The Role of Model Checking in Critiquing Based on Clinical Guidelines*

Perry Groot[1], Arjen Hommersom[1], Peter Lucas[1],
Radu Serban[2], Annette ten Teije[2], and Frank van Harmelen[2]

[1] Radboud Universiteit Nijmegen
[2] Vrije Universiteit Amsterdam

Abstract. Medical critiquing systems criticise clinical actions performed by a physician. In order to provide useful feedback, an important task is to find differences between the actual actions and a set of 'ideal' actions as described by a clinical guideline. In case differences exist, insight to which extent they are compatible is provided by the critiquing system. We propose a methodology for such critiquing, where the ideal actions are given by a formal model of a clinical guideline, and where the actual actions are derived from real world patient data. We employ model checking to investigate whether a part of the actual treatment is consistent with the guideline. Furthermore, it is shown how critiquing can be cast in terms of temporal logic, and what can be achieved by using model checking. The methodology has been applied to a clinical guideline of breast cancer in conjunction with breast cancer patient data.

1 Introduction

There is an increasing interest amongst researchers to develop computerised versions of clinical guidelines, which at the moment are still just documents, using one of the specialised guideline representation languages. The resulting computer-based guidelines can then act as a basis for the development of decision-support systems, which, thus, allow computer-based deployment of guidelines in a clinical setting. One possible application of such clinical decision-support systems is *critiquing*, i.e., to spot and analyse differences between the proposed actions taken by a medical doctor, and a set of 'ideal' actions as prescribed by the computerised guideline. As a computer-based clinical guideline is represented in a formal language, there is, also room for a formal underpinning of the various ways a guideline can be manipulated.

A natural way to formally describe the actions taken by a medical doctor in the management of the disease of a patient is offered by temporal logics. As a family of languages, logics make it possible to describe the meaning of the

---

various aspects of the disease and condition of the patient in a precise fashion. By the addition of temporal operators, *temporal* logic adds various notions of progress of the disease and sequencing of actions in time.

One way to look upon a patient and a patient's disease logically is as a concurrent system, i.e., as a system described in terms of states and state transitions in time. Model checking technology offers methods that allow one to analyse concurrent systems for their consistency. One can rely on an extensive collection of tools and techniques readily available. It is a well investigated technique for verification of systems that can be modelled by a finite transition system. However, model checking has been mainly applied to technical systems, such as hardware, software-based communication protocols, concurrent programs, etc. This raises the question when adopting this global view on the representation of diseases, patient conditions, and disease management actions, whether model checking can be used as a basis for critiquing. It is this question that is being explored in detail in this paper.

Model checking takes domain knowledge, called a system description, and sequences of actions as input. In this case, a formalised guideline is taken as a system description; the actions that have been performed on a specific patient are represented as a temporal formula. Model checking then involves investigating the consistency of the formalised guideline and actual treatment. The exploration of the use of model checking in the analysis of medical knowledge (guidelines and patient data) with the purpose of critiquing, is the innovative part of this work.

2   Approach

The common feature of a critiquing system is that the user of the system provides as input (1) a problem description (e.g., patient symptoms), and (2) a proposed solution (e.g., a treatment plan). This second input is what distinguishes critiquing systems from the more traditional expert systems, which only take a problem description as input [10,4]. The second input to a critiquing system, i.e., a proposed solution, is typically the output of an expert system.

In our approach of critiquing medical treatment plans using model checking, the input to the system consists of patient data and a treatment plan (cf. Fig. 1). Patient data consists of patient symptoms and test outcomes measured for the patient, whereas the treatment plan consists of all actions (to be) performed by the practitioner. As the critiquing process is difficult to accept by practitioners when they are continually interrupted to provide input to the system, both patient data and treatment plan are typically provided by electronic records. We will assume that these are given to the system as temporal logic formulas.

The critiquing system uses the patient data and treatment plan as specifications that need to be checked against a formal model of the guideline, i.e., a state transition system. When the specifications are consistent with the guideline model, no critique needs to be generated as the proposed treatment plan conforms with the guideline. In case an inconsistency is found between the specification and the guideline model, the specification is weakened to get insight to
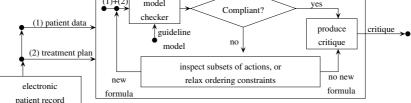
Fig. 1. Critiquing approach using model checking. Given patient data and a treatment plan as input (temporal specifications), the critiquing system uses a model checker to verify consistency w.r.t. to a guideline model (state transition system) to generate a critique (empty in case of compliance).

which extent the treatment plan is consistent with the guideline. There are two possible reasons for the incompatibility:

Non-compliant order: It is possible that each of the actions in the treatment plan can be applied to this patient, but only in a different order than the treatment plan proposes. This can be established by removing the order between some of the actions in the treatment plan.

Non-compliant actions: Another possibility is that, according to the guideline, some of the actions cannot be prescribed at all for the patient in question. This can be investigated by considering a subset of the actions in the treatment plan.

The approaches can be combined and lead to further insight into the nature of the detected inconsistency allowing the system to exploit these insights into a critique, which is then given to the practitioner.

3   Temporal Logic for Critiquing

In Subsection 3.1, the formal preliminaries of temporal logic are introduced. In Subsection 3.2, temporal logic is applied to critiquing and examples are provided.

3.1   Preliminaries

Temporal logic is a modal logic, where relationships between worlds in the usual possible-world semantics of modal logic is understood as time order. The logic that we use here for specifying properties of medical guidelines is a combination of Computation Tree Logic (CTL) and Linear Temporal Logic (LTL) [3].

In this paper we model a guideline as a Kripke structure M over a set of atomic propositions AP, which formally is defined as a four tuple $M = (S, S_0, R, L)$ where S is a finite set of states, $S_0 \subseteq S$ is the set of initial states, $R \subseteq S \times S$ is a total transition relation, and $L : S \to 2^{AP}$ is a function that labels each

state with the set of atomic propositions true in that state. A *path* in the model M from a state s is an infinite sequence $\pi = s_0 s_1 s_2 \dots$ such that $s_0 = s$ and $R(s_i, s_{i+1})$ holds for all $i \geq 0$. With π^i we denote the suffix of π starting at s_i, i.e., $\pi^i = s_i s_{i+1} s_{i+2} \dots$.

CTL uses atomic propositions, propositional connectives, *path quantifiers* and *temporal operators* for describing properties of *computation trees*, i.e., the tree that is formed by designating a state in the Kripke structure as the initial state and then unwinding the structure into an infinite tree according to the transition relation R with the initial state as root. This leads to two types of formulas: *state formulas*, which are true in a specific state, and *path formulas*, which are true along a specific path. A path formula is build up by applying one of the temporal operators to one or two state formulas. In this paper, the temporal operators used are $\mathbf{X}$, $\mathbf{G}$, $\mathbf{F}$, and $\mathbf{U}$. With $\mathbf{X}\varphi$ being true if φ holds in the next state, $\mathbf{G}\varphi$ if φ holds in the current state and all future states, $\mathbf{F}\varphi$ if φ holds in the current state or some state in the future, and $\varphi\mathbf{U}\psi$ if φ holds until eventually ψ holds. A state formula can be built inductively from atomic propositions, propositional connectives, and if f and g are path formulas, then $\mathbf{E}f$ and $\mathbf{A}f$ are state formulas. The path quantifiers $\mathbf{A}$ and $\mathbf{E}$ are used to specify that all or some of the paths starting at a specific state have some property.

The semantics of CTL is defined with respect to a Kripke structure M. Given a state formula f, the notation $M, s \models f$ denotes that f holds in state s of the Kripke structure M. Assuming that f_1 and f_2 are state formulas and g_1 and g_2 are path formulas, the relation $\models$ is defined inductively as shown in Fig. 2. The remaining syntax consisting of $\vee, \rightarrow, \mathbf{G}, \mathbf{A}$ can be defined as usual, i.e., $f_1 \vee f_2 \equiv \neg(\neg f_1 \wedge \neg f_2)$, $f_1 \rightarrow f_2 \equiv \neg f_1 \vee f_2$, $\mathbf{G}g \equiv \neg\mathbf{F}\neg g$, and $\mathbf{A}f \equiv \neg\mathbf{E}\neg f$.

In contrast to CTL, LTL provides operators for describing events along a single computation path. Each formula is of the form $\mathbf{A}f$, with f being a path formula, which is either an atomic proposition or inductively defined as $\neg f$, $f \vee g$, $f \wedge g$, $\mathbf{X}f$, $\mathbf{F}f$, $\mathbf{G}f$, $f\mathbf{U}g$ with f, g path formulas. This language can be evaluated on Kripke structures presented in Fig. 2.

$$
\begin{aligned}
M, s &\models p & &\Leftrightarrow p \in L(s) \\
M, s &\models \neg f_1 & &\Leftrightarrow M, s \not\models f_1 \\
M, s &\models f_1 \wedge f_2 & &\Leftrightarrow M, s \models f_1 \text{ and } M, s \models f_2 \\
M, s &\models \mathbf{E}g_1 & &\Leftrightarrow \text{there is a path } \pi \text{ from } s \text{ such that } M, \pi \models g_1 \\
M, \pi &\models f_1 & &\Leftrightarrow s \text{ is the first state of } \pi \text{ and } M, s \models f_1 \\
M, \pi &\models \neg g_1 & &\Leftrightarrow M, \pi \not\models g_1 \\
M, \pi &\models g_1 \wedge g_2 & &\Leftrightarrow M, \pi \models g_1 \text{ and } M, \pi \models g_2 \\
M, \pi &\models \mathbf{X}g_1 & &\Leftrightarrow M, \pi^1 \models g_1 \\
M, \pi &\models \mathbf{F}g_1 & &\Leftrightarrow \text{there exists a } k \geq 0 \text{ such that } M, \pi^k \models g_1 \\
M, \pi &\models g_1 \mathbf{U} g_2 & &\Leftrightarrow \text{there exists a } k \geq 0 \text{ such that } M, \pi^k \models g_2 \text{ and} \\
& & &\qquad \text{for all } 0 \leq j < k, M, \pi^j \models g_1
\end{aligned}
$$

Fig. 2. Semantics of temporal logic with f_1 and f_2 representing state formulas and g_1 and g_2 representing path formulas

3.2   Critiquing Formulas

Each path in the state transition system can be considered a patient who is given a treatment that is consistent with the recommendation described by the guideline. Global properties of the guideline can be checked using LTL formulas or CTL formulas starting with $\mathbf{A}$, for example, '$\mathbf{AF}$ radio-therapy', denotes that in each possible treatment, somewhere in the future radio-therapy is applied.

In the context of critiquing, CTL properties always start with an $\mathbf{E}$, i.e., it is established that *some* treatment path exists in the guideline where the proposed treatment is described. For example, abstracting from the patient, a treatment given by a sequence of actions $\alpha_1, \alpha_2, \ldots$ can then be represented as:

$$\mathbf{EF}(\alpha_1 \wedge \mathbf{EX}\,\mathbf{EF}\,(\alpha_2 \wedge \mathbf{EX}\,\mathbf{EF}\,(\ldots))) \tag{1}$$

i.e., in some treatment α_1 is done, and after that α_2, etc. In general, CTL model checking is more efficient than LTL model checking, however, in case we do not know the order between the actions, a CTL formula consists of a disjunction of each possible order of actions and considers the existence of each order. In case of n actions, with all order unknown, this leads to formulas of size $O(n \times 2^n)$. Similarly, when global properties of the treatment path are introduced, for example the state of the patient or the fact that some action *never* occurs, such knowledge becomes difficult to express. Assume for example a global property described by β, then Formula (1) must be rephrased to the rather complicated formula:

$$\mathbf{E}(\beta\,\mathbf{U}\,(\alpha_1 \wedge \beta \wedge \mathbf{EX}\,\mathbf{E}(\beta\,\mathbf{U}\,(\alpha_2 \wedge \beta \wedge \mathbf{EX}\,\mathbf{E}(\ldots \wedge \mathbf{EG}\,\beta)))))) \tag{2}$$

i.e., β holds until at some point α_1 and β (still) holds, after which β holds, etc.

Usually, the knowledge is reasonably complete and the global information is sparse, however, for a more succinct representation we can either use a more expressive logic such as CTL* [3] or consider LTL model checking. An approach using LTL is modular model checking [6], where the model is restricted using an LTL formula to those traces where the formula is valid. To prove the *existence* of a treatment in this approach, it is required to verify that the model restricted to a specification of a certain patient and treatment is not empty. Let φ be an LTL formula and $[\varphi]M\langle\perp\rangle$ denote that the set of LTL assertions φ leads to an empty model, i.e., φ describes a trace not present in the model. In contrast, if $[\varphi]M\langle\perp\rangle$ is shown to be false, then M can not be empty when restricted to φ proving that the trace described by φ exists in the model M. Formula (2) can thus be verified by showing that

$$[\mathbf{G}\beta \wedge \mathbf{F}(\alpha_1\mathbf{XF}(\alpha_2 \wedge \ldots))]M\langle\perp\rangle \tag{3}$$

is *false*. An additional benefit of this presention is that when order information is absent, the property is typically more intuitivly specified. Nonetheless, when there are few actions involved and much of the order information is present, CTL formulas are expected to be more efficient to verify.

4 Application of the Methodology to Breast Cancer

4.1 Design and Choice of Case Studies

The clinical guideline used is the Dutch breast cancer guideline[1] and was represented as a state transition system in Cadence SMV using the techniques and representation described in [2]. The models used here were developed as part of the Protocure-II project.[2] Patient data were obtained from the Dutch Comprehensive Cancer Centre South (CCC), a registry in the Netherlands used for cancer research, planning of services, and evaluation and implementation of guidelines. The data collected concerns breast cancer patients treated in the period January 2003 - June 2004, when the guideline was applicable, and therefore suitable for compliance checks with the guideline. Each patient record consists of 269 variables, which includes information about the diagnosis and treatment.

The patient data from the registry could, in principle, directly be used for critiquing w.r.t. to the guideline. However, matching such data records to the terminology of the guideline is hard [7] and differs from the course commonly followed in medicine. In medical literature, specific patient cases, called *casuistics*, are frequently discussed in detail to gain insight into the way the patient's disease was managed. These papers follow a long standing tradition and are seen as part of the 'education permanente' of the medical profession. Critiquing in this paper was therefore done casuistically by having the CCC patient data interpreted by medical experts who provided a direct mapping from the patient data in the registry to the guideline. Subsection 4.2 presents in more detail a case-study in critiquing using the casuistic interpretation of the CCC data.

A second case-study is presented in Subsection 4.3, which was obtained from the New South Wales Breast Cancer Institute, Australia.[3] These studies have been developed from the casuistic point of view to "allow clinicians, health professionals and members of the public to examine and understand some of the controversial and difficult aspects of breast cancer management". They are therefore more detailed than the patient data collected by the registry and are more suitable for an investigation of critiquing from a clinical point of view.

4.2 Case Study 1: Ductal Carcinoma in Situ

The steps of critiquing on one specific patient derived from the data, and subsequently interpreted by medical experts, is illustrated here. The diagnosis and treatment is summarised in Fig. 3. It can be said that this is a rather typical patient as it is a patient with one of the most frequent diagnoses in the data records. The following property describes the treatment sequence that our example patient has undergone. *"For a patient with diagnosis Ductal Carcinoma In Situ (DCIS), the following sequence of states is possible: the treatment starts, then axillary staging by sentinel node is activated, after which breast conserving*

---

[1] CBO: Richtlijn Behandeling van het mammacarcinoom, van Zuiden, 2002

[2] Breast cancer model can be obtained from `http://www.protocure.org`

[3] `http://www.bci.org.au/medical/caseindex.htm`

> **Medical condition**: 79 years-old woman. Lesion of right breast: carcinoma in-situ with size between 1 and 2 cm. Two lymph nodes investigated and none positive.
> **Treatment**: sentinel node biopsy + breast-conserving surgery without axillary clearance.

Fig. 3. Description of patient in conjunction with the prescribed treatment

therapy is activated". To specify and then verify that breast conserving therapy (denoted bct) can take place after axillary staging by sentinel node procedure (denoted asbSN), the following CTL formula is used:

$$\mathbf{EF}(\text{DCIS} \wedge \mathbf{EX}\,\mathbf{EF}(\text{asbSN} \wedge \mathbf{EX}\,\mathbf{EF}\,\text{bct}))$$

A more strict formula could be obtained by assuming that the diagnosis DCIS, holds up to the moment of breast conserving therapy. However, this property stated above turns out to be false as it is, i.e., this treatment is non-compliant with respect to the guideline. In other words, according to the model of the guideline describing the treatment of DCIS, the sequence of actions performed by the doctor is incorrect for this patient. It could also be explained by the fact that, according to the model, at least one of the two actions in patient treatment should not be started, or they should be started in a different sequence. To identify this inconsistency, we reduce the actions that are being performed. If we reduce the sequence to only one action, then both actions are found possible, as shown by the following property (corresponding to the case when only 'bct' is activated as part of the DCIS treatment):

$$\mathbf{EF}(\text{DCIS} \wedge \mathbf{EX}\,\mathbf{EF}\,\text{bct})$$

The new conclusion is that under these circumstances the two actions cannot be activated in this sequence, or the ordering should be reversed.

In the experiment on the seven fairly prototypical patient-cases that can be found in the Dutch CCC data-set, some deviation was found between the guideline and each of the seven prototypical cases. Interestingly, for three of these, some differences could indeed be explained by looking at the new 2004 revision of the guideline.

4.3   Case Study 2: Infiltrating Ductal Carcinoma

For the second case study we have more elaborate information available. It concerns a patient who is a female with a lump in the 3 o'clock position of the right breast and a second lump just above this. No palpable axillary nodes or other abnormalities were found. The mammography revealed no focal mass, grouped microcalcifications, or anatomic distortion. Finally, the histopathology showed two lesions: both infiltrating duct carcinoma, 20mm in size, and with similar morphology. The sentinel nodes were mapped using lymphoscintigraphy and a

biopsy was taken of a right axillary lymph node and an internal mammary node (the sentinel node procedure). In the right axillary lymph node, no malignancy was found. However, in the internal mammary node, metastatic carcinoma was identified. The treatment consisted of a total mastectomy of the right breast with immediate reconstruction. The axilla was treated by means of an axillary clearance and re-section of two further internal mammary nodes at higher levels (these were sampled partly because of the original pathology finding and partly because of ready access to the IMC).

The vocabulary of the guideline does not include the term 'infiltrating ductal carcinoma', but rather discusses 'operable invasive breast cancer' (OIBC). According to the guideline, operable invasive breast cancer is defined as T1-2 N0-1 M0, i.e., a tumour smaller than 5cm, with maximally one lymph node positive, and no distant metastasis. On basis of information provided by the diagnostic tests, the patient can be considered part of this patient group. Each of the three interventions (sentinel node procedure, mastectomy, and axillary clearance) can be mapped to terms found in the guideline. This can be done with reasonable confidence, however, some details have to be ignored such as the re-section of the internal mammary nodes as part of the axillary clearance, as this part of the treatment is not mentioned in the guideline. With respect to the order between interventions, it is only clear that the sentinel node procedure (asbSN) is performed before the other two interventions.

The treatment can again be critiqued using a CTL proof obligation, but as some of the information is missing here, we illustrate critiquing using modular model checking. The proof obligation is then described by $[\varphi]M\langle\bot\rangle$ where

$$\varphi = \mathbf{G}\,\mathrm{OIBC} \wedge \mathbf{F}\,(\mathrm{asbSN} \wedge \mathbf{X}(\mathbf{F}\,\text{axillary-clearance} \wedge \mathbf{F}\,\mathrm{mastectomy}))$$

The proof obligation $[\varphi]M\langle\bot\rangle$ is true, showing that this combination of interventions is *not* possible (cf. Subsection 3.2). The reason for this can be further analysed by removing one of the order constraints yielding

$$\varphi' = \mathbf{G}\,\mathrm{OIBC} \wedge \mathbf{F}\,\mathrm{asbSN} \wedge \mathbf{F}\,\text{axillary-clearance} \wedge \mathbf{F}\,\mathrm{mastectomy}$$

As $[\varphi']M\langle\bot\rangle$ is true, the formula φ' is further weakened by removing one of the interventions from the conjunct. This results in three new proof obligations, showing that the guideline model does not contain a trace with both a sentinel node procedure and axillary clearance for this patient, while all other combinations appear to be possible. Thus, the conclusion is that the combination of actions that are being prescribed is non-compliant with respect to the guideline.

5   Related Work

The use of the term *critiquing* to describe a system that criticises the solution provided by a human can be attributed to Miller [8], who developed his ATTENDING system for critiquing anaesthesia management. Although critiquing has first been used for evaluating medical treatment plans, since then

it has been applied to a wide variety of problems such as engineering design, decision making, word processing, knowledge base acquisition, and software engineering [10]. At the end of the 1990s, when several guideline representation languages were introduced, critiquing using guidelines became a topic of interest, e.g., the approach of Shahar *et al.* [9]. In contrast with previous work, in this approach the patient states are considered for critiquing, besides the physician's actions. Advani *et al.* [1] argued that a critiquing system should adjust its critique for cases when the physician's actions are following the spirit and overall goals or intentions of the guideline designers, even though the actions deviate from the guideline. However, in [7], a case study showed that intentions of the protocol are often implicit and moreover, the intentions reported by experts almost always differ, which makes it hard to model. Recently, there was some progress to overcome this difficulty [11], which might be interesting to integrate in our proposed methodology. Using model checking for verifying properties of formalised medical guidelines is very recent [12,2].

6 Conclusions

The main conclusion of this work is that it is, in principle, possible to use model checking on formalised models in order to critique medical guidelines against patient data. We have shown how critiquing can be characterised in temporal logic and have applied this to a case study on the treatment of breast cancer. The strong aspect of this technology is the high degree of automation as compared to theorem proving, making it suitable for deployment in a critiquing system.

Model checking provides additional value to a simulation-based critiquing of an operational version of the guideline. Such critiquing based on running the operational guideline model through an interpreter only checks the consistency of a patient record against a single trace through the guideline (namely, the one chosen by the interpreter), while model checking compares the patient record against all possible traces through the guideline. This difference is crucial when the guideline is under-specified [5], which is usually the case, and therefore contains non-deterministic choices between treatments.

The fully automated nature of model checking also brings a weakness with it: model checking only *detects* inconsistencies, but does not contribute to the interpretation of the inconsistency. In general, model checking can construct a counter-example illustrating the inconsistency, which is often a very good guide towards tracing its source. However, this only works when model checking global properties, i.e., properties dealing with all possible treatment paths, while in Section 3 we argue that critiquing deals with formulas that establish the existence of a single treatment, thereby making it impossible for the model checker to generate a counter-example. In this paper, we have proposed some general strategies to deal with this (repeated experiments with weaker specifications by relaxing order constraints and by removing actions).

A general conclusion with respect to the breast cancer case study that can be drawn is that a closer correspondence is needed between the processes of

guideline construction and data-collection. In fact, this is currently already being partially implemented by the Dutch Institute of Healthcare Improvement: newly constructed guidelines are currently being equipped with a data-collection dictionary, which will ensure the correspondence between collected data and guideline terminology.

Even though the steps in the analysis of the case studies was done manually, it is not difficult to see how to automate this process since the temporal formulas could be generated mechanically. A more challenging question is how to use the result of this process for the construction of a human readable critique. In evidence-based guidelines, explanation and references are often provided, however, formal models of guidelines often abstract from this information making it difficult to provide elaborate information to the practitioner. This is an interesting topic for future research.

References

1. Advani, A., Lo, K., Shahar, Y.: Intention-based critiquing of guideline-oriented medical care. In: Proceedings of AMIA Annual Symposium, pp. 483–487 (1998)
2. Bäumler, S., Balser, M., Dunets, A., Reif, W., Schmitt, J.: Verification of medical guidelines by model checking – a case study. In: Valmari, A. (ed.) Model Checking Software. LNCS, vol. 3925, pp. 219–233. Springer, Heidelberg (2006)
3. Clarke, E.M., Grumberg, O., Peled, A.D.: Model Checking. MIT Press, Cambridge, Massachusetts, London, England (2001)
4. Gertner, A.S.: Critiquing: effective decision support in time-critical domains. PhD thesis, Dept. of Computer & Information Science, University of Pennsylvania (1995)
5. Hommersom, A.J., Groot, P., Lucas, P.J.F., Marcos, M., Martinez-Salvador, B.: A constraint-based approach to medical guidelines and protocols. In: ECAI 2006 WS – AI techniques in healthcare: evidence based guidelines and protocols (2006)
6. Kupferman, O., Vardi, M.Y.: Modular model checking. In: de Roever, W.-P., Langmaack, H., Pnueli, A. (eds.) COMPOS 1997. LNCS, vol. 1536, pp. 381–401. Springer, Heidelberg (1998)
7. Marcos, M., Berger, G., van Harmelen, F., ten Teije, A., Roomans, H., Miksch, S.: Using critiquing for improving medical protocols: Harder than it seems. In: 8th European Conference on Artificial Intelligence in Medicine, pp. 431–441 (2001)
8. Miller, P.: A critiquing approach to Expert Computer Advice: ATTENDING. Pittman Press, London (1984)
9. Shahar, Y., Miksch, S., Johnson, P.: A task-specific ontology for the application and critiquing of time oriented clinical guidelines. In: Proceedings of the sixth Conference on Artificial Intelligence in Medicine in Europe, pp. 51–61 (1997)
10. Silverman, B.G.: Survey of expert critiquing systems: Practical and theoretical frontiers. Communications of the ACM 35(4), 106–127 (1992)
11. Sips, R., Braun, L., Roos, N.: Applying formal medical guidelines for critiquing. In: ECAI 2006 WS – AI techniques in healthcare: evidence based guidelines and protocols (2006)
12. Terenziani, P., Giodano, L., Bottrighi, A., Montani, S., Donzella, L.: SPIN model checking for the verifcation of clinical guideline. In: ECAI 2006 WS – AI techniques in healthcare: evidence-baded guidelines and protocols (2006)