Automated Theorem Proving for Quality-checking Medical Guidelines*

Arjen Hommersom, Peter Lucas, and Patrick van Bommel
Institute for Computing and Information Sciences
Radboud Universiteit Nijmegen
The Netherlands
{arjenh,peterl,pvb}@cs.ru.nl

Abstract

Requirements about the quality of medical guidelines can be represented using schemata borrowed from the theory of abductive diagnosis, using temporal logic to model the time-oriented aspects expressed in a guideline. Previously we have shown that these requirements can be verified using interactive theorem proving techniques [HLB04]. In this paper, we investigate how this approach can be mapped to the facilities of a resolution-based theorem prover, OTTER, and a complementary program that searches for finite models of first-order statements, MACE-2. It is shown that the reasoning that is required for checking the quality of a guideline can be mapped to such fully automated theorem-proving facilities. The medical quality of an actual guideline concerning diabetes mellitus 2 is investigated in this way.

1 Introduction

Health-care is becoming more and more complicated at an astonishing rate. On the one hand, the number of different patient management options has risen considerably during the last couple of decades, whereas, on the other hand, medical doctors are expected to take decisions balancing benefits for the patient against financial costs. There is a growing trend within the medical profession to believe that clinical decision-making should be based as much as possible on sound scientific evidence; this has become known as *evidence-based medicine* [Woo00]. Evidence-based medicine has given a major impetus to the development of guidelines, documents offering a detailed description of steps that must be taken and considerations that must be taken into account by health-care professionals in managing a disease in a patient, to avoid substandard practices or outcomes. Their general aim is to promote standards of medical care.

Researchers in artificial intelligence (AI) have picked up on these developments [FD00, OMGM98], and some of them, for example in the Asgaard project [SMJ98], are involved in the design of computer-oriented languages, tools and systems that support the design and deployment of medical guidelines. AI researchers see guidelines as

*This work has been partially supported by the European Commission's IST program, under contract number IST-FP6-508794 PROTOCURE II.

good real-world examples of highly structured, systematic documents that are amenable to formalisation. Previously, it was shown that for reasoning about models of medical knowledge, for example in the context of medical expert systems [Luc93], automated reasoning techniques (e.g., [Rob65, WOLB84]) are a practical option.

There are two approaches to checking the quality of medical guidelines: (1) the *object-level* approach amounts to translating a guideline to a formal language, such as Asbru [SMJ98], and next applying techniques from program verification to the resulting representation in establishing whether certain domain-specific properties hold; (2) the *meta-level* approach, which consists of formalising general properties to which a guideline should comply, and then investigating whether this is the case. Here we are concerned with the meta-level approach to guideline-quality checking. For example, a good-quality medical guideline regarding treatment of a disorder should preclude the prescription of redundant drugs, or advise against the prescription of treatment that is less effective than some alternative.

Such a meta-level approach corresponds to reasoning that occurs during the process of *designing* medical guidelines and therefore such checks could be valuable. The design of a guideline can be seen as a very complex process where formulation of knowledge and construction of conclusions and corresponding recommendations are intermingled. This makes it cumbersome to do *interactive* verification of hypotheses concerning the optimal recommendation during the construction of such a guideline, because guideline developers do not generally have the necessary background in formal methods to construct such proofs interactively. Automated theorem proving on a language could therefore be potentially more useful for supporting the guideline development process.

The goal of the research described here was to establish how feasible it is to implement such meta-reasoning techniques in existing tools for automated deduction. We will show that it is indeed possible to explore the route from informal medical knowledge to a logical formalisation and automated verification. Previously, we have shown that the theory of abductive diagnosis can be taken as a foundation for the formalisation of quality criteria of a medical guideline [Luc03] and that these can be verified using (interactive) program verification techniques [HLB04]. In this paper, we provide an alternative to this approach by translating this formalism, a restricted part of temporal logic, to standard first order logic. We will show that, because of the restricted language we used for the formalisation of the object knowledge, the translation is a relatively simple fragment of first-order logic and is therefore amenable to automated reasoning techniques.

The paper is organised as follows. In the next section, we start by explaining what medical guidelines are, and a method for formalising guidelines by temporal logic is briefly reviewed. In Section 3 the formalisation of guideline quality using a meta-level scheme which comes from the theory of abductive diagnosis is described. The guideline on the management of diabetes mellitus type 2 that has been used in the case study is given attention in Section 4, and a formalisation of this is given as well. An approach to checking the quality of this guideline using the deductive machinery offered by OTTER and MACE-2 is presented in Section 5. Finally, Section 6 discusses what has been achieved, the advantages and limitations of this approach are brought into perspective and future research plans are mentioned.

- Step 1: diet
- Step 2: if Quetelet Index (QI) $\leq$ 27, prescribe a sulfonylurea drug; otherwise, prescribe a biguanide drug
- Step 3: combine a sulfonylurea drug and biguanide (replace one of these by a α-glucosidase inhibitor if side-effects occur)
- Step 4: one of the following:
  - oral antidiabetics and insulin
  - only insulin

Figure 1: Tiny fragment of a clinical guideline on the management of diabetes mellitus type 2. If one of the steps $k = 1, 2, 3$ is ineffective, the management moves to step $k + 1$.

2 Preliminaries

2.1 The Design of Medical Guidelines

The design of a medical guideline is far from easy. Firstly, the gathering and classification of the scientific evidence underlying and justifying the recommendations mentioned in a guideline is time consuming, and requires considerable expertise in the medical field concerned. Secondly, medical guidelines are very detailed. Making sure that all the information contained in the guideline is complete for the guideline's purpose, and based on sound medical principles, is hard work. An example of a tiny portion of a guideline is shown in Figure 1; it is part of the guideline for general practitioners about the treatment of diabetes mellitus type 2. This guideline fragment is used in this paper as a running example.

One way to use formal methods in the context of guidelines is to automatically verify whether a medical guideline fulfills particular properties, such as whether it complies with quality *indicators* as proposed by health-care professionals [MBtTvH02]. For example, using particular patient assumptions such as that after treatment the levels of a substance are dangerously high or low, it is possible to check whether this situation does or does not violate the guideline. However, verifying the effects of treatment as well as examining whether a developed medical guideline complies with global criteria, such as that it avoids the prescription of redundant drugs, or the request of tests that are superfluous, is difficult to impossible if only the guideline text is available. Thus, the capability to check whether a guideline fulfills particular medical objectives may require the availability of more medical knowledge than is actually specified in a medical guideline, i.e., *background knowledge* is required.

Table 1: Used temporal operators; t stands for a time instance.

Notation	**Interpretation**	**Formal semantics**
$\mathsf{H}\varphi$	φ has always been true in the past	$t \vDash \mathsf{H}\varphi \Leftrightarrow \forall t' < t : t' \vDash \varphi$
$\mathsf{G}\varphi$	φ is true now and at all future times	$t \vDash \mathsf{G}\varphi \Leftrightarrow \forall t' \geq t : t' \vDash \varphi$

2.2 Using Temporal Logic for Guideline Representation

As medical management is a time-oriented process, diagnostic and treatment actions described in guidelines are performed in a temporal setting. It has been shown previously that the step-wise, possibly iterative, execution of a guideline, such as the example in Figure 1, can be described precisely by means of temporal logic [MBtTvH02]. This is a modal logic, where relationships between worlds in the usual possible-world semantics of modal logic is understood as time order, i.e., formulae are interpreted in a *temporal structure* $\mathcal{F} = (\mathbb{T}, <, I)$. We will assume that the progression in time is *linear*, i.e., $<$ is a strict linear order. For the representation of the medical knowledge involved it appeared to be sufficient to use rather abstract temporal operators, as proposed in literature [Luc03]. The language of standard logic, with equality and unique names assumption, is augmented with the modal operators G, H, P, and F, where the temporal semantics of the first two operators is defined in Table 1. The last two operators are simply defined in terms of the first two operators:

$$\vDash \mathsf{P}\varphi \leftrightarrow \neg\mathsf{H}\neg\varphi \quad \text{(some time in the past)}$$
$$\vDash \mathsf{F}\varphi \leftrightarrow \neg\mathsf{G}\neg\varphi \quad \text{(some time in the future)}$$

This logic offers the right abstraction level to cope with the nature of the temporal knowledge in medical guidelines required for our purposes. However, more fine-grained temporal operators can be added if needed. For a full axiomatisation of this logic, see Ref. [Tur85].

3 Application to Medical Knowledge

It is assumed that two types of knowledge are involved in detecting the violation of good medical practice:

- Knowledge concerning the (patho)physiological mechanisms underlying the disease, and the way treatment influences these mechanisms. The knowledge involved could be causal in nature, and is an example of *object-knowledge*.
- Knowledge concerning good practice in treatment selection; this is *meta-knowledge*.

Below we present some ideas on how such knowledge may be formalised using temporal logic (cf. [Luc95] for earlier work).

We are interested in the prescription of drugs, taking into account their mode of action. Abstracting from the dynamics of their pharmacokinetics, which are normally modelled using differential equations, this can be formalised in logic as follows:

$$(\mathsf{G}\, d \wedge r) \rightarrow \mathsf{G}(m_1 \wedge \cdots \wedge m_n)$$

where d is the name of a drug or possibly of a group of drugs indicated by a predicate symbol (e.g. SU(x), where x is universally quantified and 'SU' stands for sulfonylurea drugs, such as Tolbutamid), r is a (possibly negative or empty) *requirement* for the drug to take effect, and m_k is a mode of action, such as decrease of release of glucose from the liver, which holds at all future times.

The modes of action m_k can be combined, together with an *intention* n (achieving normoglycaemia, i.e., normal blood glucose levels, for example), a particular patient *condition* c, and *requirements* r_j for the modes of action to be effective:

$$(\mathsf{G}m_{i_1} \wedge \cdots \wedge \mathsf{G}m_{i_m} \wedge r_1 \wedge \cdots \wedge r_p \wedge \mathsf{H}c) \rightarrow \mathsf{G}n$$

In both formulas the antecedent is strong. For example, a drug should *always* be applied to conclude that a certain mode of actions occurs. In a strict sense, this formulation is unrealistic, but the idea is that the time points that the modalities refer to are finite and refers to the relevant information about the patient's disease. This imprecise information is enough to be able to verify a number of quality criteria, which will be shown below.

Good practice medicine can then be formalised as follows. Let $\mathcal{B}$ be background knowledge, $T \subseteq \{d_1, \ldots, d_p\}$ be a set of drugs, C a collection of patient conditions, R a collection of requirements, and N a collection of intentions which the physician has to achieve. A set of drugs T is a *treatment* according to the theory of abductive reasoning if [Poo90, Luc97]:

(M1) $\mathcal{B} \cup \mathsf{G}T \cup C \cup R \not\models \bot$ (the drugs do not have contradictory effects), and

(M2) $\mathcal{B} \cup \mathsf{G}T \cup C \cup R \models N$ (the drugs handle all the patient problems intended to be managed)

If in addition to (1) and (2) condition

(M3) $O_\varphi(T)$ holds, where O_φ is a meta-predicate standing for an optimality criterion or combination of optimality criteria φ,

then the treatment is said to be *in accordance with good-practice medicine*. A typical example of this is subset minimality $O_\subset$:

$$O_\subset(T) \equiv \forall T' \subset T : T' \text{ is not a treatment according to (1) and (2)}$$

i.e., the minimum number of effective drugs are being prescribed. For example, if $\{d_1, d_2, d_3\}$ is a treatment that satisfies condition (3) in addition to (1) and (2), then the subsets $\{d_1, d_2\}$, $\{d_2, d_3\}$, $\{d_1\}$, and so on, do not satisfy conditions (1) and (2). In the context of abductive reasoning, subset minimality is often used in order to distinguish between various solutions; it is also referred to in literature as *Occam's razor*. Another definition of the meta-predicate O_φ is in terms of minimal cost O_c:

$$O_c(T) \equiv \forall T', \text{with } T' \text{ a treatment: } c(T') \geq c(T)$$

where $c(T) = \sum_{d \in T} \mathit{cost}(d)$; combining the two definitions also makes sense. For example, one could come up with a definition of $O_{\subset,c}$ that among two subset-minimal treatments selects the one that is the cheapest in financial or ethical sense.

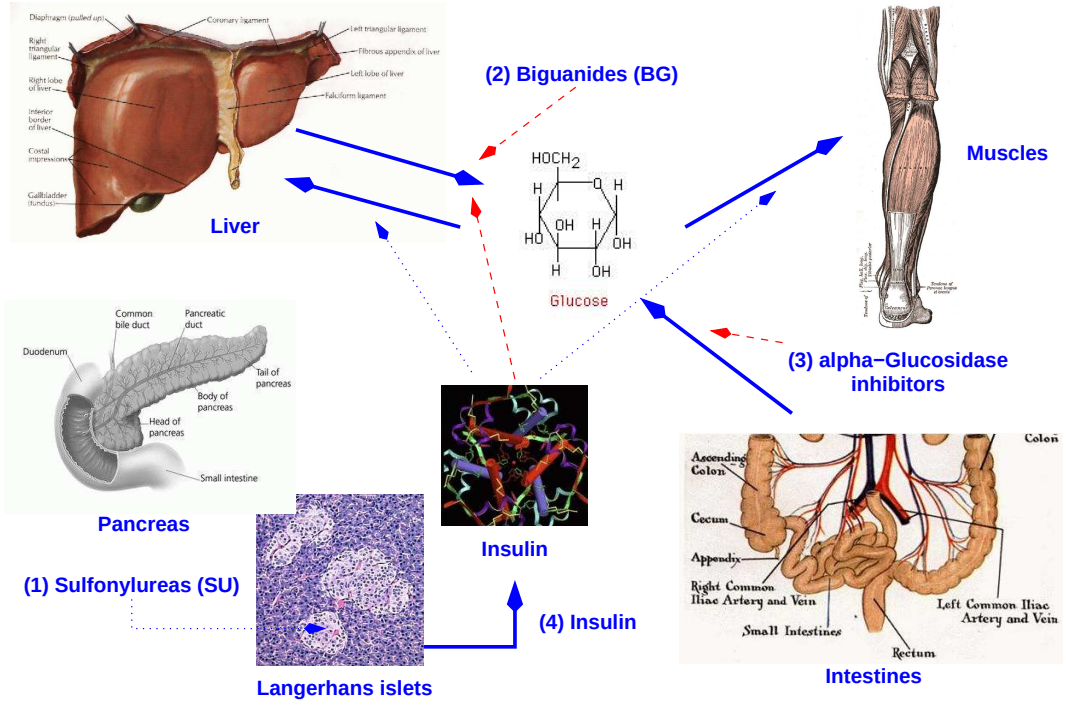

Figure 2: Summary of drugs and mechanisms controlling the blood level of glucose; − − →: inhibition, · · · · · →: stimulation.

4 Management of Diabetes Mellitus Type 2

4.1 Diabetes Type 2 Background Knowledge

It is well known that diabetes type 2 is a very complicated disease. Here we focus on the derangement of glucose metabolism in diabetic patients; however, even that is nontrivial. To support non-expert medical doctors in the management of this complicated disease in patients, access to a guideline is really essential.

One would expect that as this disorder is so complicated, the diabetes mellitus type 2 guideline is also complicated. This, however, is not the case, as may already be apparent from the guideline fragment shown in Figure 1. This indicates that much of the knowledge concerning diabetes mellitus type 2 is missing from the guideline, and that without this background knowledge it will be impossible to spot the sort of flaws we are after. Thus, the conclusion is that a deeper biological analysis is required, the results of which are presented below.

Figure 2 summarises the most important mechanisms and drugs involved in the control of the blood level of glucose. The protein hormone insulin, which is produced by the *B cells* in the Langerhans islets of the *pancreas*, has the following major effects:

- it increases the uptake of glucose by the liver, where it is stored as glycogen, and inhibits the release of glucose from the liver;
- it increases the uptake of glucose by insulin-dependent tissues, such as muscle and adipose tissue.

At some stage in the natural history of diabetes mellitus type 2, the level of glucose in the blood is too high (hyperglycaemia) due to the decreased production of insulin by the B cells.

Treatment of diabetes type 2 consists of:

- Use of *sulfonylurea* (SU) drugs, such as tolbutamid. These drugs stimulate the B cells in producing more insulin, and if the cells are not completely exhausted, the hyperglycaemia can thus be reverted to normoglycaemia (normal blood glucose levels).
- Use of *biguanides* (BG), such as metformin. These drugs inhibit the release of glucose from the liver.
- Use of *α-glucosidase inhibitors*. These drugs inhibit (or delay) the absorption of glucose from the intestines. We omit considering these drugs in the following, as they are only prescribed when treatment side-effects occur.
- Injection of *insulin*. This is the ultimate, causal treatment.

The background knowledge concerning the (patho)physiology of the glucose metabolism as summarised above is formalised using temporal logic, and kept as simple as possible. The specification is denoted by $\mathcal{B}_{\mathrm{DM2}}$:

(1) $\mathsf{G}\,\mathrm{Drug}(\mathit{insulin}) \rightarrow \mathsf{G}(\mathit{uptake}(\mathit{liver}, \mathit{glucose}) = \mathit{up} \wedge \mathit{uptake}(\mathit{peripheral\text{-}tissues}, \mathit{glucose}) = \mathit{up})$

(2) $\mathsf{G}(\mathit{uptake}(\mathit{liver}, \mathit{glucose}) = \mathit{up} \rightarrow \mathit{release}(\mathit{liver}, \mathit{glucose}) = \mathit{down})$

(3) $(\mathsf{G}\,\mathrm{Drug}(\mathrm{SU}) \wedge \neg \mathit{capacity}(\mathit{B\text{-}cells}, \mathit{insulin}) = \mathit{exhausted}) \rightarrow \mathsf{G}\mathit{secretion}(\mathit{B\text{-}cells}, \mathit{insulin}) = \mathit{up}$

(4) $\mathsf{G}\,\mathrm{Drug}(\mathrm{BG}) \rightarrow \mathsf{G}\mathit{release}(\mathit{liver}, \mathit{glucose}) = \mathit{down}$

(5) $(\mathsf{G}\mathit{secretion}(\mathit{B\text{-}cell}, \mathit{insulin}) = \mathit{up} \wedge \mathit{capacity}(\mathit{B\text{-}cells}, \mathit{insulin}) = \mathit{subnormal} \wedge \mathrm{QI} \leq 27 \wedge \mathsf{H}\,\mathrm{Condition}(\mathit{hyperglycaemia})) \rightarrow \mathsf{G}\,\mathrm{Condition}(\mathit{normoglycaemia})$

(6) $(\mathsf{G}\mathit{release}(\mathit{liver}, \mathit{glucose}) = \mathit{down} \wedge \mathit{capacity}(\mathit{B\text{-}cells}, \mathit{insulin}) = \mathit{subnormal} \wedge \mathrm{QI} > 27 \wedge \mathsf{H}\,\mathrm{Condition}(\mathit{hyperglycaemia})) \rightarrow \mathsf{G}\,\mathrm{Condition}(\mathit{normoglycaemia})$

(7) $((\mathsf{G}\mathit{release}(\mathit{liver}, \mathit{glucose}) = \mathit{down} \vee \mathsf{G}\mathit{uptake}(\mathit{peripheral\text{-}tissues}, \mathit{glucose}) = \mathit{up}) \wedge \mathit{capacity}(\mathit{B\text{-}cells}, \mathit{insulin}) = \mathit{nearly\text{-}exhausted} \wedge \mathsf{G}\mathit{secretion}(\mathit{B\text{-}cells}, \mathit{insulin}) = \mathit{up} \wedge \mathsf{H}\,\mathrm{Condition}(\mathit{hyperglycaemia})) \rightarrow \mathsf{G}\,\mathrm{Condition}(\mathit{normoglycaemia})$

(8) $(\mathsf{G}\mathit{uptake}(\mathit{liver}, \mathit{glucose}) = \mathit{up} \wedge \mathsf{G}\mathit{uptake}(\mathit{peripheral\text{-}tissues}, \mathit{glucose}) = \mathit{up}) \wedge$

$$\begin{aligned}&\mathit{capacity}(\mathit{B\text{-}cells}, \mathit{insulin}) = \mathit{exhausted} \wedge \\ &\mathsf{H}\,\mathrm{Condition}(\mathit{hyperglycaemia})) \\ &\rightarrow \mathsf{G}(\mathrm{Condition}(\mathit{normoglycaemia}) \vee \mathrm{Condition}(\mathit{hypoglycaemia}))\end{aligned}$$

(9) $(\mathrm{Condition}(\mathit{normoglycaemia}) \oplus \mathrm{Condition}(\mathit{hypoglycaemia}) \oplus \mathrm{Condition}(\mathit{hyperglycaemia})) \wedge \neg(\mathrm{Condition}(\mathit{normoglycaemia}) \wedge \mathrm{Condition}(\mathit{hypoglycaemia}) \wedge \mathrm{Condition}(\mathit{hyperglycaemia}))$

where $\oplus$ stands for the exclusive OR. Note that when the B-cells are exhausted, increased uptake of glucose by the tissues may result not only in normoglycaemia but also in hypoglycaemia (something not mentioned in the guideline).

4.2 Quality Check

As insulin can only be administered by injection, in contrast to the other drugs which are normally taken orally, doctors prefer to delay prescribing insulin as long as possible. Thus, the treatment part of the diabetes type 2 guideline mentions that one should start with prescribing oral antidiabetics (SU or BG, cf. Figure 1). Two of these can also be combined if taking only one has insufficient glucose-level lowering effect. If treatment is still unsatisfactory, the guideline suggests to: (1) either add insulin, or (2) stop with the oral antidiabetics entirely and to start with insulin.

The consequences of various treatment options were examined using the method introduced in Section 3. Hypothetical patients for whom it is the intention to reach a normal level of glucose in the blood (normoglycaemia) are considered, and treatment is selected according to the guideline fragments given in Figure 1:

- Consider a patient with hyperglycaemia due to nearly exhausted B-cells:

$$\begin{aligned}&\mathcal{B}_{\mathrm{DM2}} \cup \mathsf{G}\,T \cup \{\mathit{capacity}(\mathit{B\text{-}cells}, \mathit{insulin}) = \mathit{nearly\text{-}exhausted}\} \cup \\ &\quad \{\mathsf{H}\,\mathrm{Condition}(\mathit{hyperglycaemia})\} \models \mathsf{G}\,\mathrm{Condition}(\mathit{normoglycaemia})\end{aligned}$$

  holds for $T = \{\mathrm{Drug(SU)}, \mathrm{Drug(BG)}\}$, which also satisfies the minimality condition $O_{\subset}(T)$.

- Prescription of treatment $T = \{\mathrm{Drug(SU)}, \mathrm{Drug(BG)}, \mathrm{Drug(insulin)}\}$ for a patient with exhausted B-cells, as is suggested by the guideline, yields:

$$\begin{aligned}&\mathcal{B}_{\mathrm{DM2}} \cup \mathsf{G}\,T \cup \{\mathit{capacity}(\mathit{B\text{-}cells}, \mathit{insulin}) = \mathit{exhausted}\} \cup \\ &\quad \{\mathsf{H}\,\mathrm{Condition}(\mathit{hyperglycaemia})\} \models \\ &\qquad \mathsf{G}(\mathrm{Condition}(\mathit{normoglycaemia}) \vee \mathrm{Condition}(\mathit{hypoglycaemia}))\end{aligned}$$

In the last case, it appears that it is possible that a patient develops hypoglycaemia due to treatment; if this possibility is excluded from axiom (8) in the background knowledge, then the minimality condition $O_{\subset}(T)$, and also $O_{\subset,c}(T)$, do not hold since insulin by itself is enough to reach normoglycaemia. In either case, good practice medicine is violated, which is to prescribe as few drugs as possible, taking into account costs and side-effects of drugs. Here, three drugs are prescribed whereas only two should have been prescribed (BG and insulin, assuming that insulin alone is too costly), and the possible occurrence of hypoglycaemia should have been prevented.

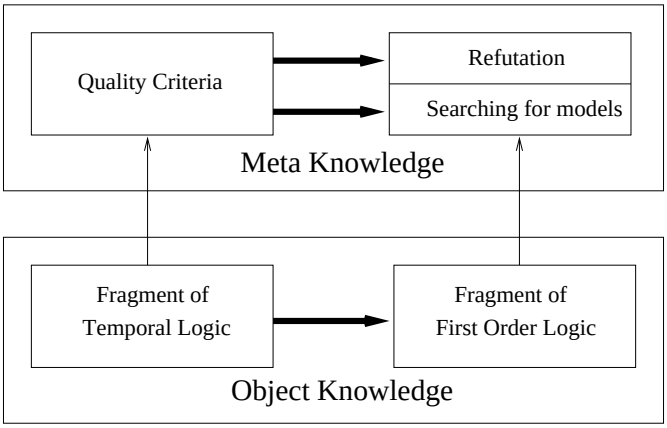

Figure 3: Translation of medical knowledge.

5 Automated Proving of Quality Requirements

As said in the introduction, we have explored the feasibility of using the automated reasoning tools OTTER and MACE-2 to check the quality of guidelines, in the sense as described above.

5.1 Motivation for the Theorem Proving Facilities

One of the most important application areas of model finders and theorem provers is program verification. Of course, with programs there is a clear beginning of the execution, which makes it intuitive to think about properties that occur after the start of the program. Therefore, it is not surprising much work has been done in the context of model finding and theorem proving with only the future time modality. However, it is more natural to model medical knowledge with past time operators, i.e., what happened to the patient in the past. It is well-known that formulas with a past-time modality can be mapped to a logical formula with only future time modalities such that both formulas are equivalent for some initial state [Gab89]. However, the main drawback to this approach is the fact that formulas will get much larger in size [Mar03] and as a consequence become much harder to verify in a theorem prover designed for modal logics.

For this reason, we have chosen to use an alternative approach which uses a *relational translation* to map the temporal logic formulas to first-order logic. As primary tools we use the resolution-based theorem prover OTTER and the finite model searcher MACE-2, which take first-order logic with equality as their input. There has been work done to improve the speed of resolution-based theorem provers on modal formulas [AGHdR00], but again, converse modalities such as the past-time operators are not considered. Nonetheless, we found that the general heuristics applicable to full first order logic are sufficient for our task

To clarify our approach, see Figure 3. We will first give a definition for translating the object knowledge to standard logic and then the translation of the meta-level knowledge will follow.

5.2 Translation

5.2.1 Translation of Object Knowledge

We assume that the formalisation is in propositional temporal logic. We do this by introducing a fresh proposition p for every equation that we find in the background knowledge. For functions with two elements in the co-domain, we have p and $\neg p$ and for the *capacity* function with three elements in its co-domain, we add a proposition p_x for each atom $\mathit{capacity}(\mathit{B\text{-}cells}, \mathit{insulin}) = x$ and the appropriate axiomatisation such that exactly one p_x holds. Technically this is not required, since we could extend the translation below to full first-order temporal logic. In practice however, we would like to avoid additional complexity from first-order formulas during the automated reasoning.

The relational translation (e.g., [Moo79, AGHdR00, SH03]) $\mathsf{ST}_t(\varphi)$, also referred to as standard translation, translates a propositional temporal logical formula φ into a formula in a first-order logic with a (time-indexed) unary predicate symbols P for every propositional variable p and one binary predicate $>$. It is defined as follows, where t is an individual variable:

$$
\begin{array}{lcl}
\mathsf{ST}_t(p) & \Leftrightarrow & P(t) \\
\mathsf{ST}_t(\neg\varphi) & \Leftrightarrow & \neg\mathsf{ST}_t(\varphi) \\
\mathsf{ST}_t(\varphi \wedge \psi) & \Leftrightarrow & \mathsf{ST}_t(\varphi) \wedge \mathsf{ST}_t(\psi) \\
\mathsf{ST}_t(\mathsf{G}\varphi) & \Leftrightarrow & \forall_{t'}\colon (t \not> t' \rightarrow \mathsf{ST}_{t'}(\varphi)) \\
\mathsf{ST}_t(\mathsf{H}\varphi) & \Leftrightarrow & \forall_{t'}\colon (t > t' \rightarrow \mathsf{ST}_{t'}(\varphi))
\end{array}
$$

Note that in our notation $\cup$ is sometimes used instead of a conjunction, so $\mathsf{ST}_t(\Gamma \cup \Delta)$ is defined as $\mathsf{ST}_t(\Gamma) \cup \mathsf{ST}_t(\Delta)$. Note that the last two elements of the definition define the meaning of the G modality and its converse, the H modality. For example, the formula $\mathsf{G}(p \rightarrow \mathsf{P}p)$ translates to $\forall_{t_2}(t \not> t_2 \rightarrow (P(t_2) \rightarrow \exists_{t_3}(t_2 > t_3 \wedge P(t_3))))$. It is easy to show that a formula in temporal logic is satisfiable if and only if its relational translation is.

In the literature a functional approach to translating modal logic has appeared as well [Ohl88], which relies on a non-standard interpretation of modal logic and could be taken as an alternative to this translation.

5.2.2 Translation of Meta-level Knowledge

Again, we consider the criteria for good practice medicine and make them suitable for the automated reasoning tools. We say that a treatment T is a treatment complying with requirements of good practice medicine iff:

(M1′) $\mathsf{ST}_t(\mathcal{B} \cup \mathsf{G}T \cup C \cup R) \nvdash \bot$

(M2′) $\mathsf{ST}_t(\mathcal{B} \cup \mathsf{G}T \cup C \cup R \cup \neg N) \vdash \bot$

(M3′) $\forall T' \subset T : T'$ is not a treatment according to (1) and (2)

It is easy to see that, because the relational translation preserves satisfiability, these quality requirements are equivalent to their unprimed counterparts. To automate this reasoning process we use MACE-2 to verify **(M1′)**, OTTER to verify **(M2′)**, and **(M3′)** can be seen as a combination of both for all subsets of the given treatment.

5.3 Proofs

In this subsection we will discuss the actual implementation in OTTER [McC03] and some results by using various heuristics.

5.3.1 Resolution Strategies

The main advantage that one gains from using a standard theorem prover is the fact that a whole range of different resolution rules are available. Note that Otter uses the set-of-support strategy [WRC65] as a standard strategy. With this strategy the original set of clauses is divided into a *set-of-support* and a *usable* set such that in every resolution step at least one of the parent clauses has to be member of the set-of-support and each resulting resolvent is added to the set-of-support.

Looking at the structure of the formulas in Section 4, one can see that formulas are of the type $p_0 \wedge \ldots \wedge p_n \rightarrow q$, where $p_0 \wedge \ldots \wedge p_n$ and q are all positive literals. Hence, we expect mostly negative literals in our clauses, which was exploited by using negative hyperresolution in OTTER. With this strategy a clause with at least one positive literal is resolved with a number of clauses which only contain negative literals (i.e., negative clauses), provided that the resolvent is a negative clause. The parent clause with at least one positive literal is called the *nucleus*, and the other, negative, clauses are referred to as the *satellites*. Positive hyperresolution, which uses positive satellites and a nucleus with at least one negative literal, was also tried. However, this did not result in successful proofs, because the background knowledge contains few positive clauses.

5.3.2 Verification

The ordering predicate $>$ that was introduced in Section 5.2.1 was defined by anti-reflexivity, anti-symmetry, and transitivity. We did not find any cases where the axiom of transitivity was required to construct the proof, which can be explained by the low modal depth of our formulas. As a consequence, the axiom was omitted with the aim to improve the speed of theorem proving.

We used OTTER to perform the two proofs which are instantiations of **(M2′)**. First we, again, consider a patient with hyperglycaemia due to nearly exhausted B-cells and prove:

$$\begin{aligned}&\mathsf{ST}_0(\mathcal{B}_{\mathrm{DM2}} \cup \mathsf{G}\,T \cup \{\mathit{capacity}(\mathit{B\text{-}cells}, \mathit{insulin}) = \mathit{nearly\text{-}exhausted}\} \cup \\ &\quad \{\mathsf{H}\,\mathrm{Condition}(\mathit{hyperglycaemia})\} \cup \{\neg\mathsf{G}\,\mathrm{Condition}(\mathit{normoglycaemia})\}) \vdash \bot\end{aligned}$$

where $T = \{\mathrm{Drug(SU)}, \mathrm{Drug(BG)}\}$,

This property was proven with OTTER in 62 resolution steps with the use of the negative hyperresolution strategy. As an example, we present a small snippet from the proof of this property. We will use the same syntax as we used in the previous sections, but each literal is augmented with a time-index. Note that $g(x, y) = \mathit{down}$ is implemented as a negative literal and functions f_1 and f_2 are Skolem functions introduced by OTTER. Both Skolem functions map a time point to a later time points. Consider the following clauses in the usable and set-of-support list. For example, assumption (53) models the capacity of the B-cells, i.e., nearly exhausted at time $t = 0$ where the property as shown above should be refuted:

2 $capacity(B\text{-}cells, insulin, t_0) \neq nearly\text{-}exhausted \vee$
$capacity(B\text{-}cells, insulin, t_0) \neq exhausted$

14 $t_0 \not> f_1(t_0) \vee capacity(B\text{-}cells, insulin, t_0) = exhausted \vee t_0 > t_1 \vee$
$secretion(B\text{-}cells, insulin, t_1) = up$

15 $\neg\text{Drug(SU)} \vee capacity(B\text{-}cells, insulin, t_0) = exhausted \vee t_0 > t_1 \vee$
$secretion(B\text{-}cells, insulin, t_1) = up\}$

51 $0 > t_0 \vee \text{Drug}(\text{SU}, t_0)$

53 $capacity(B\text{-}cells, insulin, 0) = nearly\text{-}exhausted$

Very early in the proof, OTTER deduces that if the capacity of insulin in B-cells is nearly-exhausted, then it is not completely exhausted:

56 [neg_hyper,53,2] $capacity(B\text{-}cells, insulin, 0) \neq exhausted$

Now we skip a part of the proof, which results in information about the relation between the capacity of insulin and the secretion of insulin in B-cells for a certain time point:

517 [neg_hyper,516,53] $0 \not> f_2(0)$

765 [neg_hyper,761,50,675] $capacity(B\text{-}cells, insulin, f_2(0)) \neq nearly\text{-}exhausted \vee$
$secretion(B\text{-}cells, insulin, f_2(0)) = down$

This information allows OTTER to quickly complete the proof, by combining it with the information about the effects of a sulfonylurea drug:

766 [neg_hyper,765,15,56,517] $capacity(B\text{-}cells, insulin, f_1(0)) \neq nearly\text{-}exhausted \vee$
$\neg\text{Drug(SU)}$

767 [neg_hyper,765,14,56,517] $capacity(B\text{-}cells, insulin, f_1(0)) \neq nearly\text{-}exhausted \vee$
$0 \not> f_1(0)$

after which (53) can be used as a nucleus to yield:

768 [neg_hyper,767,53] $0 \not> f_1(0)$

and consequently by taking (51) as a nucleus, we find that at time point 0 the capacity of insulin is not nearly exhausted:

769 [neg_hyper,768,51,766] $capacity(B\text{-}cells, insulin, 0) \neq nearly\text{-}exhausted$

This directly contradicts one of the assumptions and this results in an empty clause:

770 [binary,769.1,53.1] $\bot$

Similarly, we could prove that given a treatment $T = \{\text{Drug(SU)}, \text{Drug(BG)}, \text{Drug(insulin)}\}$ for a patient with exhausted B-cells, as is suggested by the guideline, it follows that:

$$\mathsf{ST}_0(\mathcal{B}_{\text{DM2}} \cup \mathsf{G}\,T \cup \{capacity(B\text{-}cells, insulin) = exhausted\} \cup$$
$$\{\mathsf{HCondition}(hyperglycaemia)\} \cup$$
$$\{\neg(\mathsf{G}(\text{Condition}(normoglycaemia) \vee \text{Condition}(hypoglycaemia)))\}) \vdash \bot$$

The proof of OTTER is omitted, but a similar magnitude of complexity in the proof can be observed, i.e., 52 resolution steps.

Weights	Clauses (binary res)	Clauses (negative hyper res)
$(0, 1)$	17729	6994
$(1, 0)$	13255	6805
$(1, 1)$	39444	7001
$(1, -1)$	13907	6836
$(2, -2)$	40548	7001
$(2, -3)$	16606	6805
$(3, -4)$	40356	7095
$(3, -5)$	27478	7001

Figure 4: Generated clauses to prove an instance of property **M2′** depending on weights (x, y) for the ordering relation on time.

5.3.3 Weighing the Clauses

In this section we consider the weighing facilities as offered by OTTER to improve the performance. Consider the example from [AGHdR00]. Suppose we have the formula $\mathsf{G}(p \rightarrow \mathsf{F}p)$. Proving this satisfiable amounts to proving that the following two clauses are satisfiable:

1. $c > t_1 \vee \neg P(t_1) \vee t_1 \not> f(t_1)$
2. $c > t_2 \vee \neg P(t_2) \vee P(f(t_2))$

The observation can be made, that although we have two possibilities to resolve these two clauses, for example on the P literal, this is useless because the negative P literal is only bound by the G-operator while the positive P literal comes from a formula at a deeper modal depth under the F-operator. Suppose we resolve these two P literals, which generates the clause:

$$c > f(t) \vee f(t) \not> f(f(t)) \vee c > t \vee \neg P(t)$$

and with (2) again we have:

$$c > f(f(t)) \vee f(f(t)) \not> f(f(f(t))) \vee c > f(t) \vee c > t \vee \neg P(t)$$

etc.

So we can see that we can generate a lot of new clauses, but clearly these nestings of the Skolem functions will not lead to a contradiction very quickly if the depth of the modalities in the formulas that we have is small.

In OTTER the weight of the clauses determines which clauses are chosen from the set-of-support and usable list to become parents in a resolution step. Clearly, because the goal of resolution is to find an empty clause, lighter clauses are preferred. By default, the weight of a clause is the sum of all occuring symbols (i.e., all symbols have weight 1), but as we have argued, the nesting of Skolem functions will not help to find such an empty clauses. Therefore it can be of use to manually change the weight of the ordering symbol, which is done in OTTER by a tuple (x, y) for each predicate, where x is multiplied with sum of the weight of its arguments and is added to y to calculate the new weight

```
> :                    Condition(hyperglycaemia) :
    | 0 1                          0 1
  --+----                        -------
  0 | F T                          T T
  1 | F F
```

Figure 5: Snippet from a MACE-2 generated model

of this predicate. For example, if $x = 2$ and $y = -3$, then $x > y$ has a total weight of $2+2-3 = 1$, while $f(f(x)) > f(y)$ has a weight of $2*3+2*2-3 = 7$. See Figure 4 where we show results when we applied this for some small values for x and y for both binary and negative hyperresolution. What these numbers tend to show (similar numbers were gained from the other property) is that the total weight of the ordering predicate should be smaller than the average weight of other, unary, predicates. Nonetheless, possibly somewhat suprisingly, the factor x should not be raised too much, although in the case of a negative hyperresolution strategy the effect is minimal. Furthermore, we can see that combining the resolution strategies with a weighing strategy does help, but the advantages are rather limited compared to the advantages of weighing in combination with binary resolution.

5.4 Disproofs

MACE-2 (Models And CounterExamples) [McC01] is a program that searches for small finite models of first-order statements using a David-Putman-Loveland-Logemann decision procedure [DP69, DLL62] as its core. Because of the relative simplicity of our temporal formulas, it is to be expected that counterexamples can be found with very few states. Hence, it can be expected that models are in the same magnitude as the propositional case and this is indeed the case. In fact, the countermodels that MACE-2 found consist of only 2 elements in the domain of the model.

The first property we check corresponds to checking if the background knowledge augmented with a patient and a therapy is consistent, i.e., criterium **(M1′)**. So, again consider a patient with hyperglycaemia due to nearly exhausted B-cells. We have used MACE-2 to verify:

$$\mathsf{ST}_0(\mathcal{B}_{\mathrm{DM2}} \cup \mathsf{G}\ T \cup \{\mathit{capacity}(\mathit{B\text{-}cells}, \mathit{insulin}) = \mathit{exhausted}\} \cup \{\mathsf{H}\,\mathrm{Condition}(\mathit{hyperglycaemia})\}) \nvdash \bot$$

for $T = \{\mathrm{Drug(SU)}, \mathrm{Drug(BG)}, \mathrm{Drug(insulin)}\}$. From this, of course, it follows that there is a model if $T = \{\mathrm{Drug(SU)}, \mathrm{Drug(BG)}\}$ and consequently we have verified **(M1′)**.

Similarly, we find that for all $T \subset \{\mathrm{Drug(SU)}, \mathrm{Drug(BG)}\}$, it holds that:

$$\mathsf{ST}_0(\mathcal{B}_{\mathrm{DM2}} \cup \mathsf{G}\,T \cup \{\mathit{capacity}(\mathit{B\text{-}cells}, \mathit{insulin}) = \mathit{nearly\text{-}exhausted}\} \cup \{\mathsf{H}\,\mathrm{Condition}(\mathit{hyperglycaemia})\} \cup \{\neg\mathsf{G}\,\mathrm{Condition}(\mathit{normoglycaemia})\}) \nvdash \bot$$

So indeed the conclusion is that the treatment complies with **(M3′)** and thus complies with the criteria of good practice medicine. See for example Figure 5, which

contains a small sample of the output that MACE-2 generated. The output is a first-order model with two elements in the domain named '0' and '1' with an interpretation of all predicates and functions to this domain. It shows that it is consistent with the background knowledge to believe that the patient will continue to suffer from hyperglycaemia if one of the drugs is not applied. It is interesting to see that there is also a smaller model where the size of the domain is 1 for this set of formulas.

Finally, consider the treatment $T = \{\text{Drug(SU)}, \text{Drug(BG)}, \text{Drug(insulin)}\}$ for a patient with exhausted B-cells, and suppose we exclude the patient developing hypoglycaemia, we can show that:

$$\begin{aligned}&\mathsf{ST}_0(\mathcal{B}_{\text{DM2}} \cup \mathsf{G}\,T \cup \{\mathit{capacity}(\mathit{B\text{-}cells}, \mathit{insulin}) = \mathit{exhausted}\} \cup \\ &\quad \{\mathsf{H}\text{Condition}(\mathit{hyperglycaemia})\} \cup \\ &\quad \{\neg\mathsf{G}(\text{Condition}(\mathit{hyperglycaemia})\} \cup \{\neg\mathsf{G}(\text{Condition}(\mathit{hypoglycaemia}))\}) \nvdash \bot\end{aligned}$$

But, it is possible to prove the same property if $T = \{\text{Drug(insulin)}\}$ and thus **(M3′)** does not hold in this case and as a consequence the guideline does not comply with the quality requirements as discussed in the previous section.

6 Discussion

The quality of guideline design is for the largest part based on its compliance with specific treatment aims and global requirements. To this purpose, use was made of the theory of abductive, diagnostic reasoning, i.e., we diagnosed potential problems with a guideline using logical abduction [Luc97, Luc03, Poo90]. This is a meta-level characterisation of the quality of a medical guideline. What was diagnosed were problems in the relationship between medical knowledge, suggested treatment actions in the guideline text and treatment effects; this is different from traditional abductive diagnosis, where observed findings are explained in terms of diagnostic hypotheses. This method allows us to examine fragments of a guideline and to prove properties of those fragments.

In earlier work [HLB04], where we used a tool for interactive program verification named KIV [Rei95], we performed a similar exercise. The main advantage of using interactive theorem proving is that the resulting proof is relatively elegant compared to automated resolution-based solutions. This might be important if one wants to convince the medical community that a guideline complies with their medical quality requirements and to promote the implementation of such a guideline. However, to support the *design* of guidelines, this argument is of less importance. Moreover, the work that needs to be done to construct a proof in an interactive theorem prover would severely slow down the development process as people with specialised knowledge are required.

Even though guideline developers might not be interested in inspecting the full proof or disproof of a certain property, it is of importance for the process that if a certain proof fails, they have a method to find out *why* the proof failed. Thus in our future work we will focus on the question whether it is possible to give hints to guideline developers on how to improve their guidelines. Furthermore, our quality requirements are far from exhaustive and the last few years research has been done in this field (e.g. [FAB+04]). Our aim will be to extend our current work with these new insights.

In this paper, we have made use of tools designed for automated reasoning to actually quality check a medical guideline using the theory of quality of guidelines developed previously [Luc03]. This complements both the earlier work on object-level verification of medical guidelines using the interactive theorem prover designed for program verification KIV [MBtTvH02], but also our earlier work where we used KIV for meta-level reasoning [HLB04].

References

[AGHdR00] C. Areces, R. Gennari, J. Heguiabehere, and M. de Rijke. Tree-based Heuristics in Modal Theorem Proving. In *Proceedings of the ECAI'2000*, Berlin, Germany, 2000.

[DLL62] M. Davis, G. Logemann, and D. Loveland. A Machine Program for Theorem Proving. *Communications of the ACM*, 5(7):394–397, 1962.

[DP69] M. Davis and H. Putman. A Computing Procedure for Quantification Theory. *Journal of the ACM*, 7:201–215, 1969.

[FAB+04] J. Fox, A. Alabassi, E. Black, C. Hurt, and T. Rose. Modelling Clinical Guidelines: a Corpus of Examples and a Tentative Ontology. In K. Kaiser, S. Miksch, and S.W. Yu, editors, *Computer-based Support for Clinical Guidelines and Protocols. Proceedings of the Symposium on Computerized Guidelines and Protocols (CGP 2004)*, volume 101 of *Studies in Health Technology and Informatics*, pages 31–45, Amsterdam, 2004. IOS Press.

[FD00] J. Fox and S. Das. *Safe and Sound: Artificial Intelligence in Hazardous Applications*. MIT Press, 2000.

[Gab89] D.M. Gabbay. The Declarative Past and Imperative Future: Executable Temporal Logic for Interactive Systems. In H. Barringer, editor, *Temporal Logic in Specification*, volume 398 of *LNCS*, pages 409–448. Springer-Verlag, Berlin, 1989.

[HLB04] A.J. Hommersom, P.J.F. Lucas, and M. Balser. Meta-level Verification of the Quality of Medical Guidelines using Interactive Theorem Proving. In J. J. Alferes and J. Leite, editors, *JELIA'04*, volume 3225 of *LNAI*, pages 654–666, Heidelberg, 2004. Springer-Verlag.

[Luc93] P.J.F. Lucas. The Representation of Medical Reasoning Models in Resolution-based Theorem Provers. *Artificial Intelligence in Medicine*, 5:395–419, 1993.

[Luc95] P.J.F. Lucas. Logic Engineering in Medicine. *The Knowledge Engineering Review*, 10(2):153–179, 1995.

[Luc97] P.J.F. Lucas. Symbolic Diagnosis and its Formalisation. *The Knowledge Engineering Review*, 12(2):109–146, 1997.

[Luc03] P.J.F. Lucas. Quality Checking of Medical Guidelines through Logical Abduction. In F. Coenen, A. Preece, and A.L. Mackintosh, editors, *Proceedings of AI-2003 (Research and Developments in Intelligent Systems XX)*, pages 309–321, London, 2003. Springer.

[Mar03] N. Markey. Temporal Logic with Past is Exponentially More Succinct. *EATCS Bulletin*, 79:122–128, 2003.

[MBtTvH02] M. Marcos, M. Balser, A. ten Teije, and F. van Harmelen. From Informal Knowledge to Formal Logic: a Realistic Case Study in Medical Protocols. In *Proceedings of the 12th EKAW-2002*, 2002.

[McC01] W. McCune. MACE 2.0 Reference Manual and Guide. Tech. Memo ANL/MCS-TM-249, Argonne National Laboratory, Argonne, IL, June 2001.

[McC03] W. McCune. Otter 3.3 Reference Manual. Tech. Memo ANL/MCS-TM-263, Argonne National Laboratory, Argonne, IL, August 2003.

[Moo79] R.C. Moore. *Reasoning about Knowledge and Action.* PhD thesis, MIT, 1979.

[Ohl88] H.J. Ohlbach. A Resolution Calculus for Modal Logics. In E. Lusk and R. Overbeek, editors, *Proceedings CADE-88: International Conference on Auomated Deduction*, volume 310 of *LNCS*, pages 500–516. Springer-Verlag, 1988.

[OMGM98] L. Ohno-Machado, J.H. Gennari, and S.N. Murphy. Guideline Interchange Format: a Model for Representing Guidelines. *Journal of the American Medical Informatics Association*, 5(4):357–372, 1998.

[Poo90] D. Poole. A Methodology for using a Default and Abductive Reasoning System. *International Journal of Intelligent System*, 5(5):521–548, 1990.

[Rei95] W. Reif. The KIV Approach to Sftware Verification. In M. Broy and S. Jähnichen, editors, *KORSO: Methods, Languages, and Tools for the Construction of Correct Software*, volume 1009 of *LNCS*. Springer-Verlag, Berlin, 1995.

[Rob65] J.A. Robinson. Automated Deduction with Hyperresolution. *International Journal of Computatational Mathematics*, 1:23–41, 1965.

[SH03] R.A. Schmidt and U. Hustadt. Mechanised Reasoning and Model Generation for Extended Modal Logics. In H.C.M. de Swart, E. Orlowska, G. Schmidt, and M. Roubens, editors, *Theory and Applications of Relational Structures as Knowledge Instrument*, volume 2929 of *LNCS*, pages 38–67. Springer, 2003.

[SMJ98] Y. Shahar, S. Miksch, and P. Johnson. The Asgaard Project: a task-specific framework for the application and critiquing of time-oriented clinical guidelines. *Artificial Intelligence in Medicine*, 14:29–51, 1998.

[Tur85] R. Turner. *Logics for Artificial Intelligence*. Ellis Horwood, Chichester, 1985.

[WOLB84] L. Wos, R. Overbeek, E. Lusk, and J. Boyle. *Automated Reasoning: Introduction and Applications*. Prentice-Hall, Englewood Cliffs, NJ, 1984.

[Woo00] S.H. Woolf. Evidence-based Medicine and Practice Guidelines: an overview. *Cancer Control*, 7(4):362–367, 2000.

[WRC65] L. Wos, G. Robinson, and D. Carson. Efficiency and Completeness of the Set of Support Strategy in Theorem Proving. *ACM Journal*, 12:536–541, October 1965.