Protocure: Supporting the Development of Medical Protocols through Formal Methods

Michael Balser,[1] Oscar Coltell,[2] Joyce van Croonenborg,[3]
Christoph Duelli,[1] Frank van Harmelen,[4] Albert Jovell,[5] Peter Lucas,[6] Mar Marcos,[2]
Silvia Miksch,[7] Wolfgang Reif,[1] Kitty Rosenbrand,[3] Andreas Seyfang,[7] Annette ten Teije[4]

[1] *Universitaet Augsburg, Germany,* [2] *Universitat Jaume I, Spain,*
[3] *Dutch Institute for Healthcare improvement CBO, The Netherlands,*
[4] *Vrije Universiteit Amsterdam, The Netherlands,* [5] *Fundacio Biblioteca Josep Laporte, Spain,*
[6] *Katholieke Universiteit Nijmegen, The Netherlands,* [7] *Vienna University of Technology, Austria*
www.protocure.org

Abstract. Medical guidelines and protocols describe the optimal care for a specific group of patients and therefore, when properly applied, improve the quality of patient care. During the last decade, a large number of medical guidelines and protocols have been published. However, the work done on developing and disseminating them far outweighs the efforts on guaranteeing their quality. Indeed, anomalies like ambiguity and incompleteness are frequent in medical guidelines and protocols. An approach grounded on a formal representation, can answer these needs, as we have demonstrated in the Protocure project[1]. The Protocure II project will aim at integrating formal methods in the life cycle of guidelines.

1 Introduction

Evidence-based guidelines are becoming an important and indispensable part of quality health-care because of their potentials to improve quality and reduces cost of health-care.

It has been proved that adherence to guidelines and protocols may reduce health-care costs up to a 25% [2]. Nevertheless, in the current scenario of guideline development, dissemination and deployment, there is a major problem with clinical guidelines:

Recommendations can be outdated or not applicable in practice, because most guidelines are only revised every 3-5 years. In contrast with this, scientific and pragmatic knowledge is growing faster every year. At this moment, a guideline is a static document which cannot be modified easily. This problem has led to a future challenge, often referred to as *"living guidelines"*:

Update of the guidelines on a more continuous basis: clinical guidelines have to become flexible, adaptable documents. The aim is to develop guidelines which present up-to-date and state-of-the-art knowledge to practitioners. To make the approach of living guidelines possible, there must be some major changes in the guideline development process. Most guidelines are authored in an unstructured narrative form. Computer-based support depends on a more formal, structured representation, and can be used to address a number of challenges at all stages of the guideline life-cycle: modelling, authoring, dissemination, implementation and

[1] This work has been partially supported by the European Commissions IST program, under contract number IST-2001-33049 Protocure, IST-FP6-508794 Protocure II.

update. At this moment, guidelines are often multi-interpretable [3], incomplete and even inconsistent [4]. In the modelling phase of guidelines, methods have to be developed to support this process.

In summary: to enable living guidelines, they must be developed in a more structured way. Formal methods can be of help here. This will be an important first step to enhance further computer-based support of guidelines and protocols.

This paper presents in section 2 our experience with applying formal methods to medical guidelines, and continues in section 3 with work in progress. This section extends the use of formal methods to the development phase of guidelines, and therefore pays attention to ”living guidelines”.

2 Protocure: Improving medical protocols by formal methods

During the last decade, the approach of evidence-based medicine has given rise to an increasing number of medical practice protocols. However, the work done on developing and distributing protocols outweighs the efforts on guaranteeing their quality. Indeed, anomalies like ambiguity and incompleteness are frequent in medical protocols. Recent efforts have tried to address the problem of protocol improvement, but they are not sufficient since they rely on informal processes and notations. As a result, many practical protocols are still ambiguous or incomplete. Even when ambiguity and incompleteness are intentional, so that organisational or personal practices can take place, it is important to make them explicit. An approach, grounded on a formal representation of protocols, can answer these needs. The solution we suggest to the problem of quality improvement of protocols consists in the utilisation of formal methods. It supposes the definition of an adequate protocol representation language, the development of techniques for the formal analysis of protocols described in that language and, more importantly, the evaluation of the feasibility of the approach based on the formalisation and verification of real-life medical protocols. For the first two aspects we rely on earlier work, namely the Asbru language [7] for protocol description and the KIV interactive verification system [5]. The third aspect, i.e. the evaluation of the use of formal methods in the quality improvement of protocols, constitutes the main objective.

The steps with which we have carried out are: (1) Take two real-life reference protocols which cover a wide variety of protocol characteristics, (2) Formalise these reference protocols, (3) Check the formalisation through the verification of interesting protocol properties, (4) Determine how many errors can be uncovered. Step 4 can be considered our measure of success.

Our main objective, which was the assessment of the possibilities of protocol improvement by formal methods, has been attained using the methodology sketched in Figure 1. This figure illustrates the process. First we have selected two medical protocols which cover together a wide range of protocol characteristics (see Selection (1) step in figure 1). Then the two selected protocols have undergone a gradual transformation into a formal representation. Starting from the original texts (Informal Protocol (2) in the figure), the protocols have been first modelled in the Asbru language (Modelling (3) step) and then translated into the KIV formal representation (Formalisation of protocol (5) step). The results of this transformation process are, respectively, a collection of Asbru plans representing the protocol (see Asbru Plans (4)) and a set of KIV programs encoding these plans (see KIV Representation (6)). In order to make this formalisation possible, a formal semantics has been defined for the main parts of the Asbru language (see Formal Semantics (10)). This is a crucial point in the process, since the formal verification we aimed at is only possible with a precise semantics. In parallel to the above transformation, a number of interesting properties has been identified from an analysis of both the original protocols and their Asbru version (Identification of

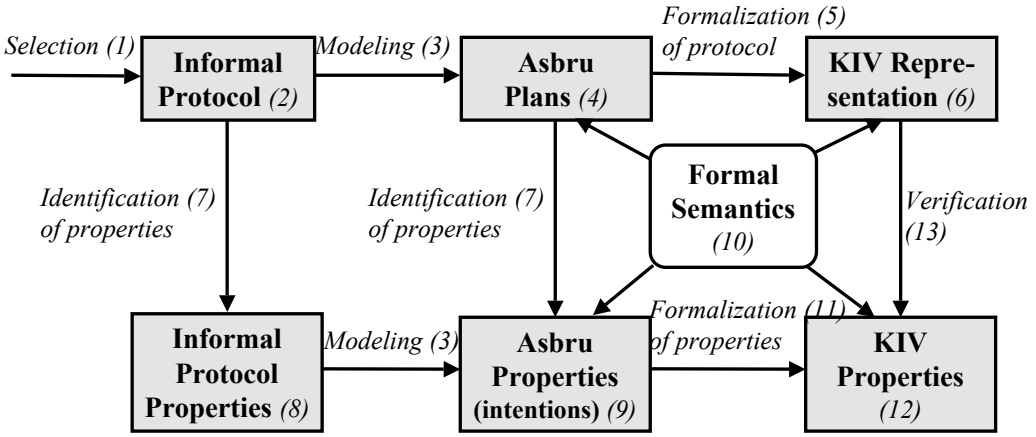

Figure 1: Process of protocol formalisation and verification in project IST-200133049 - Protocure.

properties (7) step). The result of this phase comprises both protocol-dependent properties (see Informal Protocol Properties (8)) and protocol-independent ones (see Asbru Properties (9)). Then we have selected a subset of properties from the previous list, and we have carried out the necessary proofs to verify them (Verification (13) step). Finally, the results of this verification phase have been presented for evaluation to a group of medical professionals, in order to assess the overall utility of our approach.

In accordance with this process, we have obtained the following results:

A consolidated formal language to model medical practice protocols. During the Asbru modelling of the protocols, a number of fundamental language constructs has been identified the so called Asbru-Light. The semantics of these core constructs has been clarified and formally specified.

Two protocols [1], [6] each both in Asbru and formalised in KIV. To give an idea of the size of the different formats: the original jaundice protocol is 10 pages, the Asbru code is 18 pages (40 plans, 30 pages in XML), the KIV code is 30 pages.

A list of different types of anomalies found during the Asbru modelling of the protocols. To give an idea of the extent of uncovered anomalies, some concrete numbers. In the case of jaundice protocol we found 1 ambiguity, 10 incompleteness anomalies, 6 inconsistencies and no redundancy. Regarding the diabetes protocol, we identified 4 ambiguities, 38 incompletenesses and 2 redundancies, but no inconsistency. [2]

A list of properties that medical protocols should verify. With the aim of carrying out a series of formal verification experiments, a number of interesting protocol properties has been identified.

Verification proofs for these protocols and properties using KIV. In some cases the properties have been confirmed. However, in other cases the verification was problematic, and we had to add assumptions describing the necessary conditions that made the properties true. In any case, the fact that these proofs have been completed shows that formal verification of protocols is feasible. Such proofs easily contains 1000 steps.

Evaluation by medical experts. In order to assess the success of Protocure project, the results of the verification phase have been presented for evaluation to a group of medical professionals. The interviewed experts think that the problems spotted in the protocols with the help of formal methods should be avoided, in particular from the point of view of guideline developers.

After having completed the whole formalisation and verification process for the two ref-

[2]We are aware that this task of interpretation of anomalies and absolute numbers must be carried out by medical practitioners with the relevant knowledge.

erence protocols we have illustrated that formalisation of guidelines is *doable* and *useful*.

The approach of Protocure can be summarised in the metaphor protocol $\approx$ program. Just as a program, a protocol is a set of instructions of how to behave in certain circumstances. Just as formal methods in Software Engineering are known to help improve the quality of programs, we have shown how these methods can also be successfully employed to improve the quality of protocols.

3 Protocure II: Integrating formal methods in the development process

The Protocure project focussed only on already existing protocols, and did not take into account the development process of these protocols. In order to meet the future challenge of turning guidelines into "living guidelines", Protocure II will extend the static metaphor of *"protocol $\approx$ program"* into a dynamic metaphor: *guideline and protocol development $\approx$ software engineering*. As a result, the focus of attention in Protocure II will be the incorporation of formal methods in the life-cycle of medical guidelines. Instead of aiming at developing methods to analyse only existing guidelines and protocols, we will concentrate on techniques and tools to support the whole guideline development process.

Our approach is based on a basic life-cycle model for medical guidelines and protocols: development, deployment, and maintenance. The main subjects we are working on are structured around this simple life-cycle model, including a study and enhance the life-cycle model itself:

Process model As in the design of software, the process of the design of guidelines starts with a requirement analysis with resulting specification documents, which gradually become more and more precise. Software engineering offers various methods and techniques to assist in the development of software systems. Some of these methods and techniques might be successfully deployed in the design and development of guidelines as well.

Development In this phase we work on the following issues:

Representation and modelling: To facilitate the coupling of the formal and informal version of a guideline, we will distinguish four different levels of representation: (i) Original knowledge, consisting of text books, journal articles, data collections (in various formats) and expert opinions captured in interviews. (ii) Intermediate representation, which structures the informal knowledge into pieces of knowledge with clear but informal semantics. (iii) Semi-formal representation, e.g. Asbru, which is executable but not precise enough to apply formal methods. (iv) Formal representation, e.g. KIV, which can be used as a basis for the formal verification of the guideline.

Transformation: As a consequence of the previous distinction of representation levels, the construction of a formal guideline can be viewed as a series of transformations of information. In order to obtain bi-directional links between the original and transformed versions at each of the steps, we will develop a mark-up tool which will allow the user to identify and relate pairs of fragments which are transcriptions of each other.

Visualisation: Each of the four representation levels mentioned above has its own visualisation needs, and requires adequate tools that answer these needs.

Libraries of design patterns for guideline development: One of our goal is to provide guideline developers with a number of design patterns that appear frequently in medical guidelines, which can be reused in the development of a new guideline.

Validation: Both medical guidelines and the Asbru language are complex, therefore, it is likely that mistakes are made during the modelling phase. As result the Asbru model might not reflect the original intent of the guideline. Thanks to the features of Asbru, it is possible to execute the model of the guideline using an adequate interpreter. Testing strategies will be developed.

Simultaneous development of guidelines and indicators: During the development of a guideline, from the informal knowledge to the formal model, it is possible to formulate properties with which the guideline is expected to comply. In formal terms, such properties should become proof obligations that must be fulfilled by the guideline under development.

Verification by theorem proving: Although we may find many errors through validation and testing, it is not possible to prove the absence of problems in this way. In order to ensure that the behaviour of the guideline is the desired one under any circumstances, we have to employ formal methods.

Verification by model checking: Interactive theorem proving is a very powerful technique, and often the only way to tackle certain verification problems. However, for certain restricted properties there exist tools, the so-called model checkers, that are powerful enough for a fully automatic check of the system.

Deployment In the deployment phase our focus is on refinement of guidelines and critiquing of guidelines.

Refinement: Medical guidelines and protocols contain more or less precise recommendations about appropriate healthcare for patients under specific clinical circumstances. An important issue here is the analysis of the different types of transformations that can be applied to a guideline for the development of a protocol, as well as the conditions under which they can be applied.

Critiquing: We compare the care given to the patient with the recommendations issued from the guideline. Given the patient data, we can check whether the physicians' actions follow a legal path in the guideline.

Maintenance In the scenario of "living guidelines", medical guidelines are continuously evolving over time. After changes have been made to a guideline, old tests and proofs are rendered obsolete, because they refer to an outdated version of the guideline. However, usually many parts of the guideline remain untouched, making it possible to keep part of the old tests and proofs. With regression testing techniques we can find out which tests are really affected by changes, and re-run only those. This is also possible in the case of proofs. We will focus on the development of techniques to facilitate the validation and verification of evolving guidelines, such as the reuse and replay of proofs, generation of counterexamples, and change management.

References

[1] AAP. American Academy of Pediatrics, Provisional Committee for Quality Improvement and Subcommittee on Hyperbilirubinemia. Practice parameter: management of hyperbilirubinemia in the healthy term newborn. *Pediatrics*, 94:558–565, 1994.

[2] P. Clayton and G. Hripsak. Decision support in healthcare. *International Journal of Biomedical Computing*, 39:59–66, 1995.

[3] M. Marcos, G. Berger, F. van Harmelen, A. ten Teije, H. Roomans, and S. Miksch. Using critiquing for improving medical protocols: harder than it seems. In S. Qualini, P. Barahona, and S. Andreassen, editors, *Proc. of the 8th European Conf. on AI in Medicine (AIME-01)*, number 2101 in LNAI, pages 431–441, Germany, 2001. Springer Verlag. ISBN 3-540-42294-3.

[4] M. Marcos, H. Roomans, A. ten Teije, and F. van Harmelen. Improving medical protocols through formalization: a case study. In *Proc. of the 6th Int. Conf. on Integrated Design and Process Technology (IDPT-02)*, 2002.

[5] W. Reif. The KIV-system: Systematic construction of verified software. In D. Kapur, editor, *Proc. 11 th Conference on Automated Deduction, LNAI 607*. Springer, 1992.

[6] G. Rutten, S. Verhoeven, R. Heine, W. de Grauw, P. Cromme, K. Reenders, E. van Ballegooie, and T. Wiersma. NHG-Standaard Diabetes Mellitus Type 2. *Huisarts en Wetenschap*, 42(2):67–84, 1999. First revision.

[7] Y. Shahar, S. Miksch, and P. Johnson. The Asgaard project: a task-specific framework for the application and critiquing of time-oriented clinical guidelines. *AIM*, 14:29–51, 1998.