing petri nets and data-flows for embedded controller development. In IECON 2016-42nd annual conference of the IEEE industrial electronics society (pp. 4832–4837).
- [100] **Peters, J., Wille, R., Przigoda, N., Kühne, U., & Drechsler, R.**, 2015, A generic representation of CCSL time constraints for

UML/MARTE models. In Proceedings of the 52nd annual design automation conference (p. 122).

- [101] **Petnga, L., & Austin, M.**, 2016, An ontological framework for knowledge modeling and decision support in cyber-physical systems. *Advanced Engineering Informatics*, 30 (1), 77–94.
- [102] **Plasse, J., Noble, J., & Myers, K.**, 2017, An adaptive modeling framework for bivariate data streams with applications to change detection in cyber- physical systems. In 2017 IEEE international conference on data mining workshops (ICDMW) (pp.1074-1081).
- [103] **Pohlmann, U., Trsek, H., Dürkop, L., Dziwok, S., & Oestersötebier, F.**, 2014, Application of an intelligent network architecture on a cooperative cyber- physical system: An experience report. In Proceedings of the 2014 IEEE emerging technology and factory automation (ETFFA)(pp.1-6).
- [104] **Pradhan, S. M., Dubey, A., Gokhale, A., & Lehofer, M.**, 2015, CHARIOT: A domain specific language for extensible cyber-physical systems. In Proceedings of the workshop on domain-specific modeling (pp. 9–16).
- [105] **Qian, Z., & Yu, H.**, 2013, A TAOPN approach to modeling and scheduling cyber-physical systems. In 2013 international conference on information science and applications (ICISA) (pp. 1–7).
- [106] **Rashid, N., Wan, J., Quiros, G., Canedo, A., & Al Faruque, M. A.**, 2017, Modeling and simulation of cyberattacks for resilient cyber-physical systems. In 2017 13th IEEE conference on automation science and engineering (case) (pp. 988–993).
- [107] **Reijnen, F., Goorden, M., van de Mortel-Fronczak, J., & Rooda, J.**, 2017, Supervisory control synthesis for a waterway lock. In 2017 IEEE conference on control technology and applications (CCTA) (pp. 1562–1563).

- [108] **Rocchetto, M., & Tippenhauer, N. O.**, 2017, Towards formal security analysis of industrial control systems. In Proceedings of the 2017 ACM on ASIA conference on computer and communications security (pp. 114–126).
- [109] **Sanislav, T., & Mois, G.**, 2017, A dependability analysis model in the context of cyber-physical systems. In 2017 18th international carpathian control conference (ICCC) (pp. 146–150).
- [110] **Sapienza, G., Crnkovic, I., & Potena, P.**, 2014, Architectural decisions for HW/SW partitioning based on multiple extra-functional properties. In 2014 IEEE/IFIP conference on software architecture (pp. 175–184).
- [111] **Seiger, R., Keller, C., Niebling, F., & Schlegel, T.**, 2015, Modelling complex and flexible processes for smart cyber-physical environments. *Journal of Computational Science*, 10, 137–148.
- [112] **Silva, L., Almeida, H., Perkusich, A., & Perkusich, M.**, 2015, A model-based approach to support validation of medical cyber-physical systems. *Sensors*, 15 (11), 27625–27670.
- [113] **Silva, L. C., Perkusich, M., Bublitz, F. M., Almeida, H. O., & Perkusich, A.**, 2014, A model-based architecture for testing medical cyber-physical systems. In Proceedings of the 29th annual ACM symposium on applied computing (pp. 25–30).
- [114] **Simko, G., Lindecker, D., Levendovszky, T., Jackson, E., Neema, S., & Szti-panovits, J.**, 2013, A framework for unambiguous and extensible specification of DSMLS for cyber-physical systems. In 2013 20th IEEE international conference and workshops on engineering of computer based systems (ECBS) (pp. 30–39).
- [115] **Sinha, R., Pang, C., Mart´mez, G. S., Kuronen, J., & Vyatkin, V.**, 2015, Requirements-aided automatic test case generation for industrial cyber-physical systems. In 2015 20th international conference on engineering of complex computer systems (ICECCS) (pp. 198–201).
- [116] **Son, H. S., Kim, W. Y., Kim, R. Y., & Min, H.-G.**, 2012, Metamodel design for model transformation from SIMULINK to ECML in cyber

physical systems. In Computer applications for graphics, grid computing, and industrial environment (pp. 56–60). Springer.

- [117] **Tan, F., Wang, Y., Wang, Q., Bu, L., Zheng, R., & Suri, N.**, 2013, Guaranteeing proper-temporal-embedding safety rules in wireless cps: A hybrid formal modeling approach. In 2013 43rd annual IEEE/IFIP international conference on dependable systems and networks (DSN) (pp. 1–12).
- [118] **Tariq, M. U., Florence, J., & Wolf, M.**, 2014, Design specification of cyber- physical systems: Towards a domain-specific modeling language based on SIMULINK, eclipse modeling framework, and giotto. In ACESMB@ models (pp. 6–15).
- [119] **Tariq, M. U., Nasir, H. A., Muhammad, A., & Wolf, M.**, 2012, Model-driven performance analysis of large scale irrigation networks. In 2012 IEEE/ACM third international conference on cyber-physical systems (pp. 151–160).
- [120] **Thramboulidis, K., Kontou, I., & Vachtsevanou, D. C.**, 2018, Towards an IOT-based framework for evolvable assembly systems. IFAC-PapersOnLine, 51 (11), 182–187.
- [121] **Thramboulidis, K., Vachtsevanou, D. C., & Solanos, A.**, 2018, Cyber-physical microservices: An IOT-based framework for manufacturing systems. In 2018 IEEE industrial cyber-physical systems (ICPS) (pp. 232–239).
- [122] **Tundis, A., Egert, R., & Mühlhäuser, M.**, 2017, Applying a properties modeling approach for monitoring smart grids. In 2017 IEEE 14th international conference on networking, sensing and control (ICNSC) (pp. 714–719).
- [123] **Tuo, M., Zhou, X., Yang, G., & Fu, N.**, 2016, An approach for safety analysis of cyber-physical system based on model transformation. In 2016 IEEE international conference on internet of things (ITHINGS) and IEEE green computing and communications (GREENCOM) and IEEE cyber, physical and social computing (CPSCOM) and IEEE smart data (SMARTDATA) (pp. 636–639).

- [124] **Van Acker, B., Denil, J., Vangheluwe, H., & De Meulenaere, P.**, 2015, Managing heterogeneity in model-based systems engineering of cyber-physical systems. In 2015 10th international conference on p2p, parallel, grid, cloud and internet computing (3pgcic) (pp. 617–622).
- [125] **Van Mortel-Fronczak, J. M., van der Heijden, M. H., Huisman, R. G., & Reniers, M. A.**, 2014, Supervisor synthesis in model-based automotive systems engineering. In *Iccps'14: ACM/IEEE 5th international conference on cyber- physical systems (with cps week 2014)* (pp. 187–198).
- [126] **Walch, M.**, 2017, Knowledge-driven enrichment of cyber-physical systems for industrial applications using the KBR modelling approach. In 2017 IEEE international conference on agents (ICA) (pp. 84–89).
- [127] **Weissnegger, R., Schuss, M., Kreiner, C., Pistauer, M., Römer, K., & Steger, C.**, 2016, Simulation-based verification of automotive safety-critical systems based on EAST-ADL. *Procedia computer science*, 83, 245–252.
- [128] **Whalen, M. W., Murugesan, A., Rayadurgam, S., & Heimdahl, M. P.**, 2014, Structuring SIMULINK models for verification and reuse. In *Proceedings of the 6th international workshop on modeling in software engineering* (pp.19–24).
- [129] **Whitsitt, S., Sprinkle, J., & Lysecky, R.**, 2014, Generating model transformations for mending dynamic constraint violations in cyber physical systems. In *Proceedings of the 14th workshop on domain-specific modeling* (pp. 35– 40).
- [130] **Xin, S., Guo, Q., Sun, H., Zhang, B., Wang, J., & Chen, C.**, 2015, Cyber-physical modeling and cyber-contingency assessment of hierarchical control systems. *IEEE Transactions on Smart Grid*, 6 (5), 2375–2385.
- [131] **Yan, W., Xue, Y., Li, X., Weng, J., Busch, T., & Sztipanovits, J.**, 2012, Integrated simulation and emulation platform for cyber-physical system security experimentation. In *Proceedings of the 1st international conference on high confidence networked systems* (pp. 81–88).

- [132] **Zhang, K., & Sprinkle, J.**, 2013, Model-based software synthesis for self-reconfigurable sensor network in water monitoring. In 2013 20th IEEE international conference and workshops on engineering of computer-based systems (ECBS) (pp. 40–48).
- [133] **Zhang, K., & Sprinkle, J.**, 2014, A closed-loop model-based design approach based on automatic verification and transformation. In Proceedings of the 14th workshop on domain-specific modeling (pp. 1–6).
- [134] **Zhang, L.**, 2011, Aspect-oriented MDA approach for non-functional properties of distributed cyber physical systems. In 2011 10th international symposium on distributed computing and applications to business, engineering and science (pp. 284–288).
- [135] **Zhang, L.**, 2013a, An integration approach to specify and model automotive cyber physical systems. In 2013 international conference on connected vehicles and expo(ICCVE)(pp.568-573).
- [136] **Zhang, L.**, 2013b, View oriented approach to specify and model aerospace cyber-physical systems. In 2013 IEEE 11th international conference on dependable, autonomic and secure computing (pp. 296–303).
- [137] **Zhang, L.**, 2014, A framework to specify big data driven complex cyber physical control systems. In 2014 IEEE international conference on information and automation (ICIA) (pp. 548–553).
- [138] **Zhang, L., & Feng, S.**, 2014, Integration design and model transformation for cyber physical systems. In 2014 IEEE 5th international conference on software engineering and service science (pp. 754–757).
- [139] **Zhao, H., Apvrille, L., & Mallet, F.**, 2017, Multi-view design for cyber-physical systems.
- [140] **Zhou, Y., Gong, X., Li, J., & Li, B.**, 2018, Verifying cps for self-adaptability. In 2018 IEEE/ACIS 17th international conference on computer and information science (ICIS) (pp. 166–172).

Appendix 1: modeling activities, employed modeling approaches in each activity and their corresponding studies.

<b>Activity</b>	<b>Approach</b>	<b>Relevant studies</b>
Development	DSL	[5, 7, 132, 63, 118, 20, 104, 47]
	Metamodeling	[67, 31, 32, 81, 23, 11, 66, 30]
	Model-based approach	[41, 19, 24, 125, 107, 98, 85]
	Component based approach	[91, 70, 103]
	UML	[120]
	equation based	[81, 58]
	petri nets	[99]
	Integrated approaches	[138, 18]
	architecture based approach	[92, 43]
	Aspect-Oriented approach	[71]
	Process based modeling	[44]
	text based modeling	[88]
Other	[121, 99]	
Analysis	DSL	[119, 4, 34]
	Metamodeling	[110, 32, 21, 83, 139, 69]
	Model-based approach	[35, 24, 82, 123, 87, 37]
	Component based approach	[110, 61]
	UML	[36, 34, 8, 28]
	equation based	[119, 105, 130]
	petri nets	[105, 72, 17, 52, 55]
	Simulation	[73, 89]
	Integrated approaches	[136, 135, 42, 94]
	Aspect-Oriented approach	[105]
	Other	[129, 3, 52]
V&V	DSL	[5, 77, 40, 133, 13]
	Model-based approach	[95, 62, 123, 128, 33]
	Component based approach	[97, 113]
	Simulation	[112, 79, 60]
	UML	[140]
	equation based	[112, 102]
	petri nets	[48]
	Ontology based	[75, 25]
	architecture based approach	[49]
	Process based modeling	[45]
Other	[133, 90]	
Security	Metamodeling	[2, 78, 76]
	Model-based approach	[22, 9]
	Component based approach	[50, 53]
	Simulation	[54, 60, 106]
	UML	[39, 84]
	equation based	[93]
petri nets	[72, 26]	



	Integrated approaches	[65]
	pattern-based modeling	[84, 117]
	text based modeling	[108]
	Other	[131]
Simulation	DSL	[86, 63, 10, 114]
	Metamodeling	[80, 29]
	Model-based approach	[95]
	Component based approach	[137, 68]
	Simulation	[80, 60, 68, 127, 124, 12, 96]
	UML	[56]
	equation based	[114]
	Integrated approaches	[18]
Monitoring	DSL	[77]
	Metamodeling	[23, 14, 57]
	Model-based approach	[22]
	Process based modeling	[14]
	Other	[51, 122]
Time	UML	[100]
	text based modeling	[88]
	Aspect-Oriented approach	[134]
	Process based modeling	[45]
Adaptability	Metamodeling	[76]
	component based approach	[38]
	equation based	[102]
Correctness	component based approach	[46, 16]
	equation based	[74]
Integration	Integrated approaches	[136]
	model-based approach	[125]
	Other	[129]
Efficiency	Ontology based	[75]
	component based approach	[46]
Flexibility	model-based approach	[15]
	component based approach	[111]
Failure identification	UML	[1]
	Ontology based	[1]
Uncertainty	Other	[27, 64]
Complexity	Metamodeling	[111]
dependability	UML	[109]
performance	model-based approach	[22]
Reliability	Simulation	[54]
resilience	Other	[122]
self-assessment	model-based approach	[22]
Test case Generation	model-based approach	[115]
contingency assessment	equation based	[130]
Fault tolerance	equation based	[102]

Decision making support	Ontology based	[101]
-------------------------	----------------	-------

Appendix 2: Used languages/frameworks for developing DSMLs, Metamodels, or tools

Type	Used tool/language	Relevant studies	
DSML	EMF	[5, 77, 118]	
	Acceleo	[77]	
	GEMOC	[47]	
	GME	[119, 132, 63, 10, 114, 133, 4]	
	GReAT	[10, 114, 133]	
	Kermeta	[47]	
	OCL	[5, 77]	
	Papyrus	[7]	
	Simulink	[118]	
	Xtext	[86, 104]	
	WebGME	[20]	
	Sirius	[47]	
	Metamodel	EMF	[116, 2, 15, 56, 111, 139]
		ADOxx	[76]
GME		[78]	
GEMOC		[56]	
KMF		[51]	
ISA95 and ISA88 based		[57]	
UML		[76, 31, 32, 120, 83, 134, 69, 30, 1]	
SysML		[25, 11]	
Xtext		[56]	
Sirius		[56]	
-		[140, 131]	
Extension	UML	[36]	
	BPMN 2.0	[44]	
	-	[14, 53]	
Tool	EMF	[68]	
	Sirius	[68]	
	Acceleo	[68]	
	Python	[6]	
	ADOxx	[126]	
	OSATE	[41, 42]	
	oAW	[19]	
	Java	[60, 115]	
	MetaEdit+	[23]	
	Papyrus	[127]	
	VC++.NET	[70]	
	UML	[98]	
	-	[59, 95, 122, 45, 37, 66]	