Meta-level Verification of the Quality of Medical Guidelines using Interactive Theorem Proving*

Arjen Hommersom[1], Peter Lucas[1] and Michael Balser[2]

[1]Institute for Computing and Information Sciences, University of Nijmegen,
{arjenh,peterl}@cs.kun.nl
[2]Institut für Informatik, Universität Augsburg,
balser@informatik.uni-augsburg.de

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. In this paper, we investigate how this approach can be mapped to the facilities offered by a theorem proving system for program verification, KIV. It is shown that the reasoning that is required for checking the quality of a guideline can be mapped to such theorem-proving facilities. The medical quality of an actual guideline concerning diabetes mellitus 2 is investigated in this way, and some problems discovered are discussed.

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* [12]. 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, and some of them, for example in the Asgaard project [10], 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 good real-world examples of highly structured, systematic documents that are amenable to formalisation.

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

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 [10], and next applying techniques from program verification to the resulting representation in establishing partial or total correctness; (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. Carrying out such checks could be valuable, in particular during the process of *designing* medical guidelines.

In this paper we explore the route from an informal medical guideline to its logical formalisation and 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 [7]. In this paper we study the use of logical deduction using temporal logic to formally establish whether a guideline fulfils particular quality requirements. For this purpose use was made of the theorem prover KIV [1]. This is a somewhat unusual approach, as KIV and its underlying logics are especially targeted at the verification of parallel programs, whereas here we are concerned with a type of reasoning that comes from AI.

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, including the logic supported by the theorem prover KIV, are briefly reviewed. In Section 3 the formalisation of guideline quality using a meta-level schema 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 to in Section 4 and a formalisation of this is given. The approach to checking the quality of this guideline using the deductive machinery offered by KIV is presented in Section 5. Finally, Section 6 discusses what has been achieved and suggests some future plans for research.

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, and 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 Fig. 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.

- Step 1: diet
- Step 2: if Quetelet Index (QI) $\leq$ 24, 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

Fig. 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 $k+1$.

One way to use formal methods in the context of guidelines is to automatically verify whether a medical guideline fulfils particular properties, such as whether it complies with quality *indicators* as proposed by health-care professionals [8]. 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 fulfils 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 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 Fig. 1, can be described precisely by means of temporal logic [8]. 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 [7]. 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:

$\models \mathsf{P}\varphi \leftrightarrow \neg\mathsf{H}\neg\varphi$ (somewhere in the past)

$\models \mathsf{F}\varphi \leftrightarrow \neg\mathsf{G}\neg\varphi$ (somewhere 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. [11].

Even though this logic was shown to be suitable for representation purposes, we had to map it to the temporal logic underlying KIV, which we had chosen as the system to be used for formal verification. As a consequence, in the next section, this temporal logic is briefly described. The mapping is given in Section 5.1.

2.3 Temporal Logic in KIV

The interactive theorem prover KIV offers support for future-time linear temporal logic [2]. Reactive systems can be described in KIV by means of state-charts or parallel programs; here we use parallel programs. A state of a system can be described by first-order logic. Furthermore, static variables v, which have the same values at each time point, are distinguished from dynamic variables V. A specialty of KIV is the use of primed and double-primed variables: a primed variable V' represents the value of this variable after a system transition, the double-primed variable V'' is interpreted as the value after an environment transition. System and environment transitions alternate, with V'' being equal to V in the successive state.

The supported future-time temporal operators include the operators from Table 2, where $\mathrm{succ}(t)$ is the set of zero or one successors of t. Note that all formulae are interpreted with respect to the first point of time. Let e denote an arbitrary (first-order) expression, then constructs for parallel programs include: $V := e$ (*assignments*), **if** ψ **then** ϕ_1 **else** ϕ_2 (*conditionals*), **while** ψ **do** ϕ (*loops*), **var** $V = e$ **in** ϕ (*local variables*), **patom** ϕ **end** (*atomic execution*), $\phi_1 \parallel \phi_2$ (*interleaved execution*), and $p(e, V)$ (*call to procedure p with value parameters e and var parameters V*).

A temporal logic property for a parallel program is verified in KIV by symbolic execution with induction. Hence, there is a major difference between the temporal logic underlying KIV and the one discussed in the previous section, both in intention and in expressive power.

3 Application to Medical Knowledge

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

Table 2. Used temporal operators; t stands for a time instance.

Notation	Interpretation	Formal semantics
$\Box\ \varphi$	φ will always be true	$t \vDash \Box\ \varphi \Leftrightarrow \forall t' \geq t : t' \vDash \varphi$
$\Diamond\ \varphi$	φ will eventually be true	$t \vDash \Diamond\ \varphi \Leftrightarrow \exists t' \geq t : t' \vDash \varphi$
φ **until** ψ	φ holds until ψ eventually holds	$t \vDash \varphi$ **until** ψ $\Leftrightarrow \exists\ t' \geq t :\quad t' \vDash \psi$ $\wedge \forall\ t \leq t'' < t' : t'' \vDash \varphi$
φ **unless** ψ	φ holds unless ψ holds	$t \vDash \varphi$ **unless** ψ $\Leftrightarrow \forall\ t' \geq t :\quad t' \vDash \varphi$ $\vee \exists\ t \leq t'' \leq t' : t'' \vDash \psi$
$\circ\ \varphi$	execution does not terminate and the next state satisfies φ	$t \vDash \circ\ \varphi \Leftrightarrow \exists\ t' \in \mathrm{succ}(t) : t' \vDash \varphi$
$\bullet\ \varphi$	either execution terminates or the next state satisfies φ	$t \vDash \circ\ \varphi \Leftrightarrow \forall\ t' \in \mathrm{succ}(t) : t' \vDash \varphi$
last	the current state is the last	$t \vDash$ **last** $\Leftrightarrow \mathrm{succ}(t) = \emptyset$

- 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. [5] 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, 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:

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

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 [9, 6]:

(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 $c(T) = \sum_{d \in T} cost(d)$; 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.

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 Fig. 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. Hence, the conclusion is that a deeper biological analysis is required, the results of which are presented below.

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) $(\mathsf{G}\mathit{uptake}(\mathit{liver}, \mathit{glucose}) = \mathit{up} \wedge$

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

$$\begin{aligned}
(9)\ (&\mathrm{Condition}(\mathit{normoglycaemia}) \oplus \mathrm{Condition}(\mathit{hypoglycaemia}) \\
&\oplus \mathrm{Condition}(\mathit{hyperglycaemia}))
\end{aligned}$$

where $\oplus$ stands for the exclusive OR. Note that when the B-cells are exhausted, increased uptake of glucose by the tissues may not only result 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. Fig. 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 Fig. 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, 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.

5 Quality Checking using Symbolic Execution with Induction

In the previous section we have seen that temporal logic can be used to formally check a medical guideline, but so far only from a theoretical point of view. Here we will study how such proofs can be constructed semi-automatically in terms of symbolic execution with induction using the theorem prover KIV.

5.1 Translation to KIV

In this paper we will only discuss the translation of the constructs that were employed in the formalisation in the previous section. Firstly, the universal quantification of the axioms over all points in time is made explicit. Secondly, the modal operators have to be translated. The only modal operators that were used were G and H. The operator G is semantically equivalent to KIV's $\Box$ operator. However, KIV does not support past-time operators, but as Gabbay et al. have shown [4], it is possible to translate any temporal formula with past-time operators to an equivalent temporal formula with only future-time operators that includes 'until'. This implies that after translation it is possible, at least in principle, to verify the temporal formulas introduced in sections 3 and 4. Axioms hold over all points in time of which the the ones with past-time formulas are of the following fixed form (see section 3):

$$(\varphi \wedge \mathsf{H}\,\mathrm{Condition}(\mathit{hyperglycaemia})) \rightarrow \psi$$

We can rewrite this semantically and obtain a pure future-time formula, i.e. a formula with only future-time operators, as follows:

$$\begin{aligned}
&\forall t : t \vDash (\varphi \wedge \mathsf{H}\,\mathrm{Condition}(\mathit{hyperglycaemia})) \rightarrow \psi \\
\Leftrightarrow\ &\forall t : t \vDash (\varphi \rightarrow \psi) \vee \neg\, \mathsf{H}\,\mathrm{Condition}(\mathit{hyperglycaemia})) \\
\Leftrightarrow\ &\forall t : t \vDash \varphi \rightarrow \psi \text{ or } t \nvDash \mathsf{H}\,\mathrm{Condition}(\mathit{hyperglycaemia})) \\
\Leftrightarrow\ &\forall t : t \vDash \varphi \rightarrow \psi \text{ or } \neg \forall t' < t : t' \vDash \mathrm{Condition}(\mathit{hyperglycaemia}) \\
\Leftrightarrow\ &\neg \exists t : t \vDash \neg(\varphi \rightarrow \psi) \text{ and } \forall t' < t : t' \vDash \mathrm{Condition}(\mathit{hyperglycaemia}) \\
\Leftrightarrow\ &\neg\,(\mathrm{Condition}(\mathit{hyperglycaemia}) \text{ until } \neg\,(\phi \rightarrow \psi))
\end{aligned}$$

5.2 Specification in KIV

In KIV datatypes are expressed in a many-sorted algebra with possibilities for parameterisation, allowing the creation of specific sorts by defining constraints on the parameters. The sorts with associated data elements required to create a specification of the domain of diabetes mellitus type 2 are listed in Table 3.

In KIV, functions and predicates are static, i.e. they do not change over time. Therefore, for the formalisation in KIV functions and predicates were mapped to dynamic variables. For example, *secretion*(*B-cells*, *insulin*) was mapped to a dynamic variable named `BsecretionI`. Since variables in axioms of algebraic specifications are universally quantified, a procedure with name '`patient`' was used to bind these variables. This gives each relevant variable a context and prohibits instantiations of axioms with variables that have different names.

Table 3. Data specifications.

Specification	Data elements
capacity	exhausted, nearly-exhausted, subnormal
condition	hyperglycaemia, hypoglycaemia, normoglycaemia
updown	up, down
drug	SU, BG, glucosidase, insulin
setdrugs	set of elements of sort drug
setsetdrugs	set of elements of sort setdrugs

The axioms (3), (4) and (7) were selected and translated to KIV's syntax as indicated in Section 5.1. In addition, a number of variables were primed to deal with the consistency condition mentioned in Section 3, as will be discussed in Section 5.4. This yielded the following three sequents, denoted by $\mathcal{A}$:

$$\texttt{[patient(; Drugs, Condition, UptakeLG, UptakePG, ReleaseLG BcapacityI, BsecretionI, QI)]} \vdash$$
$$\Box\ (((\Box\ \texttt{SU} \in \texttt{Drugs}) \wedge \texttt{BcapacityI} \neq \texttt{exhausted}) \rightarrow \Box\ \texttt{BsecretionI}' = \texttt{up});$$

$$\texttt{[patient(; Drugs, Condition, UptakeLG, UptakePG, ReleaseLG BcapacityI, BsecretionI, QI)]} \vdash$$
$$\Box\ ((\Box\ \texttt{BG} \in \texttt{Drugs}) \rightarrow (\Box\ \texttt{ReleaseLG}' = \texttt{down}));$$

$$\texttt{[patient(; Drugs, Condition, UptakeLG, UptakePG, ReleaseLG BcapacityI, BsecretionI, QI)]} \vdash$$
$$\neg(\texttt{Condition} = \texttt{hyperglycaemia until}$$
$$\neg((((\Box\ \texttt{ReleaseLG}' = \texttt{down}) \vee (\Box\ \texttt{UptakePG} = \texttt{up}))$$
$$\wedge(\texttt{BcapacityI} = \texttt{nearly-exhausted}) \wedge \Box\ \texttt{BsecretionI}' = \texttt{up})$$
$$\rightarrow (\Box\ \texttt{Condition}' = \texttt{normoglycaemia})));$$

Now define $\mathcal{B}'_{\mathrm{DM2}}$ as the conjunction of the right-hand-sides of $\mathcal{A}$. We will show how the meta-level properties follow from these right-hand-sides. The procedure `patient` only acts as a placeholder.

5.3 Proof

Again, consider a patient with hyperglycaemia due to nearly exhausted B-cells and $T = \{\mathrm{Drug(SU)}, \mathrm{Drug(BG)}\}$. The following sequent, which corresponds to condition M2 from section 3, was proven by KIV in about 50 steps:

$$\texttt{[patient(; Drugs, Condition, UptakeLG, UptakePG, ReleaseLG BcapacityI, BsecretionI, QI)]}$$
$$\vdash \neg(\texttt{Condition} = \texttt{hyperglycaemia until}$$
$$\neg(((\Box\ \texttt{Drug} = \{\texttt{SU}, \texttt{BG}\}) \wedge \texttt{BcapacityI} = \texttt{nearly-exhausted})$$
$$\rightarrow (\Box\ \texttt{Condition}' = \texttt{normoglycaemia})));$$

The proof relies on the fact that the axioms can be inserted with the appropriate (program-)variables, after which the patient procedure can be removed from the sequent and the real work starts. Hence, the consequent of the sequent is deduced from the axioms $\mathcal{B}'_{\mathrm{DM2}}$. This yields:

$$\mathcal{B}'_{\mathrm{DM2}} \vdash \neg(\texttt{Condition} = \texttt{hyperglycaemia until}\ \neg(((\Box\ \texttt{Drug} = \{\texttt{SU}, \texttt{BG}\}) \wedge \texttt{BcapacityI} = \texttt{nearly-exhausted}) \rightarrow (\Box\ \texttt{Condition}' = \texttt{normoglycaemia})));$$

An outline of this proof follows. The proof obligation $\Gamma \vdash \Delta,\ \neg(\varphi\ \mathbf{until}\ \psi)$ is equivalent to $\Gamma,\ \varphi\ \mathbf{until}\ \psi \vdash \Delta$. The sequent is proved by induction over the number of steps it takes to satisfy ψ. For this, introduce a fresh dynamic variable N and generalise the sequent to $(N = N'' + 1 \wedge \phi)\ \mathbf{until}\ \psi,\ \Gamma \vdash \Delta$. The equation $N = N'' + 1$ ensures that N decreases in each step. Now, we can perform induction with induction term N which yields

$$(N = N'' + 1 \wedge \phi)\ \mathbf{until}\ \psi,\ \Gamma,\ N = n,\ \Box(N < n \rightarrow \mathsf{IndHyp}) \vdash \Delta$$

where $\mathsf{IndHyp} = ((N = N'' + 1 \wedge \phi)\ \mathbf{until}\ \psi) \wedge \bigwedge \Gamma \rightarrow \bigvee \Delta$ and n is a new static variable. We move to the next state by symbolically executing the temporal formulae. For example,

$$\phi\ \mathbf{until}\ \psi \Leftrightarrow \psi \vee (\phi \wedge \circ(\phi\ \mathbf{until}\ \psi))$$

is used to execute the **until** operator. In this case, the induction hypothesis can be applied in all possible successive states.

5.4 Disproofs

The final part of this section we will show disproofs of properties that do not follow from $\mathcal{B}'_{\mathrm{DM2}}$ by using program verification techniques. In the previous section we reasoned with the given axioms $\mathcal{A}$, but here we use a more extensive implementation of the `patient` procedure as shown in Fig. 5.4, which not only binds variables, but implements part of the therapeutic reasoning.

Now define the theory

$$M = \{\texttt{[patient(...)]}\} \cup \bigcup_{x \neq \texttt{Drugs}} \{\Box\ x' = x''\}$$

where the last term denotes that variables, except for `Drugs`, are not altered by the environment, but only by the program itself. In about 400 steps using KIV it was proved that $M \vdash \mathcal{B}'_{\mathrm{DM2}}$, which implies $M \vDash \mathcal{B}'_{\mathrm{DM2}}$ assuming KIV is sound. From this and the fact that M is consistent (since a program is consistent and the environment is not altered), we have shown that $\mathcal{B}'_{\mathrm{DM2}} \nvDash \bot$ and therefore condition M1. The number of steps shows that this proof was significantly harder. The reason is that in many cases an invariant could only be defined after an initial symbolic execution. This caused an explosion of states that had to be considered. Furthermore, the invariants that had to be formulated were less straightforward.

Now showing that this set of drugs is a minimal treatment (condition M3), as discussed in Section 4, we construct for all $T' \in \wp\{\mathrm{SU}, \mathrm{BG}\},\ T' \neq \{\mathrm{SU}, \mathrm{BG}\}$:

$$M_{T'} = M \cup \{\Box\ \texttt{Drugs}' = \texttt{Drugs}'', \texttt{Condition} = \texttt{hyperglycaemia}, \texttt{BsecretionI} = \texttt{down}, \texttt{BcapacityI} = \texttt{nearly-exhausted}, \texttt{ReleaseLG} = \texttt{up}, \texttt{UptakePG} = \texttt{down}, \texttt{Drugs} = T'\}$$

```
patient(var Drugs, Condition, UptakeLG, UptakePG,
            ReleaseLG, BcapacityI, BsecretionI, QI)

begin
  var oncebcapi = false, hchyper = true, nownormal = false in
    while true do
      patom
        if SU ∈ Drugs ∧ (BcapacityI ≠ exhausted ∨ oncebcapi) then
        begin
          BsecretionI := up;
          oncebcapi   := true
        end;
        if BG ∈ Drugs then ReleaseLG := down;
        if  (ReleaseLG = down ∨ UptakePG = up) ∧ BsecretionI = up ∧
            ((Bcapacity = nearly-exhausted ∧ hchyper) ∨ nownormal) then
        begin
          nownormal := true;
          hchyper   := false;
          Condition := normoglycaemia
        end
      end
end
```

Fig. 2. Declaration of the patient procedure.

Again, $M_{T'}$ is consistent. It was proved in about 25 steps with KIV that:

$$M_{T'} \vdash (\texttt{Condition} = \texttt{hyperglycaemia until} \\ \neg(((\Box\ \texttt{Drugs} = T') \wedge \texttt{BcapacityI} = \texttt{nearly-exhausted}) \\ \rightarrow (\Box\ \texttt{Condition}' = \texttt{normoglycaemia})));$$

Because of monotony of temporal logic and $M \vDash \mathcal{B}'_{\mathrm{DM2}}$, we have $M_{T'} \vDash \mathcal{B}'_{\mathrm{DM2}}$. Since $M_{T'}$ is consistent, we can conclude:

$$\mathcal{B}'_{\mathrm{DM2}} \nvdash \neg(\texttt{Condition} = \texttt{hyperglycaemia until} \\ \neg(((\Box\ \texttt{Drugs} = T') \wedge \texttt{BcapacityI} = \texttt{nearly-exhausted}) \\ \rightarrow (\Box\ \texttt{Condition}' = \texttt{normoglycaemia})));$$

Hence, $T = \{\mathrm{Drug(SU)}, \mathrm{Drug(BG)}\}$ is a minimal treatment. As one might expect, it shows that after the construction of the appropriate countermodel, disproofs are fairly easy.

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 proposed to diagnose potential problems with a guideline using logical abduction [6, 9]. 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 this paper, we have made use of the interactive theorem prover KIV [1] to actually quality check a medical guideline using the theory of quality of guidelines developed previously [7]. This complements the earlier work on object-level verification of medical guidelines using KIV [8]. About half of the steps that were needed to complete the proofs had to be done manually. Fortunately, most of the interactive steps were rather straightforward. We are confident that with more specific heuristics, the proposed meta-level approach can be almost fully automated in KIV.

References

1. Balser M, Reif W, Schellhorn G, and Stenzel K. KIV 3.0 for provably correct systems. In: Current Trends in Applied Formal Methods, Lecture Notes in Computer Science 1641, Springer-Verlag, Berlin, 1999.
2. Balser M, Duelli C, Reif W, and Schellhorn G. Verifying Concurrent Systems with Symbolic Execution. Journal of Logic and Computation 2002; 12(4): 549–560.
3. Gabbay DM. The declarative past and imperative future: executable temporal logic for interactive systems. In: H. Barringer (ed.). Temporal Logic in Specification, Lecture Notes in Computer Science 398, Springer-Verlag, Berlin, 1989. pp. 409–448.
4. Gabbay D, Pnueli A, Shelah S, and Stavi J. The Temporal Analysis of Fairness. Pros. 7th ACM Syrup. on Principles of Programming Languages, Las Vegas, 1980, 163–173.
5. Lucas PJF. Logic engineering in medicine. The Knowledge Engineering Review 1995; 10(2): 153–179.
6. Lucas PJF. Symbolic diagnosis and its formalisation. The Knowledge Engineering Review 1997; 12(2): 109–146.
7. Lucas PJF. Quality checking of medical guidelines through logical abduction. In: F. Coenen, A. Preece, A.L. Mackintosh (eds.). Proceedings of AI-2003 (Research and Developments in Intelligent Systems XX), Springer, London, pp. 309–321, 2003.
8. Marcos M, Balser M, Ten Teije A, and Van Harmelen F. From informal knowledge to formal logic: a realistic case study in medical protocols. Proceedings of the 12th EKAW-2002, 2002.
9. Poole D. A methodology for using a default and abductive reasoning system. International Journal of Intelligent Systems 1990, 5(5), 521–548.
10. Shahar Y, Miksch S, and Johnson P. The Asgaard project: a task-specific framework for the application and critiquing of time-oriented clinical guidelines. Artificial Intelligence in Medicine 1998; 14: 29–51.
11. Turner R. Logics for Artificial Intelligence. Ellis Horwood, Chichester, 1985.
12. Woolf SH. Evidence-based medicine and practice guidelines: an overview. Cancer Control 2000; 7(4): 362–367.