A systematic literature review of the use of formal methods in medical software systems: SLR of the use of Formal Methods in Medical Software Systems

Article *in* Journal of Software: Evolution and Process · February 2018

DOI: 10.1002/smr.1943

Silvia Bonfanti
University of Bergamo
52 PUBLICATIONS **419** CITATIONS
SEE PROFILE

Angelo Gargantini
University of Bergamo
200 PUBLICATIONS **3,278** CITATIONS
SEE PROFILE

Atif Mashkoor
Johannes Kepler University Linz
126 PUBLICATIONS **960** CITATIONS
SEE PROFILE

JOURNAL OF SOFTWARE: EVOLUTION AND PROCESS
J. Softw. Evol. and Proc. 2017; **00**:1–23
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/smr

A Systematic Literature Review of the use of Formal Methods in Medical Software Systems[*]

Silvia Bonfanti[*], Angelo Gargantini[1], Atif Mashkoor[2]

[*] *University of Bergamo, silvia.bonfanti@unibg.it* [1] *University of Bergamo, angelo.gargantini@unibg.it* [2] *Software Competence Center Hagenberg GmbH, atif.mashkoor@scch.at*

SUMMARY

The use of formal methods is often recommended to guarantee the provision of necessary services and to assess the correctness of critical properties, such as functional safety, cybersecurity and reliability, in medical and health-care devices. In the past, several formal and rigorous methods have been proposed and consequently applied for trustworthy development of medical software and systems. In this paper, we perform a systematic literature review on the available state of the art in this domain. We collect the relevant literature on the use of formal methods for modeling, design, development, verification and validation of software-intensive medical systems. We apply standard systematic literature review techniques and run several queries in well-known repositories in order to obtain information that can be useful for people who are either already working in this field or planning to start. Our study covers both quantitative and qualitative aspects of the subject. Copyright © 2017 John Wiley & Sons, Ltd.

Received ...

KEY WORDS: Formal Methods, Systematic Literature Review, Medical Device Software

1. INTRODUCTION

In modern medical devices, human safety depends upon the correct operation of software controlling the device: software malfunctioning can cause injuries to, or even the death of, patients. A crucial issue is how to guarantee that medical software has all the qualities (e.g., safety, security and dependability) expected for critical components. One way to improve and assess software quality, as suggested by literature [1–3], is to use formal methods or in general rigorous methods for design, validation, and verification of medical software. Some processes for improvement of medical standards, based on formal approaches, have already been proposed [4–6], although their adoption in industrial applications is rather limited.

The overall aim of this paper is twofold: 1) to provide guidance to researchers starting to work on this topic, and 2) to assess the state of the art which is more useful for researchers already working on this subject. We have applied a Systematic Literature Review (SLR) process to the topic of rigorous methods for designing and validation of medical software and systems by following the guidelines presented in [7–9] with slight improvements such as a wider range of repositories is queried for information retrieval and the subject is covered from both quantitative and qualitative aspects. Through this analysis, we give an overview of the research literature about formal methods

[*] The research reported in this paper has been partly supported by the Austrian Ministry for Transport, Innovation and Technology, the Federal Ministry of Science, Research and Economy, and the Province of Upper Austria in the frame of the COMET center SCCH.

[*] Correspondence to: University of Bergamo, viale Marconi 5, Bergamo, IT silvia.bonfanti@unibg.it

applications to model, to verify, and to validate medical systems. Moreover, we include processes and tools that translate models written using formal languages in machine code. We would like to underline that the SLR carried out in this work considers only formal methods applied to medical device/software. Other processes applied to medical device/software are outside of our goal (e.g., code implementation, testing [8], and hardware configuration).

The goals of our SLR are (1) to gather a sufficient number of relevant articles, (2) to perform a series of analyses, and (3) to publish results of findings to allow researchers to browse in the collected data. This activity follows a systematic process to avoid possible biases in order to include as much information as possible, but at the same time capable of identifying only relevant papers.

This paper extends our previous work [10] in several directions: we consider a larger amount of repositories during the collection of papers, we implement an improved and strengthened filtering process to exclude papers not related to the topic under investigation, and we integrate the quantitative analysis, in addition to a qualitative analysis, in terms of notations used, case studies and activities performed.

The article is organized as follows: in Section 2, we explain the SLR process we applied in order to collect and analyze papers. In Section 3, we show the data sources we use. In Section 4, we perform an analysis of data by answering a set of research questions (RQ1 to RQ7) in order to extract useful information. In Section 5, we discuss some limitations of the SLR process. The article is concluded in Section 6.

2. THE SLR PROCESS

A SLR is the review of a clearly formulated question that uses systematic and explicit methods to identify, select, appraise relevant research, collect, and analyze data from the studies that are included in the review [11]. A SLR defines a precise process for literature review: criteria for inclusion and exclusion are explicitly stated and therefore it provides a transparent and repeatable selection process. The final aim is to minimize bias and increase objectivity of the review outcomes.

Figure 1 shows our devised SLR process, it is divided into two main steps:

- `Papers collection`: collect papers for the SLR analysis;
- `Classification and analysis`: classify and analyze papers obtained in the first step.

The final goal of this process is to provide a list of papers that fit the topic of the SLR and several analyses over the selected papers. The first step consists in the definition of terms used to perform queries on different databases. Once the user runs queries on databases (see Section 3 for further details on databases), the results can be saved in different file formats. We have chosen the `.bib` file type, which is very often used for the bibliography, it adopts a standardized format, and it is easily processable with tools and own scripts since it is a textual file. After that, all obtained `.bib` files are merged into the `Merged.bib` file. The user checks whether all already known papers (a list of papers already known to the user about the topic) are included into the file `Merged.bib`. If not, s/he has to identify terms from omitted papers and rerun queries to also include them. Despite the fact that the list of known papers is manually selected based on user's previous experience, the chances of introducing a possible bias is minimal. This is because the set of previously known papers is only used for checking the completeness of the query and for the analysis we consider all the papers resulting from the search.

Then the classification and analysis phase starts. First, the pre-processing activity is performed (e.g., delete duplicates and unrelated papers). The result of pre-processing is the `Final.bib` file ready for the analysis. Then `Final.bib` is translated into the `.xml` file using our own `bib2xml` script. Starting from the `.xml` file, we perform QUANTITATIVE ANALYSIS (see Section 4.1) to automatically extract a set of statistical information. Quantitative analysis has a low degree of human interaction because only fields in the bibliographic entries are used (e.g., year, type of publication, and number of citations). We use our own scripts written using EXtensible Stylesheet Language Transformation (`.xslt`), and results are saved into `.txt` or `.csv` files. After that, results are

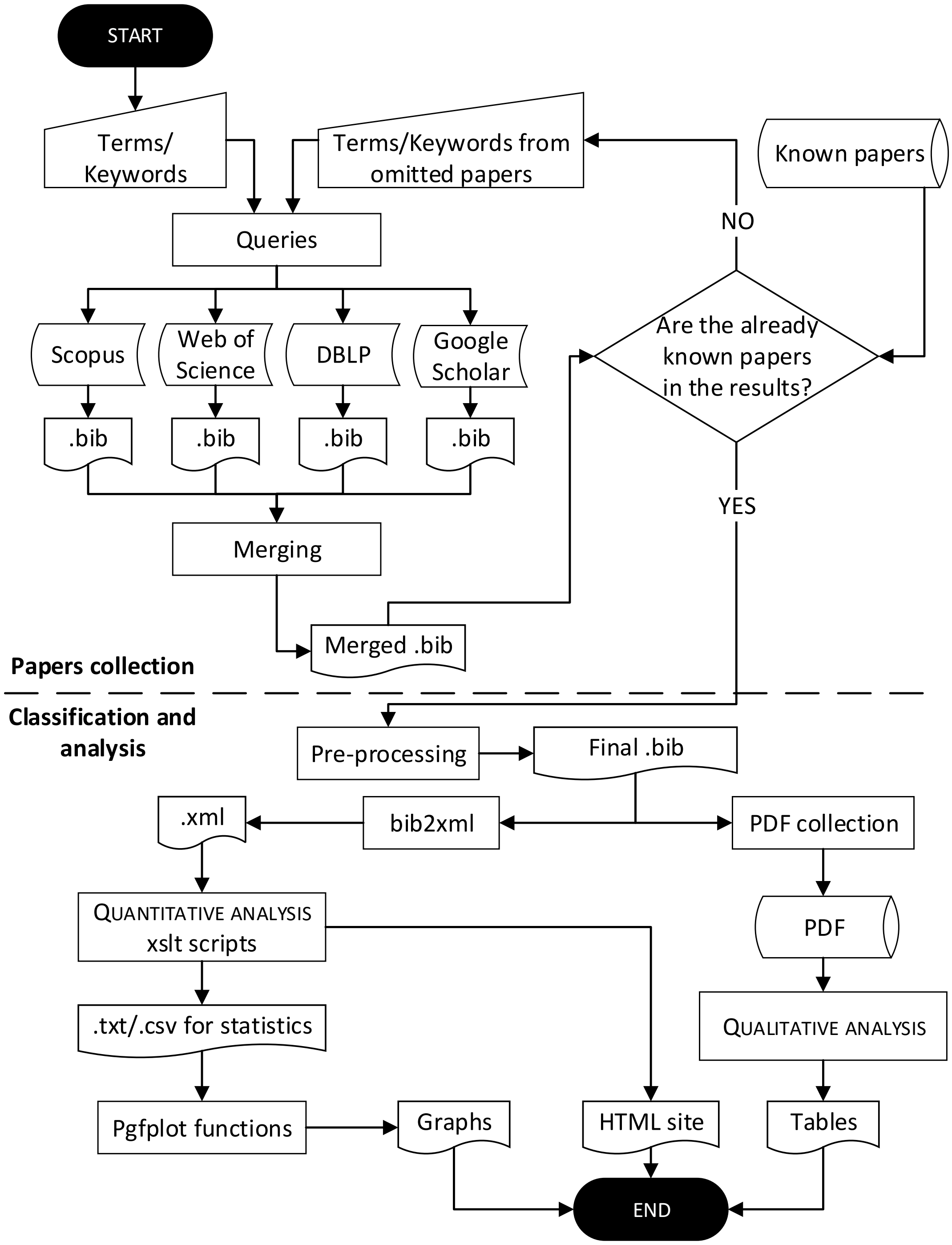

Figure 1. SLR Process

plotted using `pgfplot` package provided by LaTeX. The quantitative analysis results are statistical data obtained using mathematical measurements and calculations.

Starting from `Final.bib`, we collect PDF files of resulted papers and complete a QUALITATIVE ANALYSIS to provide more technical details (e.g., the case study analyzed and the notation used). Qualitative analysis (see Section 4.2) requires user interaction since the content of the papers must be analyzed.

DOI: 10.1002/smr

Furthermore, starting from the `.xml` file, we generate a `HTML` website by using the `.xslt` script. The website contains all publications listed in the `Final.bib` file and is available on-line[‡].

3. PAPERS COLLECTION

During a preliminary analysis [10], we used only Scopus (http://www.scopus.com), but we missed some relevant papers published in journals and in proceedings not included in the Scopus database. For this reason, now we consider other repositories besides Scopus, i.e., Web of Science (https://apps.webofknowledge.com), DBLP (http://dblp.uni-trier.de/) and Google Scholar (https://scholar.google.com/). We do not consider IEEE Xplore (http://ieeexplore.ieee.org/), Springer (http://www.springer.com), and ACM Digital Library (http://dl.acm.org/) because they are already included in Scopus and DBLP. Furthermore, we noticed that the search function of IEEE Xplore has some limitations: during the search activity some queries returned unrelated results which did not contain the terms entered in the query.

Scopus is the largest database of scientific publications owned by Elsevier, it contains scientific journals, books and conference proceedings. It encloses more than 60 million records, over 21,500 peer-reviewed journals, over 360 trade publications, 7.2 million conference papers, 27 million patents, 5,000 articles-in-press from international publishers including Cambridge University Press, IEEE, Nature Publishing Group, Springer, ACM Digital Library and Wiley. It includes more than 113,000 books. Thomson Reuters maintains **Web of Science** that provides a citation search. It gives access to several databases in different disciplinary researches: Conference Proceedings Citation Index (covers more than 160,000 conference titles), Science Citation Index Expanded (covers more than 8,500 journals encompassing 150 disciplines), Social Sciences Citation Index (covers more than 3,000 journals in social science), Arts & Humanities Citation Index (covers more than 1,700 arts and humanities journals), Book Citation Index (covers more than 60,000 books). **DBLP** provides open bibliographic information of major computer science journals and proceedings. It contains more than 3 million publications relating to more than 4,500 conferences and more than 1,400 journals. This service is provided by University of Trier and Schloss Dagstuhl. **Google Scholar** furnishes a way to search literature and it contains a large quantity of documents. It is possible to search across many sources: articles, theses, books, abstracts, court opinions, academic publishers, professional societies, on-line repositories, universities, and patents.

Given the repositories listed above, we have chosen a set of terms based on the goal of the SLR to perform queries[§]. The queries are a combination of terms using “and” and “or” logical operators. The majority of queries perform the search only in titles, while few of them are executed on author keywords and abstracts. This limitation is due to the search functions provided by the repositories, as shown below. Due to the intrinsic limited functionality, the number of queries run for each repository is not the same.

Scopus has the most performing search system. It allows to execute an advanced search, e.g., by title (TITLE), by author keywords (AUTHKEY), by abstract (ABS), or by their discrete combination. Furthermore, it allows to search terms that start with a prefix or end with a suffix by using ‘*’ symbol (e.g., “method*” means “method, methods, methodology, methodologies”).

Web of Science is similar to Scopus but it does not allow to perform an advanced search, e.g., by keywords or by author keywords.

DBLP has a limited search technology. It is not possible to specify the search field (e.g., title, keywords) and it is not allowed to perform the words search by suffix or prefix.

Google Scholar provides two search options: the first limits the search to the title, the second extends the search anywhere in the documents (title, content, keywords, authors) at the same

[‡] The list of publications is available at http://cs.unibg.it/bonfanti/FMMedicalDeviceSLR/
[§] The list of queries is available at http://cs.unibg.it/bonfanti/FMMedicalDeviceSLR/listquery.html

time. The latter option introduces many papers out of scope, so only the first one is considered. Furthermore, Google Scholar allows to search for specific authors, specific publication venues, and time intervals. We do not use these types of searches because we do not know in advance the data we should look for.

Regarding the terms and keywords, we have identified two subsets of possible terms based on our experience. The two subsets of terms are combined in the searches: the first one refers to the use of formal techniques and the second one refers to the application field; the two sets are displayed in Table I.

Approaches	**Application fields**	
Formal	Medical device/s	Syringe pump
Formal method/s	Medical software	ECG
Formal specification/s	Medical system/s	e-health
Formal modeling/ modelling	Medical pump	Medical communication protocol
Formal validation	Hemodialysis	Stereoacuity test
Formal code generator/ generation	Infusion pump	Pulse oximeter
Formal certification	Pacemaker	Imatinibdose

Table I. Terms used in the queries

We have combined all the terms in the first column with all the terms in the second column. Moreover, while "Formal" was searched only in the title, the others were searched in TITLE, ABS, and AUTHKEY, if the search engine provides this kind of search. For instance, in SCOPUS, we have performed the following queries:

TITLE = (formal) AND TITLE-ABS-AUTHKEY = (infusion pump)
TITLE-ABS-AUTHKEY = (formal validation) AND TITLE-ABS-AUTHKEY = (hemodialysis)

After the execution of queries in the databases, we found 359 papers including duplicates because some papers fit in more than one query, either in the same repository or the other.

By following the process defined in Figure 1, the next step consists of exporting all papers found in the repositories and merging them in the `Merged.bib` file.

4. CLASSIFICATION AND ANALYSIS

Before performing classification, we make pre-processing activities that consist of the following steps:

1. conform authors names using the format (`first name initial, surname`),
2. update authors names: add missing accents, e.g., ö, *é*, *ś*,
3. delete duplicates,
4. delete unrelated papers.

The first activity is automatically performed using our own script. The script analyses the author field and adjust names and surnames considering the target format. We perform the second activity manually, because we did not find any process to make it automatic. The third activity is performed using JabRef tool[¶]. The tool has a function to find duplicates, after a duplicate is identified the user

[¶] `http://www.jabref.org/`

can choose how to handle it. The possible solutions are: keep one of them, keep both, or keep the merged entry only. The last activity, deleting unrelated papers and keeping only those of interest, is the most time-consuming because it requires involvement of the one who is performing the SLR. As this may introduce some bias, for this reason we have precisely defined the following main inclusion/exclusion criteria:

1. **formal methods**: The main focus is on formal methods or formal techniques like modeling, verification, and so on. The activity must include the use of a formal (abstract) notation and present, or at least enable, some form of rigorous analysis. For this reason, papers presenting semi-formal approaches are discarded. We classify semi-formal notations, those that do not have precise (mathematical/logical) semantics, e.g., UML. Also papers that apply other methodologies (like agile or test-driven development) to medical software are discarded. Papers using verification or model checking directly applied to code are not considered because we require a part of modeling in the process.

2. **medical software**: To be included, a paper must focus primarily on medical software or systems. Each paper must explicitly deal with or pay particular attention to a concrete medical application, software, system, or device. If the paper does not include examples specific to medical systems, it is discarded.

We analyze all the papers by reading their abstracts and over-viewing their content in ambiguous cases. We delete all papers that do not describe the application of formal methods to a medical device/software. Moreover, we remove all documents that are not included in proceedings of peer-reviewed conferences/workshops/journals in order to maintain the number of documents manageable and the quality of the collection high. The result of pre-processing activity is the `Final.bib` file. Starting from this file, we perform two types of analysis: Quantitative Analysis and Qualitative Analysis.

Several analyses we perform in the following take as input the number of citations of papers. Several repositories provide citation data (with different degree of completeness and accuracy), and it is difficult to merge such data coming from different sources. Moreover, repositories like Google Scholar may be easily manipulated [12], fail to correctly identify authors, include "grey" publications, or include duplicate citations [13]. For these reasons, we decided to use only citations given by Scopus, which offers a very good compromise between completeness and reliability [13]. Citations in Scopus are subject to change over time since they are constantly updated by Scopus. In this paper, we consider the citations retrieved on July 15th, 2017.

4.1. Quantitative analysis

Quantitative analysis has a low degree of human interaction and we performed it by using own scripts written in EXtensible Stylesheet Language Transformation (.xslt). For this analysis, we consider the number of publications written over the years, the type of publications, the number of papers written by authors and the number of citations. We answer the set of research questions as shown below.

RQ1: Which is the trend of publications? As a first question, we want to observe the trend of publications about formal methods applied in the medical field. We analyze the number of publications from 1992 (the year of the oldest publication we found) until 2016. As shown in Figure 2, until 2004 only five publications have been published. Starting from 2004, the interest on application of formal methods to medical devices increases. From 2010, the behavior of the number of papers shows a progressive increase until 2016[||]. The overall behavior of the graph shows an increasing interest in this topic by the community mostly in the last years.

[||] The search is performed in July 2017, we did not consider the 2017 in the graphs because the year is not finished yet.

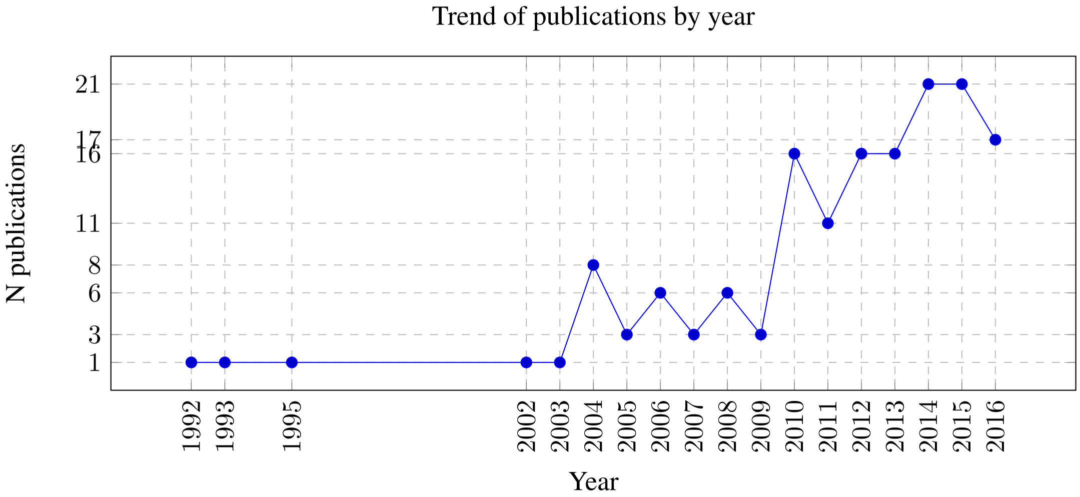

Figure 2. RQ1: Trend of publications

RQ2: Which is the trend of publications considering the type? In Figure 3, the pie chart shows the percentage of publications grouped by type. Note that from the databases we obtain only journals and conference papers (InProceedings). The number of publications in proceedings (approx. 71%) is greater than the number of publications in journals (approx. 29%).

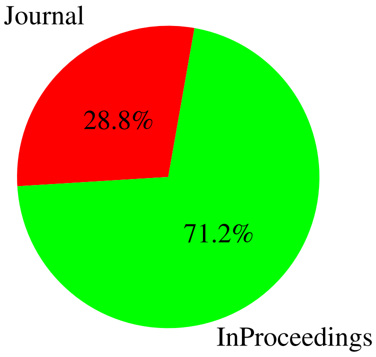

Figure 3. RQ2: Types of publications

Figure 4 depicts the trend of the number of publications in journals and in proceedings. In the last few years, we notice an increase in number of publications in conferences. This trend can be affected by different factors: - the subject is still rather novel and there is an increasing interest of the computer science community towards this topic; - usually it is easier to publish in conferences than in journals; - the publication in conference is preferred because it provides the opportunity to expose the research to a wider audience and to have an immediate feedback by the experts in the field.

RQ3: How many papers about this topic have been written by the same author? Figure 5 shows the number of publications per author. The most apparent observation is that the majority of authors (approx. 69%) have published only once about this topic, 18.5% of authors have two publications and 5.7% of authors have three publications. Only 6.8% of authors have more than three publications. An explanation could be that this topic is rather new in the scientific community and authors are starting their activities in these years. We expect an increase in number of publications per author in upcoming years due to ongoing scientific studies.

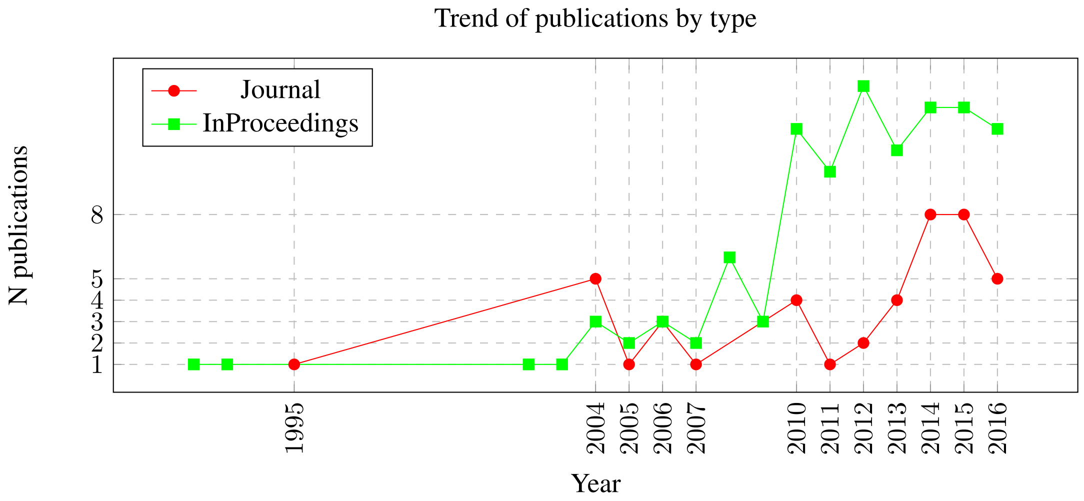

Figure 4. Publications in Journal/Proceedings per year

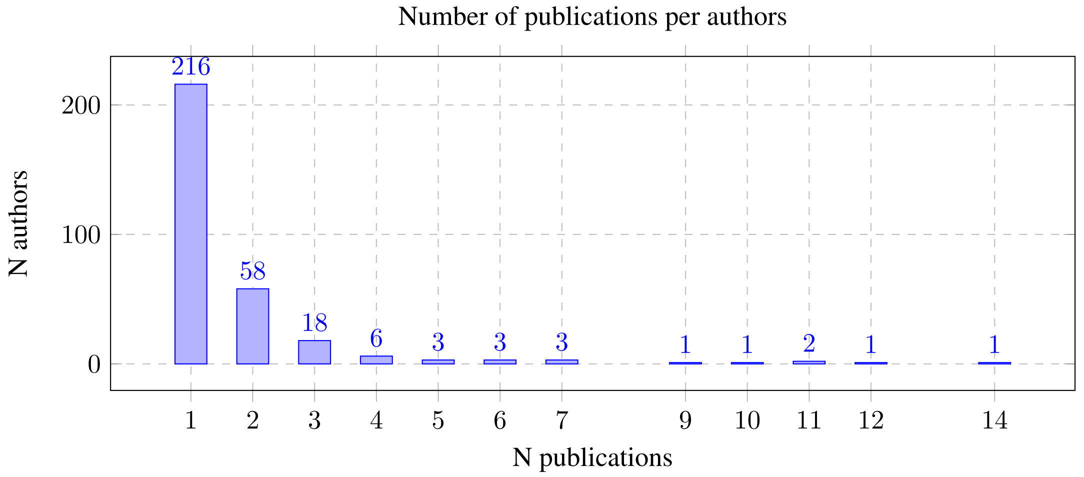

Figure 5. RQ3: Number of papers by the same author

RQ4: Which are the most cited publications? Before introducing which are the most cited papers, we analyzed the general behavior of the number of citations (see Fig. 6). Overall, approx. 33.7% of publications do not have citations. Approx. 47.4% of publications have less than ten citations and approx. 18.9% of publications have more than ten citations. The low number of citations could be due to the interest in novel verification methods rather than application of existing methods to specific case studies. Moreover, Scopus could introduce a delay in the collection of the citations.

Table II shows the most cited publications by considering only the citations given by Scopus. The publication with most citations [14] presents the utilization of formal methods in the improvement of medical protocols. A new formal language and theorem prover have been defined to help medical experts in medical protocol definition. In the second paper [15], the authors present a framework for formal verification of a real-time extension of UML statecharts and present its application to the pacemaker example. The paper number three [16] describes a closed-loop testing environment that, given the set of requirements of the pacemaker, produces a set of test cases. The fourth paper [17] highlights the importance of formal methods used in pre-market and post-market analysis for medical devices. Paper number five [18] shows how to derive the timed-automata model of the

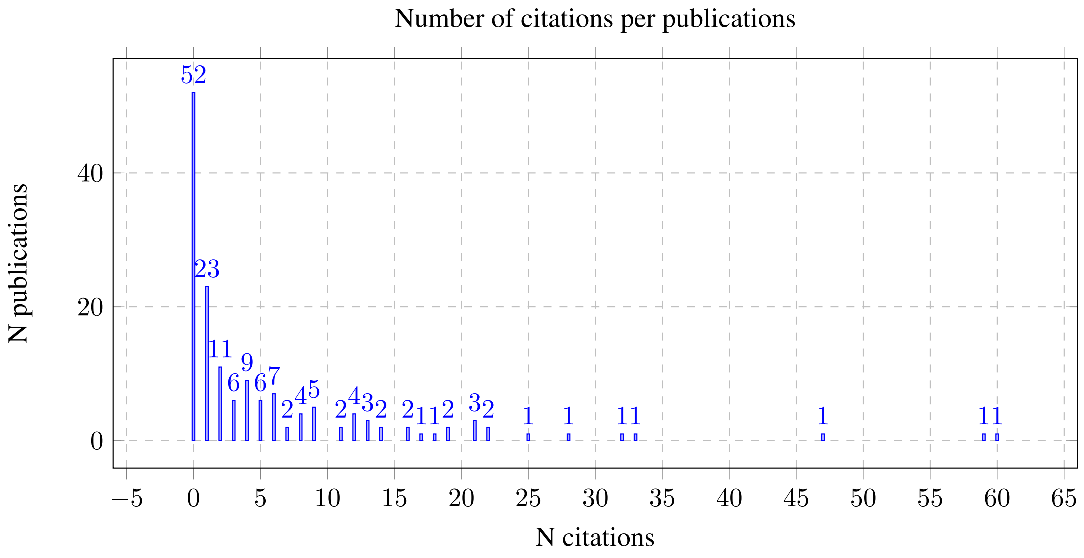

Figure 6. RQ4: Number of citations/publication

heart. In paper number six [19], the tool UPP2SF is presented. The tool translates UPPAAL models into Simulink/StateFlow models to take advantage of the automatic code generator available in Simulink. Again, the case study is the pacemaker. Paper number seven [20] applies formal methods to infusion pumps. The authors model the system using Extended Finite State Machines (EFSM) and apply validation and verification techniques using the UPPAAL framework. Furthermore, they generate test cases from the formal specification of the system to test the machine code. Also paper eight [21] models the infusion pump using EFSM and applies verification technique using the Software Cost Reduction (SRC) tool. In paper number nine [22], measurement-based timing analysis is used to guarantee timing properties in implementation as well as in the formal model. When timing properties may be violated in the implementation due to timing delay, it is suggested to measure the time deviation and reflect it to the code explicitly by modifying guards. The case study is the pacemaker.

4.2. Qualitative analysis

Qualitative analysis has the main goal of identifying the topics of interest and for which activities formal methods are more often used. It cannot be carried out by automatic scripts like quantitative analysis. The publications were manually analyzed to extract the following information:

- the notations used (e.g., automata, state charts, and abstract state machines);
- the case studies analyzed (e.g., pacemaker and infusion pump);
- the target of the application (e.g., model verification, model validation, and certification);
- the methodologies applied (e.g., modeling, refinement, verification, and conformance checking).
- the tools used (e.g., MATLAB, SPIN, Asmeta, and Rodin);

RQ5: Which are the notations used? In Table III, the notations used in resulted papers are classified into five macro categories:

- Logic: the notation is based on a logical language that consists of logical symbols and is characterized by having a fixed interpretation. The combination of these symbols compose well-formed formulas.
- State Based: in a state-based approach, an execution of a system is viewed as a sequence of states, where a state is an assignment of values to some set of components [23].
- Event Based: an event-based approach views an execution as a sequence of events [23].

N	Publications	# cit
1	Ten Teije A, Marcos M, Balser M, Van Croonenborg J, Duelli C, Van Harmelen F, Lucas P, Miksch S, Reif W, Rosenbrand K, *et al.*. Improving medical protocols by formal methods. *Artificial Intelligence in Medicine* 2006; **36**(3):193–209, doi: 10.1016/j.artmed.2005.10.006	60
2	David A, Oliver Möller M, Yi W. Formal verification of UML statecharts with real-time extensions. *Fundamental Approaches to Software Engineering*, *Lecture Notes in Computer Science*, vol. 2306, Springer Berlin Heidelberg, 2002; 218–232, doi: 10.1007/3-540-45923-5_15	59
3	Jiang Z, Pajic M, Mangharam R. Cyber-physical modeling of implantable cardiac medical devices. *Proceedings of the IEEE* jan 2012; **100**(1):122–137, doi: 10.1109/JPROC.2011.2161241	47
4	Jetley R, Iyer S, Jones P. A formal methods approach to medical device review. *Computer* 2006; **39**(4):61–67, doi: 10.1109/MC.2006.113	33
5	Jiang Z, Pajic M, Connolly A, Dixit S, Mangharam R. Real-time heart model for implantable cardiac device validation and verification. *Proceedings - Euromicro Conference on Real-Time Systems*, Brussels, Belgium, 2010; 239–248, doi: 10.1109/ECRTS.2010.36	32
6	Pajic M, Jiang Z, Lee I, Sokolsky O, Mangharam R. From verification to implementation: A model translation tool and a pacemaker case study. *Real-Time Technology and Applications - Proceedings*, IEEE: Beijing, China, 2012; 173–184, doi: 10.1109/RTAS.2012.25	28
7	Arney D, Jetley R, Jones P, Lee I, Sokolsky O. Formal methods based development of a PCA infusion pump reference model: Generic infusion pump (GIP) project. *High Confidence Medical Devices, Software, and Systems and Medical Device Plug-and-Play Interoperability*, IEEE: Boston, MA, USA, 2007; 23–33, doi: 10.1109/HCMDSS-MDPnP.2007.36	25
8	Alur R, Arney D, Gunter E, Lee I, Lee J, Nam W, Pearce F, Van Albert S, Zhou J. Formal specifications and analysis of the computer-assisted resuscitation algorithm (CARA) Infusion Pump Control System. *International Journal on Software Tools for Technology Transfer* 2004; **5**(4):308–319, doi: 10.1007/s10009-003-0132-7	22
9	Jee E, Wang S, Kim J, Lee J, Sokolsky O, Lee I. A safety-assured development approach for real-time software. *Proceedings - 16th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications, RTCSA 2010*, IEEE: Macau SAR, China, 2010; 133–142, doi: 10.1109/RTCSA.2010.42	22

Table II. Publications with most citations

The most used notations are Automata, Event-B, Z, and Extended Finite State Machine (EFSM). All of them are state-based notations. One of the possible reasons why state-based notations are popular in medical software systems is because they support a model-driven development paradigm where requirements are transformed into functional code through a systematic process. In this fashion, it is easier to manage complexity, it is easier to reason about the behavior of the system, and efforts spent in earlier phases of the development ultimately result in generation of code that is correct by construction.

Notation type	Languages
Logic	Higher-Order Logic, Linear Temporal Logic (LTL), Computational Tree Logic (CTL), Temporal Ordering of Events, Timed Computational Tree Logic (TCTL)
State Based	Automata, B, Extended Finite State Machines (EFSM), Abstract State Machine (ASM), Z, Circus, State Machine, Event-B, Vienna Development Method (VDM), UML activity diagram, UML state machines, Algebraic State-Transition Diagrams (ASTD), Visual Contract Language (VCL), Mixed Signal Assertion Language (MSAL)
Event Based	Predicate/Transition Nets, Petri Nets, Activity Newtorks, Timed Transition System (TTS)

Table III. Notations used in the literature

RQ6: Which are the case studies analyzed and which activities are performed? Table IV shows which are the methods applied for each case study analyzed. Starting from the second column, we have identified a set of steps applied during the software development process. The first activity performed is modeling. Depending on the tool used by the user, a system is modeled using different notations (see RQ5). After that, a set of activities can be performed on the model. Model verification verifies whether the model satisfies given properties of interest. Model simulation is used to perform different activities, e.g., to check the behavior of the system and to demonstrate the expected results. Software validation analyses the behavior of software as compared to the model. Code generation derives software directly from the model previously defined using a tool. The certification activity aims at identifying a connection between the process applied using formal methods and the standards/guidelines that guide the medical software approval. The last column of Table IV collects papers that provide a theoretical study concerning medical devices specified in the first column.

The first column of Table IV shows the medical devices or systems which formal methods are applied to. Some case studies were proposed by research groups. Recently, at the ABZ 2016 conference [159], the hemodialysis machine case study has been provided to advocate the use of formal methods in medical applications [160]. Pacemaker requirements are provided by Software Quality Research Laboratory (SQRL) to the formal methods community to address the needs of industrial and government sectors that rely on the production of software that is critical for their missions and/or for the safety and effectiveness of their products[**]. The infusion pump case study has been provided by FDA [161][††]. It invites the formal methods community to provide techniques and tools that improve the overall reliability of medical devices. The remaining case studies proposed by researchers are derived from their own experiences.

Regarding the activity applied to the case studies, we can draw the following observations. The hemodialysis machine case study is modeled by [24–33]. Some of them perform validation [25, 30–32] and verification [24, 25, 27–29, 32, 33]. Paper [32] generates the machine code starting from formal specification, and paper [25] defines a set of characteristics to fit formal methods with the standards for medical software development process. A model of pacemaker is described in papers [16,19,22,34–61]. Other authors apply model verification [15,16,18,19,22,36,38–42,44,46, 48–52,54–57,59,61–65,65,65–67] and model validation [16,38,47,58,59,66,68,69]. The pacemaker software is validated in papers [22, 47, 56, 59, 65, 65], while papers [18, 34, 43, 44, 47, 56, 66, 70] contribute to develop a new step that consists in translating the model into machine code. The infusion pump case study is modelled [20, 21, 71–101], verified [20, 21, 71, 73–82, 84, 86, 87, 90, 91, 95–97, 101–104] and validated [71–73, 78, 83, 88, 92, 103, 105] by using different tools

[**] http://sqrl.mcmaster.ca/pacemaker.htm

[††] http://www.fda.gov/MedicalDevices/ProductsandMedicalProcedures/GeneralHospitalDevicesandSupplies/InfusionPumps/

DOI: 10.1002/smr

and languages. Papers [84, 105] present an approach for generating the machine code starting from the formal model and in [20, 86] authors use different approaches to validate software. In papers [79, 91], authors develop a method to support the approval process. Jetley et. al. [17] explain how to apply formal methods to premarket and post-market evaluations in case of medical devices (infusion pump). Formal methods are also used in medical image processing for modeling and model verification [106–109]. The paper [110] provides a survey about formal methods applied to image processing. The paper [111] shows a process that includes modeling, model simulation, model verification, model validation and software validation by comparing the code with the formal requirements model applied to a medical screening test. Electrocardiography is the process of recording the electrical activity of the heart. The system is modeled [112–121], verified [115, 119, 122], and validated [112, 114, 117] using different tools. E-health systems are a recent classification of health-care systems supported by electronic processes and communication. Papers [123–128] model the system, while it is verified in [123, 125, 127, 129, 130] and validated in [123, 131]. Furthermore, the certification criteria followed by EMR applications in category of patients' privacy protection are studied in papers [132, 133]. Suggestions on how to use formal methods are given also in the medical protocol case study [135]. Papers [14,134] propose a model of medical protocol and verify whether the overall process is correct or not. Pulse oximeter is modeled in [136] and verified in [136,137]. Models of medical device connections are verified [138–149] and validated [140–142, 145, 149]. Modelling and verification techniques are applied to imatinib dose in [150, 151]. Model validation, model verification and modeling activities are performed for left ventricular assist device [152]. A formal model is defined for clinical neutron therapy system [153] and for syringe pump [154], furthermore syringe pumps are validated in papers [68, 154]. Code for blood separator machine is automatically derived in [156], starting from the formal model that has been previously verified. A technique to model, verify and validate medical machine interface is presented in [157], while paper [158] models and verifies the endotracheal intubation system.

Overall, the most common activities performed are modeling and model verification, followed by model validation. A trend towards code generation has also been observed in recent publications. Although activities like software validation show little traction, we also consider them crucial in the medical software and systems development.

Considering all case studies, the most analyzed ones are about infusion pumps and pacemakers. Also hemodialysis machines have been studied in several articles. We can note that all these systems are proposed by research groups as benchmarks and they come with a (quite) complete description of requirements and a good documentation. Since the application of formal methods to medical systems is a relatively new phenomenon, the researchers appear to prefer to test their methods and tools in well-known case studies to compare the obtained results rather easily. Furthermore, we can add that companies may not always be inclined to provide or publish data and code to test new techniques.

RQ7: Which are the tools used for each performed activity? Table V shows which tools or tool families are used for each software development activity. The majority of tools operate on models written in their specific languages. After the modeling process, a number of activities are performed on the model. The validation step checks that relevant behaviors of the real systems are accurately captured by the model. Verification, on the other hand, analyses whether the model is consistent w.r.t. the behavior and expected properties are verifiable. There exist different verification techniques, those used in the papers found are:

- *Model Checking*: it is applied to finite-state systems. The properties are translated in formulas of a temporal logic (e.g., CTL (Computational Tree Logic) or LTL (Linear Temporal Logic) and efficient symbolic algorithms are used to verify that the model satisfies specified properties of the system. If the property is false, a counterexample is displayed. The limitation of this technique is the state space explosion: the transition graph grows exponentially which can make model checking too inefficient for the analysis of complex models.

- SMT (Satisfiability Modulo Theories) solvers: they are given an expression with boolean variables along with and/or predicates, based on which they determine the conditions that would make the expression true.
- Theorem Provers: they are based on deduction methods (rather than state space exploration as in model checking), and they may not be fully automatic, they may require user intervention to complete the proof or demonstrate the presence of a design anomaly. The lack of full automation, however, is balanced by expressive specification languages and better handling of complex models as compared to model checking.

Software validation returns differences between the behavior captured by the real system and the requirements model defined by the developer. Code generation automatically produces the code for the real system, while test case generation derives test cases for the real system.

Regarding the types of activities, Table V hints that formal approaches are primarily used for modeling purposes. Also verification and, in particular, model checking play a predominant role in this area. The main advantage of model checking is that it is fully automated, so it is the preferred mean of verification. Activities like software validation (e.g., testing) and code generation are surprisingly not common areas for the application of formal methods in medical software systems.

Regarding the tools used, we can see that most approaches use a rather small set of tools including B tools, MATLAB and Simulink, and UPPAAL. These tools have a good support and a commercial backing. Although tools like Mathworks' MATLAB and Simulink are strictly commercial, other tools like B tools and UPPAAL allow and encourage non-commercial uses as well. Other tools are used only in few case studies.

5. LIMITATIONS

Despite the efforts made to collect the papers and to perform a complete and objective analysis, the SLR process presented in this paper has some limitations as listed below.

- **Choice of the keywords**: the keywords used for this work have been chosen based on the experience of researchers involved with this SLR. During the process, some relevant keywords could be missed as some publications authors or publishers may have used different keywords. Moreover, we perform the searches in Scopus only in the "author keywords" field because Scopus automatically adds many keywords that are often irrelevant or misleading.

- **Choice of the databases**: we have chosen Scopus, Web of Science, DBLP, and Google Scholar. The use of multiple sources contributes towards the completeness of search results. However, despite our utmost efforts, it may still be possible that some relevant papers remain unnoticed, for example, a technical report that is never published in a conference.

- **Use of Scopus for citations**: we have chosen Scopus for the number of citations because of its reliability. However, this may introduce a delay in the count of the number of citations since Scopus takes a while to update citation information and this may underestimate the impact of some papers.

6. CONCLUSION AND FUTURE WORK

In this paper, we have presented a SLR about the use of formal methods in modeling, design, analysis and development of medical software and systems. In order to collect relevant papers, we have run several complex queries on a variety of databases (see Section 3). We performed a mixture of quantitative and qualitative analysis (see Section 4) to provide information that helps researchers who are either already working within this domain or planning to start. We have observed that the number of publications per year is showing an upward trend and researchers prefer to publish more in conferences than journals. Most authors have published only once in this field, and only a handful of authors have published more than two papers. It may be because researchers are interested

DOI: 10.1002/smr

in testing their methods and tools to different domains rather than just medical systems. While analyzing the most cited papers (see RQ4), we found that the authors are interested in the application of formal methods to real case studies, which work as preferred benchmarks for methodologies and techniques. This is highlighted also in Table IV. Regarding the activities performed by rigorous methodologies, we have found that most of the work revolves around modeling, validation, and verification mainly due to the availability of tools for these activities (as shown in Table V). We also found that new trends, e.g., automatic code generation and software validation, are also emerging that promise a tighter integration of formal methods with the overall development process. Principally, we have found that the use of formal and rigorous methods for medical software systems development interests very much the computer science community.

In order to minimize the bias in our systematic review, we have adopted several strategies. We have performed the queries on multiple bibliographic repositories. Although we have used a set of previously known papers, it only determines the completeness of our search and the analysis has been performed on the whole resulting set. We have identified the terms and keywords to be used in the queries starting from generic terms (like ”medical software”) refined to specific terms and case studies (like ”syringe pump”). In order to be systematic and to avoid the inclusion of too many irrelevant papers, we have included only papers that are captured by the queries and we have stated precise criteria for inclusion.

Despite our utmost care, we may still miss some relevant papers in case the authors use specific aspect of a device as keywords, e.g., user interface, control logic, or the type of considered properties, e.g., usability, performance and security. In order to avoid any such incident, the data presented in Table IV and V can be further detailed by specifying various aspects of the devices that have been modeled and analyzed and the considered properties.

During our research, we did not find any prevalent tool or formal notation that can be considered as *standard*. Although the use of formal methods is gaining attention and is being recommended by regulatory authorities, we did not find any evidence to estimate how often formal techniques are employed for modeling and analysis of medical devices by manufacturers. Ideally, we would like to have devices which are fully formally proven for their dependability and correctness, but we are far from this goal. Two reasons may be accountable for this situation. First, the current nature of medical software development*mostly builds on code inspection, which is a less rigorous approach as compared to formal methods [17]. Second, the use of formal methods for medical devices of high integrity level is only “recommended” by standards, e.g., IEC 62304, and not enforced. By showing the usefulness and effectiveness of the application of formal methods, researchers can inspire device manufacturers to eventually adopt these methods as the standard medical software development technology.

Table IV. Application of formal methods to medical devices

Medical device	Main Activity						Survey
	Modelling	Model verification	Model validation	Software validation	Code generation	Certification	
Hemodialysis Machine	[24–33]	[24, 25, 27–29, 32, 33]	[25, 30–32]		[32]	[25]	
Pacemaker	[16, 19, 22, 34–61]	[15,16,18,19,22,36, 38–42,44,46,48–52, 54–57,59,61–65,65, 65–67]	[16, 38, 47, 58,59,66,68, 69]	[22, 47, 56, 59,65,65]	[18, 34, 43, 44,47,56,66, 70]		
Infusion pump	[20, 21, 71–101]	[20, 21, 71, 73–82, 84,86,87,90,91,95–97,101–104]	[71–73, 78, 83, 88, 92, 103,105]	[20,86]	[84,105]	[79,91]	[17]
Medical image processing	[106–109]	[106–109]					[110]
Stereoacuity test	[111]	[111]	[111]	[111]			
ECG (Electrocardiography)	[112–121]	[115,119,122]	[112, 114, 117]				
e-Health system	[123–128]	[123, 125, 127, 129, 130]	[123,131]			[132,133]	
Medical protocol	[14,134]	[14,134]					[135]
Pulse oximeter	[136]	[136,137]					
Medical device connection	[138–144]	[138–149]	[140–142, 145,149]				
Imatinib dose	[150,151]	[150,151]					
Left Ventricular Assist Device	[152]	[152]	[152]				
Clinical Neutron Therapy System	[153]						
Syringe Pump	[154,155]	[155]	[68,154]				
Blood separator machine	[156]	[156]			[156]		
Medical interface	[157]	[157]	[157]				
Endotracheal intubation	[158]	[158]					

Table V. Tools used for each methodology

	Modelling	Model validation	Verification				Software validation	Code generation
			Model checking	SMT	Theorem prover	Other Not specified		
MATLAB and Simulink	[19,41,71,74,78,79, 83,86,91,112–114]	[18, 71, 78, 83,112,114]	[74,78,86]	[74]		[18, 41, 65, 65, 71, 74, 79,91,158]	[65, 65, 86]	[19]
SPIN	[53,78,106]		[52,78,106,157]					
UPPAAL	[20,40,42,44,47, 56,63,78,86,107–109,125,150]	[47,78]	[15, 19, 20, 40, 42, 44, 47, 56, 78, 86, 107–109, 125,150]	[40]			[20,86]	[44]
SCR	[21,52,71]	[71]	[21]					
ASMETA	[25,111]	[25,111]	[25,111]				[111]	
Z tools	[37,77,120,124, 143,153]		[77,143]					[43]
SAL	[76,81,119,152]	[152]	[76,119]	[81, 152]				
B tools	[26–28, 30–32, 34, 38, 39, 46, 73, 85, 93, 95, 115, 118,154]	[30–32, 38, 73,103,154]	[27,32,38]		[32]	[28, 39, 46,73,95, 103, 115, 117,122]		[32, 34, 70]
PVS	[72, 75, 79, 80, 84,155]	[72,105]			[75, 84]	[64,79,80, 155]		[84, 105]
VDM	[123]	[123]				[123]		
ProVerif	[138]					[138]		
Perfect Developer	[43]							
Real-Time Maude	[60,68,139]	[68]	[139]					
NuSMV			[137]					
CBMC			[137]			[62]		
SATABS			[137]					
CEGAR	[42,55]		[42,55]					
VCB	[35]							
Asbru	[134]							
KIV					[14]	[134]		
ABV	[24]					[24]		
Yices	[152]		[152]	[152]				
Z3	[40]		[40]	[40]				
BLESS	[36,54]					[36,54]		
OSCP	[144]					[144]		
IVY workbench	[82,87]		[82]			[87]		
Circus	[29]		[29]					
CellExcite	[116]							
LOTOS tools	[140]	[140]				[140]		
V2T	[141,142]	[141,142]				[141,142]		
MathSAT 5				[146]				
PRISM			[129,147,148]					
MetaEdit+	[156]					[156]		[156]
SystemJ	[49]					[49]		
CPS-MAS	[101]					[101]		
StateRover	[96]	[96]				[96]		
TextBT	[92]							
SimTree		[92]						
TIMES	[151]		[151]					

REFERENCES

1. Jones P, Jetley R, Abraham J. Formal methods-based verification of medical device software analysis. *Electronic Engineering Times* 2010; :31–32.
2. Tirat-Gefen Y. Formal methods in verification of medical devices towards hybrid nano- And microsystems. *Proceedings of the IEEE Annual Northeast Bioengineering Conference*, vol. 2006, Easton, PA, 2006; 139–140.
3. Böckenholt U, Weber EU. Use of formal Methods in Medical Decision Making: A Survey and Analysis. *Medical Decision Making* dec 1992; **12**(9):298–306, doi: 10.1177/0272989X9201200409.
4. Sivakumar M, Casey V, McCaffery F, Coleman G. Improving verification & validation in the medical device domain. *Communications in Computer and Information Science* 2011; **172**:61–71, doi: 10.1007/978-3-642-22206-1.
5. Curls C. FDA-regulated validation in clinical and nonclinical environments (Regulatory Affairs). *IEEE Engineering in Medicine and Biology Magazine* Jan 2007; **26**(1):91–97, doi: 10.1109/MEMB.2007.289127.
6. Lin W, Fan X. Software Development Practice for FDA-Compliant Medical Devices. *Computational Sciences and Optimization, 2009. CSO 2009. International Joint Conference on*, vol. 2, IEEE: Sanya, Hainan, China, 2009; 388–390, doi: 10.1109/CSO.2009.191.
7. Doğan S, Betin-Can A, Garousi V. Web Application Testing: A Systematic Literature Review. *J. Syst. Softw.* May 2014; **91**:174–201, doi: 10.1016/j.jss.2014.01.010.
8. Majikes JJ, Pandita R, Xie T. Literature Review of Testing Techniques for Medical Device Software. *Proceedings of the Medical Cyber Physical Systems Workshop*, Philadelphia, USA, 2013.
9. Kitchenham B, Brereton OP, Budgen D, Turner M, Bailey J, Linkman S. Systematic literature reviews in software engineering? A systematic literature review. *Information and Software Technology* 2009; **51**(1):7 – 15, doi: http://dx.doi.org/10.1016/j.infsof.2008.09.009.
10. Bonfanti S, Gargantini A, Mashkoor A. A Preliminary Systematic Literature Review of the use of Formal Methods in Medical Software Systems. *Industrial Proceedings of the 23rd EuroAsiaSPI Conference, Graz University of Technology, Graz, Austria*, 2016; 15–23.
11. Higgins JP, Green S. *Cochrane handbook for systematic reviews of interventions*, vol. 4. John Wiley & Sons: London, UK, 2011.
12. Delgado Lopez-Cozar E, Robinson-Garcia N, Torres-Salinas D. Manipulating Google Scholar Citations and Google Scholar Metrics: simple, easy and tempting. *Technical Report*, Computing Research Repository Dec 2012.
13. Yang K, Meho LI. Citation Analysis: A Comparison of Google Scholar, Scopus, and Web of Science. *Proceedings of the American Society for Information Science and Technology* 2006; **43**(1):1–15, doi: 10.1002/meet.14504301185.
14. Ten Teije A, Marcos M, Balser M, Van Croonenborg J, Duelli C, Van Harmelen F, Lucas P, Miksch S, Reif W, Rosenbrand K, *et al.*. Improving medical protocols by formal methods. *Artificial Intelligence in Medicine* 2006; **36**(3):193–209, doi: 10.1016/j.artmed.2005.10.006.
15. David A, Oliver Möller M, Yi W. Formal verification of UML statecharts with real-time extensions. *Fundamental Approaches to Software Engineering*, *Lecture Notes in Computer Science*, vol. 2306, Springer Berlin Heidelberg, 2002; 218–232, doi: 10.1007/3-540-45923-5_15.
16. Jiang Z, Pajic M, Mangharam R. Cyber-physical modeling of implantable cardiac medical devices. *Proceedings of the IEEE* jan 2012; **100**(1):122–137, doi: 10.1109/JPROC.2011.2161241.
17. Jetley R, Iyer S, Jones P. A formal methods approach to medical device review. *Computer* 2006; **39**(4):61–67, doi: 10.1109/MC.2006.113.
18. Jiang Z, Pajic M, Connolly A, Dixit S, Mangharam R. Real-time heart model for implantable cardiac device validation and verification. *Proceedings - Euromicro Conference on Real-Time Systems*, Brussels, Belgium, 2010; 239–248, doi: 10.1109/ECRTS.2010.36.
19. Pajic M, Jiang Z, Lee I, Sokolsky O, Mangharam R. From verification to implementation: A model translation tool and a pacemaker case study. *Real-Time Technology and Applications - Proceedings*, IEEE: Beijing, China, 2012; 173–184, doi: 10.1109/RTAS.2012.25.
20. Arney D, Jetley R, Jones P, Lee I, Sokolsky O. Formal methods based development of a PCA infusion pump reference model: Generic infusion pump (GIP) project. *High Confidence Medical Devices, Software, and Systems and Medical Device Plug-and-Play Interoperability*, IEEE: Boston, MA, USA, 2007; 23–33, doi: 10.1109/HCMDSS-MDPnP.2007.36.
21. Alur R, Arney D, Gunter E, Lee I, Lee J, Nam W, Pearce F, Van Albert S, Zhou J. Formal specifications and analysis of the computer-assisted resuscitation algorithm (CARA) Infusion Pump Control System. *International Journal on Software Tools for Technology Transfer* 2004; **5**(4):308–319, doi: 10.1007/s10009-003-0132-7.
22. Jee E, Wang S, Kim J, Lee J, Sokolsky O, Lee I. A safety-assured development approach for real-time software. *Proceedings - 16th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications, RTCSA 2010*, IEEE: Macau SAR, China, 2010; 133–142, doi: 10.1109/RTCSA.2010.42.
23. Abadi M, Lamport L. Composing specifications. *ACM Transactions on Programming Languages and Systems* 1993; **15**(1):73–132.
24. Lämmermann S, Ruf J, Pielawa ABL, Kropf JT, Schlemminger WR, Hein A. Heterogeneous Assertion-Based Verification for Medical Devices Development. *Proceedings SASIMI*, 2012; 211 – 216.
25. Arcaini P, Bonfanti S, Gargantini A, Riccobene E. How to assure correctness and safety of medical software: The hemodialysis machine case study. *International Conference on Abstract State Machines, Alloy, B, TLA, VDM, and Z - ABZ 2016*, *Lecture Notes in Computer Science*, vol. 9675, Butler M, Schewe KD, Mashkoor A, Biro M (eds.), Springer Verlag: Linz, Austria, 2016; 344–359, doi: 10.1007/978-3-319-33600-8_30.
26. Fayolle T, Frappier M, Gervais F, Laleau R. Modelling a hemodialysis machine using Algebraic state-Transition Diagrams and B-like methods. *International Conference on Abstract State Machines, Alloy, B, TLA, VDM, and Z - ABZ 2016*, *Lecture Notes in Computer Science*, vol. 9675, Butler M, Schewe KD, Mashkoor A, Biro M (eds.), Springer Verlag: Linz, Austria, 2016; 394–408, doi: 10.1007/978-3-319-33600-8_33.

27. Mashkoor A. Model-driven development of high-assurance active medical devices. *Software Quality Journal* 2016; **24**(3):571–596, doi: 10.1007/s11219-015-9288-0.
28. Banach R. Hemodialysis machine in Hybrid Event-B. *International Conference on Abstract State Machines, Alloy, B, TLA, VDM, and Z - ABZ 2016*, *Lecture Notes in Computer Science*, vol. 9675, Butler M, Schewe KD, Mashkoor A, Biro M (eds.), Springer Verlag: Linz, Austria, 2016; 376–393, doi: 10.1007/978-3-319-33600-8_32.
29. Gomes A, Butterfield A. Modelling the haemodialysis machine with Circus. *International Conference on Abstract State Machines, Alloy, B, TLA, VDM, and Z - ABZ 2016*, *Lecture Notes in Computer Science*, vol. 9675, Butler M, Schewe KD, Mashkoor A, Biro M (eds.), Springer Verlag: Linz, Austria, 2016; 409–424, doi: 10.1007/978-3-319-33600-8_34.
30. Mashkoor A, Biro M, Dolgos M, Timar P. Refinement-based development of software-controlled safety-critical active medical devices. *Software Quality. Software and Systems Quality in Distributed and Mobile Environments - 7th International Conference, SWQD 2015, Vienna, Austria, January 20-23, 2015, Proceedings*, *Lecture Notes in Business Information Processing*, vol. 200, 2015; 120–132, doi: 10.1007/978-3-319-13251-8_8.
31. Hoang T, Snook C, Ladenberger L, Butler M. Validating the requirements and design of a hemodialysis machine using iUML-B, BMotion studio, and co-simulation. *International Conference on Abstract State Machines, Alloy, B, TLA, VDM, and Z - ABZ 2016*, *Lecture Notes in Computer Science*, vol. 9675, Butler M, Schewe KD, Mashkoor A, Biro M (eds.), Springer Verlag: Linz, Austria, 2016; 360–375, doi: 10.1007/978-3-319-33600-8_31.
32. Mashkoor A, Biro M. Towards the Trustworthy Development of Active Medical Devices: A Hemodialysis Case Study. *IEEE Embedded Systems Letters* March 2016; **8**(1):14–17, doi: 10.1109/LES.2015.2494459.
33. Pielawa L, Frenken M, Hein A. A workflow for design and evaluation of embedded control systems in medical devices. *International Journal of Biomedical Engineering and Technology* 2013; **13**(3):257–269, doi: 10.1504/IJBET.2013.058446.
34. Méry D, Singh N. Formal Development and Automatic Code Generation Cardiac Pacemaker. *International Conference on Computers and Advanced Technology in Education*, Beijing, China, 2011; 210–225.
35. Leemans J, Amálio N. Modelling a cardiac pacemaker visually and formally. *VLHCC*, IEEE: Innsbruck, Austria, 2012; 257–258, doi: 10.1109/VLHCC.2012.6344542.
36. Larson BR. Formal semantics for the PACEMAKER System Specification. *ACM SIGAda Ada Letters*, Association for Computing Machinery, Inc: Portland, Oregon, 2014; 47–59, doi: 10.1145/2663171.2663182.
37. Gomes A, Oliveira M. Formal specification of a cardiac pacing system. *International Symposium on Formal Methods - FM 2009*, vol. 5850 LNCS, Eindhoven, The Netherlands,, 2009; 692–707, doi: 10.1007/978-3-642-05089-3_44.
38. Méry D, Singh N. Formal specification of medical systems by proof-based refinement. *Transactions on Embedded Computing Systems* 2013; **12**(1):15:1–15:25, doi: 10.1145/2406336.2406351.
39. Singh N, Lawford M, Maibaum T, Wassyng A. Formalizing the Cardiac Pacemaker Resynchronization Therapy. *Digital Human Modeling. Applications in Health, Safety, Ergonomics and Risk Management: Ergonomics and Health: 6th International Conference, DHM 2015, Held as Part of HCI International 2015, Los Angeles, CA, USA, August 2-7, 2015, Proceedings, Part II*, *Lecture Notes in Computer Science*, vol. 9185, Springer, 2015; 374–386, doi: 10.1007/978-3-319-21070-4_38.
40. Shuja S, Srinivasan S, Jabeen S, Nawarathna D. A formal verification methodology for DDD mode pacemaker control programs. *Journal of Electrical and Computer Engineering* 2015; **2015**:57–67, doi: 10.1155/2015/939028.
41. Kwiatkowska M, Lea-Banks H, Mereacre A, Paoletti N. Formal modelling and validation of rate-adaptive pacemakers. *Healthcare Informatics (ICHI) IEEE International Conference on*, Institute of Electrical and Electronics Engineers Inc.: Verona, Italy, 2014; 23–32, doi: 10.1109/ICHI.2014.11.
42. Jiang Z, Pajic M, Alur R, Mangharam R. Closed-loop verification of medical devices with model abstraction and refinement. *International Journal on Software Tools for Technology Transfer* 2014; **16**(2):191–213, doi: 10.1007/s10009-013-0289-7.
43. Gomes A, Oliveira M. Formal development of a cardiac pacemaker: From specification to code. *Brazilian Symposium on Formal Methods - SBMF 2010: Formal Methods: Foundations and Applications*, vol. 6527 LNCS, Natal, Brazil, 2011; 210–225, doi: 10.1007/978-3-642-19829-8_14.
44. Jiang Z, Abbas H, Jang KJ, Mangharam R. The Challenges of High-Confidence Medical Device Software. *Computer* Jan 2016; **49**(1):34–42, doi: 10.1109/MC.2016.20.
45. Xu J, Venkatasubramanian K, Sfyrla V. A methodology for systematic attack trees generation for interoperable medical devices. *10th Annual International Systems Conference, SysCon 2016 - Proceedings*, 2016, doi: 10.1109/SYSCON.2016.7490632.
46. Sulskus G, Poppleton M, Rezazadeh A. Modelling complex timing requirements with refinement. *Proceedings - 2016 IEEE 17th International Conference on Information Reuse and Integration, IRI 2016*, Pittsburgh, Pennsylvania, 2016; 118–125, doi: 10.1109/IRI.2016.23.
47. Jiang Z, Mangharam R. High-confidence medical device software development. *Foundations and Trends in Electronic Design Automation* 2015; **9**(4):309–391, doi: 10.1561/1000000040.
48. Grosu R, Cherry E, Clarke E, Cleaveland R, Dixit S, Fenton F, Gao S, Glimm J, Gray R, Mangharam R, *et al.*. Compositional, approximate, and quantitative reasoning for medical cyber-physical systems with application to patient-specific cardiac dynamics and devices. *Leveraging Applications of Formal Methods, Verification and Validation. Specialized Techniques and Applications*, *Lecture Notes in Computer Science*, vol. 8803, Springer Berlin Heidelberg, 2014; 356–364, doi: 10.1007/978-3-662-45231-8_26.
49. Park H, Malik A, Nadeem M, Salcic Z. The cardiac pacemaker-System J versus safety critical java. *ACM International Conference Proceeding Series*, vol. 2014-October, Niagara Falls, NY, 2014; 37–46, doi: 10.1145/2661020.2661030.
50. Bessling S, Huhn M. Towards formal safety analysis in feature-oriented product line development. *Foundations of Health Information Engineering and Systems*, *Lecture Notes in Computer Science*, vol. 8315, Springer Berlin Heidelberg, 2014; 217–235, doi: 10.1007/978-3-642-53956-5_15.

51. Banach R, Zhu H, Su W, Wu X. A continuous ASM modelling approach to pacemaker sensing. *ACM Transactions on Software Engineering and Methodology* 2014; **24**(1):2:1–2:40, doi: 10.1145/2610375.
52. Scilingo G, Novaira M, Degiovanni R, Aguirre N. Analyzing formal requirements specifications using an off-the-shelf model checker. *Proceedings of the 2013 39th Latin American Computing Conference, CLEI 2013*, 1-9, Naiguata, Venezuela, 2013, doi: 10.1109/CLEI.2013.6670611.
53. Sharma A. A refinement calculus for promela. *Proceedings of the IEEE International Conference on Engineering of Complex Computer Systems, ICECCS*, Singapore, 2013; 75–84, doi: 10.1109/ICECCS.2013.20.
54. Larson B, Chalin P, Hatcliff J. BLESS: Formal specification and verification of behaviors for embedded systems with software. *NASA Formal Methods: 5th International Symposium, NFM 2013, Moffett Field, CA, USA, May 14-16, 2013. Proceedings*, vol. 7871, Springer Berlin Heidelberg: Moffett Field, CA, USA, 2013; 276–290, doi: 10.1007/978-3-642-38088-4_19.
55. Jiang Z, Mangharam R. Multi-scale modeling of the heart for closed-loop evaluation of pacemaker software. *ASME 2013 Conference on Frontiers in Medical Devices: Applications of Computer Modeling and Simulation, FMD 2013*, Washington, DC, USA, 2013; V001T10A051, doi: 10.1115/FMD2013-16192.
56. Jiang Z, Pajic M, Moarref S, Alur R, Mangharam R. Modeling and verification of a dual chamber implantable pacemaker. *Tools and Algorithms for the Construction and Analysis of Systems*, *Lecture Notes in Computer Science*, vol. 7214 LNCS, Springer Berlin Heidelberg, 2012; 188–203, doi: 10.1007/978-3-642-28756-5_14.
57. De Oliveira R, Santos G, Farines JM, Becker L. Contributions to improvement of the formal properties verification process in AADL programs. *Proceedings - 2011 Brazilian Symposium on Computing System Engineering, SBESC 2011*, Florianópolis, Brazil, 2011; 27–32, doi: 10.1109/SBESC.2011.28.
58. Jee E, Lee I, Sokolsky O. Assurance cases in model-driven development of the pacemaker software. *Leveraging Applications of Formal Methods, Verification, and Validation: 4th International Symposium on Leveraging Applications, ISoLA 2010, Heraklion, Crete, Greece, October 18-21, 2010, Proceedings, Part II*, *LNCS*, vol. 6416, Springer Berlin Heidelberg, 2010; 343–356, doi: 10.1007/978-3-642-16561-0_33.
59. Jiang Z, Pajic M, Connolly A, Dixit S, Mangharam R. Demo abstract: A platform for implantable medical device validation. *Proceedings - Wireless Health 2010, WH'10*, San Diego, CA, USA, 2010; 208–209, doi: 10.1145/1921081.1921115.
60. Sun M, Meseguer J, Sha L. A formal pattern architecture for safe medical systems. *Rewriting Logic and Its Applications: 8th International Workshop, WRLA 2010, Held as a Satellite Event of ETAPS 2010, Paphos, Cyprus, March 20-21, 2010, Revised Selected Papers*, vol. 6381 LNCS, 2010; 157–173, doi: 10.1007/978-3-642-16310-4_11.
61. Tuan L, Zheng M, Tho Q. Modeling and verification of safety critical systems: A case study on pacemaker. *SSIRI 2010 - 4th IEEE International Conference on Secure Software Integration and Reliability Improvement*, Singapore, 2010; 23–32, doi: 10.1109/SSIRI.2010.28.
62. Li C, Raghunathan A, Jha N. Improving the trustworthiness of medical device software with formal verification methods. *IEEE Embedded Systems Letters* Sept 2013; **5**(3):50–53, doi: 10.1109/LES.2013.2276434.
63. Guo Y, Yin L, Li C. Automatically verifying STRAC policy. *Proceedings - IEEE INFOCOM*, Toronto, ON, Canada, 2014; 141–142, doi: 10.1109/INFCOMW.2014.6849195.
64. Bernardeschi C, Domenici A, Masci P. Integrated simulation of implantable cardiac pacemaker software and heart models. *CARDIOTECHNIX 2014 - Proceedings of the 2nd International Congress on Cardiovascular Technologies*, Lisbon, Portugal, 2014; 55–59.
65. Jiang Z, Connolly A, Mangharam R. Using the Virtual Heart Model to validate the mode-switch pacemaker operation. *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*, IEEE: Buenos Aires, Argentina, 2010; 6690–6693, doi: 10.1109/IEMBS.2010.5626262.
66. Pajic M, Jiang Z, Connolly A, Dixit S, Mangharam R. A platform for implantable medical device validation. *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks, IPSN '10*, Stockholm, Sweden, 2010; 418–419, doi: 10.1145/1791212.1791284.
67. Cheng A. Cyber-physical medical and medication systems. *Proceedings - International Conference on Distributed Computing Systems*, Beijing, China, 2008; 529–532, doi: 10.1109/ICDCS.Workshops.2008.67.
68. Sun M, Meseguer J. Distributed Real-Time Emulation of Formally-Defined Patterns for Safe Medical Device Control. *RTRTS*, *EPTCS*, vol. 36, Longyearbyen, Norway, 2010; 158–177.
69. Ai W, Patel N, Roop P. Requirements-centric closed-loop validation of implantable cardiac devices. *Proceedings of the 2016 Design, Automation and Test in Europe Conference and Exhibition, DATE 2016*, Dresden, Germany, 2016; 846–849. URL `https://dl.acm.org/citation.cfm?id=2971808.2972003`.
70. Singh N. *Using event-B for critical device software systems*. Springer London, 2013, doi: 10.1007/978-1-4471-5260-6.
71. Silva L, Almeida H, Perkusich A, Perkusich M. A model-based approach to support validation of medical cyber-physical systems. *Sensors (Switzerland)* 2015; **15**(11):27 625–27 670, doi: 10.3390/s151127625.
72. Blandford A, Cauchi A, Curzon P, Eslambolchilar P, Furniss D, Gimblett A, Huang H, Lee P, Li Y, Masci P, *et al.*. Comparing actual practice and user manuals:A case study based on programmable infusion pumps. *Intl. Workshop on Engineering Interactive Computing Systems for Medicine and Health Care.*, vol. 727, 2011; 59–64.
73. Singh N, Wang H, Lawford M, Maibaum T, Wassyng A. Stepwise formal modelling and reasoning of insulin infusion pump requirements. *Conference of 6th International Conference on Digital Human Modeling, DHM 2015 Held as Part of 17th International Conference on Human-Computer Interaction, HCI International 2015*, vol. 9185, VG D (ed.), Springer Verlag, 2015; 387–398, doi: 10.1007/978-3-319-21070-4_39.
74. Murugesan A, Whalen M, Rayadurgam S, Heimdahl M. Compositional verification of a medical device system. *ACM SIGAda Ada Letters*, Pittsburgh, Pennsylvania, USA, 2013; 51–64, doi: 10.1145/2527269.2527272.
75. Masci P, Zhang Y, Jones P, Curzon P, Thimbleby H. Formal verification of medical device user interfaces using PVS. *Conference of 17th International Conference on Fundamental Approaches to Software Engineering, FASE 2014 - Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2014*, vol. 8411 LNCS, Springer Verlag: Grenoble, 2014; 200–214, doi: 10.1007/978-3-642-54804-8_14.

76. Masci P, Rukšenas R, Oladimeji P, Cauchi A, Gimblett A, Li Y, Curzon P, Thimbleby H. The benefits of formalising design guidelines: a case study on the predictability of drug infusion pumps. *Innovations in Systems and Software Engineering* 2015; **11**(2):73–93, doi: 10.1007/s11334-013-0200-4.
77. Babamir S, Borhani M. Formal verification of medical monitoring software using Z language: A representative sample. *Journal of Medical Systems* 2012; **36**(4):2633–2648, doi: 10.1007/s10916-011-9739-5.
78. Jetley R, Carlos C, Iyer S. A case study on applying formal methods to medical devices: Computer-aided resuscitation algorithm. *International Journal on Software Tools for Technology Transfer* 2004; **5**(4):320–330, doi: 10.1007/s10009-003-0137-2.
79. Curzon P, Masci P, Oladimeji P, Rukšenas R, Thimbleby H, D'Urso E. Human-Computer Interaction and the Formal Certification and Assurance of Medical Devices: The CHI+ MED Project. *2nd Workshop on Verification and Assurance (Verisure2014), in association with Computer-Aided Verification (CAV)*, 2014.
80. Masci P, Curzon P, Harrison M, Ayoub A, Lee I, Thimbleby H. Verification of interactive software for medical devices: PCA infusion pumps and FDA regulation as an example. *EICS 2013 - Proceedings of the ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, ACM: London, United Kingdom, 2013; 81–90, doi: 10.1145/2494603.2480302.
81. Rukšenas R, Curzon P, Blandford A, Back J. Combining human error verification and timing analysis: A case study on an infusion pump. *Formal Aspects of Computing* 2014; **26**(5):1033–1076, doi: 10.1007/s00165-013-0288-1.
82. Harrison M, Campos J, Rukšenas R, Curzon P. Modelling information resources and their salience in medical device design. *Conference of 8th ACM SIGCHI Symposium on Engineering Interactive Computing Systems, EICS 2016*, Association for Computing Machinery, Inc: Brussels, Belgium, 2016; 194–203, doi: 10.1145/2933242.2933250.
83. Banerjee A, Zhang Y, Jones P, Gupta S. Using formal methods to improve home-use medical device safety. *Biomedical Instrumentation and Technology* 2013; **47**(SPRING):43–48, doi: 10.2345/0899-8205-47.s1.43.
84. Masci P, Ayoub A, Curzon P, Lee I, Sokolsky O, Thimbleby H. Model-based development of the generic PCA infusion pump user interface prototype in PVS. *International Conference on Computer Safety, Reliability, and Security - SAFECOMP 2013: Computer Safety, Reliability, and Security*, vol. 8153 LNCS, 2013; 228–240, doi: 10.1007/978-3-642-40793-2_21.
85. Poerschke C, Lightfoot D, Nealon J. A formal specification in B of a medical decision support system. *ZB 2003: Formal Specification and Development in Z and B*, vol. 2651, 2003; 497–512, doi: 10.1007/3-540-44880-2_29.
86. Jetley R, Purushothaman Iyer S, Jones P, Spees W. A formal approach to pre-market review for medical device software. *Annual International Computer Software and Applications Conference*, vol. 1, Chicago, IL, 2006; 169–177, doi: 10.1109/COMPSAC.2006.9.
87. Campos J, Doherty G, Harrison M. Analysing interactive devices based on information resource constraints. *International Journal of Human Computer Studies* 2014; **72**(3):284–297, doi: 10.1016/j.ijhcs.2013.10.005.
88. Ling CL, Shen W, Kountanis D. Applying SOFL to a generic insulin pump software design. *Structured Object-Oriented Formal Language and Method*, vol. 7787 LNCS, Springer Berlin Heidelberg, 2013; 116–132, doi: 10.1007/978-3-642-39277-1_9.
89. Martin J. Formal methods software engineering for the CARA system. *International Journal on Software Tools for Technology Transfer* 2004; **5**(4):301–307, doi: 10.1007/s10009-003-0113-x.
90. Stark E. Formally specifying CARA in Java. *International Journal on Software Tools for Technology Transfer* 2004; **5**(4):331–350, doi: 10.1007/s10009-003-0124-7.
91. Liu J, Backes J, Cofer D, Gacek A. From design contracts to component requirements verification. *NASA Formal Methods: 8th International Symposium, NFM 2016, Minneapolis, MN, USA, June 7-9, 2016, Proceedings*, vol. 9690, Springer International Publishing, 2016; 373–387, doi: 10.1007/978-3-319-40648-0_28.
92. Zafar S, Farooq-Khan N, Ahmed M. Requirements simulation for early validation using Behavior Trees and Datalog. *Information and Software Technology* 2015; **61**:52–70, doi: 10.1016/j.infsof.2015.01.005.
93. Xu H, Maibaum T. An event-B approach to timing issues applied to the generic insulin infusion pump. *Foundations of Health Informatics Engineering and Systems*, vol. 7151 LNCS, Springer Berlin Heidelberg, 2012; 160–176, doi: 10.1007/978-3-642-32355-3_10.
94. Ayoub A, Kim B, Lee I, Sokolsky O. A safety case pattern for model-based development approach. *NASA Formal Methods: 4th International Symposium, NFM 2012, Norfolk, VA, USA, April 3-5, 2012. Proceedings*, vol. 7226 LNCS, Springer Berlin Heidelberg, 2012; 141–146, doi: 10.1007/978-3-642-28891-3_14.
95. Babamir S, Jalili S. Synthesizing a specification-based monitor for safety requirements. *Iranian Journal of Science and Technology, Transaction B: Engineering* 2010; **34**(3):235–256.
96. Drusinsky D, Shing MT, Demir K. Creation and validation of embedded assertion statecharts. *Proceedings of the International Workshop on Rapid System Prototyping*, vol. 2006, 2006; 17–23, doi: 10.1109/RSP.2006.12.
97. Zafar S, Dromey R. Integrating safety and security requirements into design of an embedded system. *Proceedings - Asia-Pacific Software Engineering Conference, APSEC*, vol. 2005, Taipei, Taiwan, 2005; 629–636, doi: 10.1109/APSEC.2005.75.
98. Drusinsky D. Visual formal specification using (N)TLCharts: Statechart automata with temporal logic and natural language conditioned transitions. *Proceedings - International Parallel and Distributed Processing Symposium, IPDPS 2004 (Abstracts and CD-ROM)*, vol. 18, Santa Fe, NM, USA, 2004; 3673–3680.
99. Drusinsky D, Shing MT. TL charts: Armor-plating Harel statecharts with temporal logic conditions. *Proceedings of the International Workshop on Rapid System Prototyping*, Geneva, Switzerland, 2004; 29–36, doi: 10.1109/IWRSP.2004.1311092.
100. Drusinsky D, Shing MT, Demir K. Creating and validating embedded assertion statecharts. *IEEE Distributed Systems Online* 2007; **8**(5):3–3, doi: 10.1109/MDSO.2007.25.
101. Banerjee A, Gupta S, Fainekos G, Varsamopoulos G. Towards modeling and analysis of cyber-physical medical systems. *Proceedings of the 4th International Symposium on Applied Sciences in Biomedical and Communication Technologies - ISABEL 11*, ACM Press, 2011, doi: 10.1145/2093698.2093852.

102. Ray A, Jetley R, Jones P, Zhang Y. Model-based engineering for medical-device software. *Biomedical Instrumentation and Technology* 2010; **44**(6):507–518, doi: 10.2345/0899-8205-44.6.507.
103. Singh N, Wang H, Lawford M, Maibaum T, Wassyng A. Formalizing the glucose homeostasis mechanism. *Digital Human Modeling. Applications in Health, Safety, Ergonomics and Risk Management: 5th International Conference, DHM 2014, Held as Part of HCI International 2014, Heraklion, Crete, Greece, June 22-27, 2014. Proceedings*, vol. 8529 LNCS, Springer International Publishing, 2014; 460–471, doi: 10.1007/978-3-319-07725-3_46.
104. Ray A, Cleaveland R. Unit verification: The CARA experience. *International Journal on Software Tools for Technology Transfer* 2004; **5**(4):351–369, doi: 10.1007/s10009-003-0134-5.
105. Mauro G, Thimbleby H, Domenici A, Bernardeschi C. Extending a user interface prototyping tool with automatic MISRA C code generation. *Electronic Proceedings in Theoretical Computer Science, EPTCS*, vol. 240, 2017; 53–66, doi: 10.4204/EPTCS.240.4.
106. Ding J, He X. Formal specification and analysis of an agent-based medical image processing system. *International Journal of Software Engineering and Knowledge Engineering* 2010; **20**(3):311–345, doi: 10.1142/S021819401000475X.
107. Arney D, Lee I, Goldman J, Whitehead S. Synchronizing an x-ray and anesthesia machine ventilator: A medical device interoperability case study. *BIODEVICES 2009 - Proceedings of the 2nd International Conference on Biomedical Electronics and Devices*, Porto, Portugal, 2009; 52–60.
108. Arney D, Goldman J, Whitehead S, Lee I. Improving patient safety with X-Ray and anesthesia machine ventilator synchronization: A medical device interoperability case study. *Communications in Computer and Information Science* 2010; **52**:96–109, doi: 10.1007/978-3-642-11721-3_7.
109. Daw Z, Cleaveland R, Vetter M. Formal verification of software-based medical devices considering medical guidelines. *International Journal of Computer Assisted Radiology and Surgery* 2014; **9**(1):145–153, doi: 10.1007/s11548-013-0919-2.
110. Neufeld E, Kuster N. Verification & Validation Benchmarks for Assessing and Demonstrating the Credibility of Computational Medical Device Evaluation. *2015 9th European Conference on Antennas and Propagation, EuCAP 2015*, Institute of Electrical and Electronics Engineers Inc.: Lisbon, Portugal, 2015; 1–2. URL `http://ieeexplore.ieee.org/abstract/document/7228601/`.
111. Arcaini P, Bonfanti S, Gargantini A, Mashkoor A, Riccobene E. Formal validation and verification of a medical software critical component. *Formal Methods and Models for Codesign (MEMOCODE)*, Institute of Electrical and Electronics Engineers Inc.: Austin, TX, USA, 2015; 80–89, doi: 10.1109/MEMCOD.2015.7340473.
112. Sobrinho, Cunha P, Da Silva L, Perkusich A, Cordeiro T, Rêgo J. A simulation approach to certify electrocardiography devices. *International Conference on E-health Networking, Application Services (HealthCom)*, Boston, MA, USA, 2015; 86–90, doi: 10.1109/HealthCom.2015.7454478.
113. Barbot B, Kwiatkowska M, Mereacre A, Paoletti N. Estimation and verification of hybrid heart models for personalised medical and wearable devices. *International Conference on Computational Methods in Systems Biology - CMSB 2015*, vol. 9308, Roux O BJ (ed.), Springer Verlag: Nantes, France, 2015; 3–7, doi: 10.1007/978-3-319-23401-4_1.
114. Sobrinho, Cunha P, Da Silva L, Perkusich A, Cordeiro T, Rego J. A methodology for modeling and simulation of biomedical signal acquisition devices. *International Conference on E-health Networking, Application & Services*, Boston, MA, USA, 2015; 227–231, doi: 10.1109/HealthCom.2015.7454503.
115. Méry D, Singh N. Medical protocol diagnosis using formal methods. *International Symposium on Foundations of Health Informatics Engineering and Systems - FHIES 2011*, *Lecture Notes in Computer Science*, vol. 7151, Johannesburg, South Africa, 2012; 1–20, doi: 10.1007/978-3-642-32355-3_1.
116. Bartocci E, Corradini F, Grosu R, Merelli E, Riganelli O, Smolka S. StonyCam: A formal framework for modeling, analyzing and regulating cardiac myocytes. *Concurrency, Graphs and Models*, vol. 5065 LNCS, Springer Berlin Heidelberg, 2008; 493–502, doi: 10.1007/978-3-540-68679-8_30.
117. Al-Hamadi H, Gawanmeh A, Al-Qutayri M. Formal validation of QRS wave within ECG. *2015 International Conference on Information and Communication Technology Research, ICTRC 2015*, Abu Dhabi, United Arab Emirates, 2015; 190–193, doi: 10.1109/ICTRC.2015.7156454.
118. Al-Hamadi H, Gawanmeh A, Al-Qutayri M. Formalizing electrocardiogram (ECG) signal behavior in Event-B. *2014 IEEE 16th International Conference on e-Health Networking, Applications and Services, Healthcom 2014*, Natal, Brazil, 2015; 55–60, doi: 10.1109/HealthCom.2014.7001813.
119. Poroor J, Jayaraman B. Formal analysis of event-driven cyber physical systems. *ACM International Conference Proceeding Series*, Kollam, India, 2012; 1–8, doi: 10.1145/2490428.2490429.
120. Carvalho L, Motta G, Meira S. Object oriented formal specifications: application in the development of an automatic exercise ECG processing system. *Computers in Cardiology, Proceedings*, IEEE: London, UK, 1993; 903–906.
121. Pantelopoulos A, Bourbakis N. A formal language approach for multi-sensor wearable health-monitoring systems. *8th IEEE International Conference on BioInformatics and BioEngineering, BIBE 2008*, Athens, Greece, 2008; 1–7, doi: 10.1109/BIBE.2008.4696772.
122. Al-Hamadi H, Gawanmeh A, Al-Qutayri M, Ismail M. A framework for the verification of an ECG biosensor algorithm. *Analog Integrated Circuits and Signal Processing* 2017; **90**(3):523–538, doi: 10.1007/s10470-016-0919-6.
123. Tahir H, Nadeem M, Zafar N. Specifying electronic health system with Vienna Development Method specification language. *2015 National Software Engineering Conference, NSEC 2015*, Institute of Electrical and Electronics Engineers Inc.: Rawalpindi, Pakistan, 2015; 61–66, doi: 10.1109/NSEC.2015.7396346.
124. Azeem M, Ahsan M, Minhas N, Noreen K. Specification of e-Health system using Z: A motivation to formal methods. *2014 International Conference for Convergence of Technology, I2CT 2014*, Institute of Electrical and Electronics Engineers Inc.: Pune, India, 2014, doi: 10.1109/I2CT.2014.7092123.

125. Amato F, Moscato F. A model driven approach to data privacy verification in e-health systems. *Transactions on Data Privacy* 2015; **8**(3):273–296.
126. Kukec M, Ljubic S, Glavinic V. Need for usability and wish for mobility: Case study of client end applications for primary healthcare providers in Croatia. *Information Quality in e-Health: 7th Conference of the Workgroup Human-Computer Interaction and Usability Engineering of the Austrian Computer Society, USAB 2011, Graz, Austria, November 25-26, 2011. Proceedings*, *Lecture Notes in Computer Science*, vol. 7058, Springer Berlin Heidelberg, 2011; 171–190, doi: 10.1007/978-3-642-25364-5_15.
127. Frau S, Torabi-Dashti M. Integrated specification and verification of security protocols and policies. *Proceedings - IEEE Computer Security Foundations Symposium*, Cernay-la-Ville, France, 2011; 18–32, doi: 10.1109/CSF.2011.9.
128. Bugeaud F, Soulier E. A mereology-based ontology for services science: Example of an e-health service modelling. *Ontology, Conceptualization and Epistemology for Information Systems, Software Engineering and Service Science*, vol. 62 LNBIP, Springer Berlin Heidelberg, 2010; 123–134, doi: 10.1007/978-3-642-16496-5_9.
129. Pervez U, Hasan O, Latif K, Tahar S, Gawanmeh A, Hamdi M. Formal reliability analysis of a typical FHIR standard based e-Health system using PRISM. *2014 IEEE 16th International Conference on e-Health Networking, Applications and Services, Healthcom 2014*, Natal, Brazil, 2015; 43–48, doi: 10.1109/HealthCom.2014.7001811.
130. Addas R, Zhang N. Formal security analysis and performance evaluation of the linkable anonymous access protocol. *Information and Communication Technology*, vol. 8407 LNCS, Springer Berlin Heidelberg, 2014; 500–510, doi: 10.1007/978-3-642-55032-4_51.
131. Abdmeziem M, Tandjaoui D. An end-to-end secure key management protocol for e-health applications. *Computers and Electrical Engineering* 2015; **44**:184–197, doi: 10.1016/j.compeleceng.2015.03.030.
132. Kralj D, Konňar M, Tonković S. A survey on patients' privacy and safety protection with EMR applications in primary care. *IFMBE Proceedings*, vol. 37, 2011; 1132–1135, doi: 10.1007/978-3-642-23508-5_293.
133. Kralj D, Končar M, Tonković S. A methodology to assess experiences in implementing e-Health solutions in croatian family medicine. *Studies in Health Technology and Informatics*, vol. 165, 2011; 129–134, doi: 10.3233/978-1-60750-735-2-129.
134. Groot P, Hommersom A, Lucas P, Balser M, Schmitt J. Experiences in quality checking medical guidelines using formal methods. *Proceedings Verification and Validation of Software Systems (VVSS 2007) March 23, 2007*, Eindhoven, The Netherlands,, 2007; 164–78.
135. Balser M, Coltell O, Van Croonenborg J, Duelli C, Van Harmelen F, Jovell A, Lucas P, Marcos M, Miksch S, Reif W, *et al.*. Protocure: Supporting the development of medical protocols through formal methods. *Studies in Health Technology and Informatics*, vol. 101, 2004; 103–107, doi: 10.3233/978-1-60750-944-8-103.
136. Carneiro E, Maciel P, Callou G, Tavares E, Nogueira B. Mapping SysML state machine diagram to Time Petri Net for analysis and verification of embedded real-time systems with energy constraints. *Proceedings - International Conference on Advances in Electronics and Micro-electronics, ENICS 2008*, Valencia, Spain, 2008; 1–6, doi: 10.1109/ENICS.2008.19.
137. Cordeiro L, Fischer B, Chen H, Marques-Silva J. Semiformal verification of embedded software in medical devices considering stringent hardware constraints. *Proceedings - 2009 International Conference on Embedded Software and Systems, ICESS 2009*, Zhejiang, China, 2009; 396–403, doi: 10.1109/ICESS.2009.82.
138. May M, Shin W, Gunter C, Lee I. Securing the drop-box architecture for assisted living. *Proceedings of the Fourth ACM Workshop on Formal Methods in Security Engineering, FMSE'06. A workshop held in conjuction with the 13th ACM Conference on Computer and Communications Security, CCS'06*, Alexandria, Virginia, USA, 2006; 1–12, doi: 10.1145/1180337.1180338.
139. Ölveczky PC. Towards formal modeling and analysis of networks of embedded medical devices in Real-Time Maude. *Proc. 9th ACIS Int. Conf. Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, SNPD 2008 and 2nd Int. Workshop on Advanced Internet Technology and Applications*, Phuket, 2008; 241–248, doi: 10.1109/SNPD.2008.42.
140. Curran P, Norrie K. An approach to verifying concurrent systems-a medical information bus (MIB) case study. *Proceedings Fifth Annual IEEE Symposium on Computer-Based Medical Systems*, IEEE Comput. Soc. Press: Durham, NC, USA, 1992; 74–83, doi: 10.1109/CBMS.1992.244961.
141. Sloane E, Schrenker R. Conceptual design and resources for a general-purpose safety and performance Verification and Validation Toolkit (V2T) for life-critical Wireless Medical Device Networks (WMDN). *Annual International Conference of the Engineering in Medicine and Biology Society*, vol. 7 VOLS, IEEE: Shanghai, China, 2005; 178–181, doi: 10.1109/IEMBS.2005.1616371.
142. Cehlot V, Sloane E. Ensuring patient safety in wireless medical device networks. *Computer* April 2006; **39**(4):54–60, doi: 10.1109/MC.2006.125.
143. Bowen J, Reeves S. Modelling safety properties of interactive medical systems. *ACM SIGCHI symposium on Engineering interactive computing systems*, EICS '13, ACM: London, United Kingdom, 2013; 91–100, doi: 10.1145/2480296.2480314. URL `http://doi.acm.org/10.1145/2494603.2480314`.
144. Leucker M, Schmitz M, à Tellinghusen D. Runtime verification for interconnected medical devices. *International Symposium on Leveraging Applications of Formal Methods - ISoLA 2016*, vol. 9953 LNCS, Steffen B MT (ed.), Springer Verlag: Imperial, Corfu, Greece, 2016; 380–387, doi: 10.1007/978-3-319-47169-3_29.
145. Hoglund D. Validation and verification of WLAN medical devices. *Biomedical instrumentation & technology / Association for the Advancement of Medical Instrumentation* 2012; **Suppl**:79–83.
146. Decker N, Kuhn F, Thoma D. Runtime verification of web services for interconnected medical devices. *Proceedings - International Symposium on Software Reliability Engineering, ISSRE*, IEEE Computer Society: Naples, Italy, 2014; 235–244, doi: 10.1109/ISSRE.2014.16.
147. Pervez U, Mahmood A, Hasan O, Latif K, Gawanmeh A. Formal reliability analysis of Device Interoperability Middleware (DIM) based E-health system using PRISM. *2015 17th International Conference on E-Health Networking, Application and Services, HealthCom 2015*, Boston, MA, USA, 2016; 108–113, doi: 10.1109/HealthCom.2015.7454482.

148. Pervez U, Mahmood A, Hasan O, Latif K, Gawanmeh A. Improvement strategies for Device Interoperability Middleware using formal reliability analysis. *Scalable Computing* 2016; **17**(3):155–170, doi: 10.12694/scpe.v17i3.1178.
149. Bae WS, Han KH. An authentication system for safe transmission of medical information in U-health environment. *International Journal of Applied Engineering Research* 2014; **9**(20):7909–7918.
150. Simalatsar A, De Micheli G. Medical guidelines reconciling medical software and electronic devices: Imatinib case-study. *12th International Conference on Bioinformatics & Bioengineering (BIBE)*, IEEE: Larnaca, Cyprus, 2012; 19–24, doi: 10.1109/BIBE.2012.6399700.
151. Simalatsar A, You W, Gotta V, Widmer N, De Micheli G. Representation of medical guidelines with a computer interpretable model. *International Journal on Artificial Intelligence Tools* 2014; **23**(3):1460 003, doi: 10.1142/S0218213014600033.
152. Abbate A, Throckmorton A, Bass E. A Formal Task-Analytic Approach to Medical Device Alarm Troubleshooting Instructions. *IEEE Transactions on Human-Machine Systems* 2016; **46**(1):53–65, doi: 10.1109/THMS.2015.2494462.
153. Jacky J. Specifying a safety-critical control system in Z. *IEEE Transactions on Software Engineering* Feb 1995; **21**(2):99–106, doi: 10.1109/32.345826.
154. Bowen J, Reeves S. Modelling user manuals of modal medical devices and learning from the experience. *EICS'12 - Proceedings of the 2012 ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, Copenhagen, Denmark, 2012; 121–130, doi: 10.1145/2305484.2305505.
155. Niezen G, Eslambolchilar P. A Human Operator Model for Medical Device Interaction Using Behavior-Based Hybrid Automata. *IEEE Transactions on Human-Machine Systems* 2016; **46**(2):291–302, doi: 10.1109/THMS.2015.2487509.
156. Tolvanen JP, Djukić V, Popovic A. Metamodeling for medical devices: Code generation, model-debugging and run-time synchronization. *Procedia Computer Science*, vol. 63, 2015; 539–544, doi: 10.1016/j.procs.2015.08.382.
157. Berstel J, Reghizzi S, Roussel G, San Pietro P. A scalable formal method for design and automatic checking of user interfaces. *ACM Transactions on Software Engineering and Methodology* 2005; **14**(2):124–167, doi: 10.1145/1061254.1061256.
158. Gholami MR, Boucheneb H. Applying Formal Methods into Safety-Critical Health Applications. *Model-Based Safety and Assessment*, vol. 8822, Springer International Publishing, 2014; 195–208, doi: 10.1007/978-3-319-12214-4_15.
159. Butler MJ, Schewe K, Mashkoor A, Biró M ((eds.)). *Abstract State Machines, Alloy, B, TLA, VDM, and Z - 5th International Conference, ABZ 2016, Linz, Austria, May 23-27, 2016, Proceedings*, *Lecture Notes in Computer Science*, vol. 9675, Springer, 2016, doi: 10.1007/978-3-319-33600-8. URL `https://doi.org/10.1007/978-3-319-33600-8`.
160. Mashkoor A. The hemodialysis machine case study. *Abstract State Machines, Alloy, B, TLA, VDM, and Z - 5th International Conference, ABZ 2016, Linz, Austria, May 23-27, 2016, Proceedings*, Linz, Austria, 2016; 329–343, doi: 10.1007/978-3-319-33600-8_29.
161. Zhang Y, Jetley R, Jones PL, Ray A. Generic Safety Requirements for Developing Safe Insulin Pump Software. *Journal of Diabetes Science and Technology* Nov 2011; **5**(6):1403–1419. URL `http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3262707/`, 00008.