The Representation of Medical Reasoning Models in Resolution-based Theorem Provers*

Peter Lucas
Department of Computer Science, Utrecht University
P.O. Box 80.089, 3508 TB Utrecht
e-mail: lucas@cs.uu.nl

Abstract

First-order predicate logic essentially is a language to express knowledge concerning objects and relationships between objects in a domain. Many medical problems can be cast naturally in such terms. In this paper the suitability of logic as a knowledge-representation formalism in building medical expert system is investigated. In particular, we investigate the logical representation of three typical reasoning models in medicine: diagnostic, anatomical and causal reasoning. It turns out that each of these models has its own characteristic logical structure. Furthermore, the pragmatics of using theorem-proving techniques in consulting such logic-based medical expert systems is discussed. In particular, attention is paid to the use of a meta-level architecture to improve the applicability of theorem-proving techniques in building expert systems.

Keywords & Phrases: logic programming in medicine, medical knowledge representation, expert systems, theorem proving.

1 Introduction

In building medical expert systems, several obstacles may be encountered. One of these, the difficulty of collecting knowledge in the process of constructing an expert system, has drawn much attention [3, 21, 31, 32]. At the other end of the development cycle we have the validation of an expert system, which recently has been reviewed in several journals [30, 29]. However, only slow progress is being made in clarifying the potentials and limitations of representation formalisms for the actual encoding of medical knowledge. The present article attempts to shed some light on applying logic as a language for the representation of medical knowledge.

Logic is frequently taken as a language to which other, more specialized, languages for the representation of knowledge are mirrored. It is not hard to see why logic takes such a prominent position. Firstly, logic, in particular first-order predicate logic, is essentially a language to express knowledge concerning objects and relationships between objects. Many real-world problems can naturally be described in these terms. Secondly, the language has a well-defined syntax and a clean mathematical semantics. There can be no misunderstanding whatsoever what the precise meaning of a sentence in first-order predicate logic is. As

*Published in: *Artificial Intelligence in Medicine*, 5(5), 395–414, 1993

a knowledge-representation language, logic is one of the few languages taking such a firm position.

The author views logic as one of the major candidates as a knowledge-representation language in future-generation expert systems. The reasons for this are two-fold:

- Most other knowledge-representation languages exist in many different flavours; almost none of these languages is completely understood.
- Logic is the unifying framework for integrating expert systems and database systems.

In particular, databases based on the relational data model may be viewed as a set of formulas in first-order logic. Much work is now being done on extending the expressive power of database systems, moving the capabilities of these systems towards those of expert systems. DATALOG is a recent example of such a language [5]. The integration of medical database systems and expert systems is without doubt one of the most important challenges in medical informatics.

Interestingly, logic has seldom been used as a knowledge-representation formalism in building expert systems. Although there seems to be ample reasons to study the suitability of logic as a knowledge-representation language for building expert systems, only few researchers have actually investigated expert systems from this perspective [15, 18, 34]. The only expert systems having a basis akin to logic are those developed in the field of logic programming, usually applying PROLOG. It should, however, be noted that logic programming is only concerned with a subset of first-order logic (Horn clause logic), and that PROLOG is a programming language, still far removed from being a completely declarative knowledge-representation language.

In this paper, we investigate several common reasoning models in medicine, familiar from the artificial intelligence literature, and discuss the mappings of those models to a logical representation. The purpose of the translation is to obtain a representation that permits automated interpretation by a logic-based theorem prover. Our two working hypotheses are that the use of a logical language may be a means to reveal the underlying structure of a given medical problem, and that standard first-order logic is sufficiently flexible for the representation of a significant fragment of medical knowledge. The treatment of every model includes an example taken from a medical textbook, and we show some results produced by the theorem prover OTTER [28]. Finally, some results from our research aimed at making logic a practical language for building expert systems are presented. We start with a brief review of the most important principles of first-order logic.

2 The language of first-order logic

In this section, we briefly review the syntax and semantics of first-order predicate logic. Furthermore, some of the considerations to make logic a practical language for building expert systems are discussed. Only the notions required for understanding the remainder of this paper are discussed here (cf. [11, 24, 41]).

2.1 Syntax and semantics

First-order predicate logic essentially is a language to express knowledge concerning the relationship between individual objects and classes of object. Syntactically, each relation is expressed using a *predicate symbol*, such as P; the objects in the relation are denoted by *terms* t_i, which function as arguments to a predicate symbol thus yielding *atomic formulas*, or *atoms* for short, of the form $P(t_1, t_2, \ldots, t_n)$. A term either denotes an individual object, in which case it is called a *constant*, a class of objects, then called a *variable*, or expresses dependencies upon other objects, in which case we speak of a *function term*. These dependencies are expressed by means of a *function symbol*, denoting a mathematical function, having as arguments terms that again denote objects in the domain of discourse. In this paper, variables usually will be denoted by the letters x, y, z, u, v, w; the type of the other symbols can be determined from their position in the atom. Atoms are the user-defined building blocks from which, together with the *logical connectives* such as $\forall$ (universal quantifier) and $\rightarrow$ (implication), formulas in first-order logic may be constructed.

In logic-based theorem-proving programs, one often restricts the syntax of formulas to *clausal form*. A *clause* is a sentence consisting of a finite disjunction of literals, where a *literal* is an atom (positive literal) or a negation of an atom (negative literal). A *Horn clause* is a clause that contains at most one positive literal. A clause having null literals is called the *empty clause*, and will be denoted by $\Box$.

By convension, we assume that a collection of formulas $F_1, F_2, \ldots, F_n$, stands for the conjunction $F_1 \wedge F_2 \wedge \cdots \wedge F_n$. We shall frequently refer to such a collection of formulas as a *logical theory* T.

As mentioned in the introduction to this paper, one of the distinguishing features of first-order logic is its clean mathematical semantics. Assigning meaning to a formula in first-order logic amounts to interpreting its constituting symbols in a mathematical *structure of relations* $\mathcal{S}$. Basically, a domain of discourse D, for example medicine, is chosen and predicates and functions defined on this domain D are associated with the predicate and function symbols, repectively, appearing in the logical theory. The interpretation of a formula φ starts with determining which of the atoms in φ are *true* and which are *false* in the structure $\mathcal{S}$; next, the truth values obtained are combined using the meanings of the logical connectives. The empty clause $\Box$ is *false* by definition. We are mainly interested in formulas in which every occurring variable is bound by a quantifier (then called *sentences*), since the meaning of such formulas is completely determined by the problem domain in which the formula is interpreted, and not by the particular fill-in for the variables. A sentence φ being satisfied (i.e. *true*) within a structure $\mathcal{S}$ will be denoted by $\models_{\mathcal{S}} \varphi$. It is said that a sentence φ is a *semantic consequence* of a theory $T = \{\sigma_1, \sigma_2, \ldots, \sigma_n\}$, denoted by $T \models \varphi$, if we have that for every structure $\mathcal{S}$, $\models_{\mathcal{S}} (\sigma_1 \wedge \sigma_2 \wedge \cdots \wedge \sigma_n)$ implies $\models_{\mathcal{S}} \varphi$. We call the formula φ a *theorem* of T. The notion of semantic consequence corresponds to the intuitive idea of determining what is true given some initial information, such as concerning a patient.

Finally, we call a sentence φ *inconsistent* iff it is falsified in every structure $\mathcal{S}$, denoted by $\not\models \varphi$.

2.2 Logical data representation in medicine

In medical problem-solving usually some form of reasoning is carried out with regard to the complaints and signs, physiological states, etcetera of the patient. So, the precise representation of such medical data is of fundamental importance. In every medical domain, the following aspects can be distinguished:

1. A number of individual *objects* (e.g. patients, substances), which simply can be represented as constants in first-order predicate logic;

2. Properties of these objects (e.g. physiological states, levels of substances) which can be represented using predicate symbols or function symbols of the same name.

We now distinguish two types of properties of an object:

- Properties that are unique at a certain point in time, such as the age of a patient; we call such properties *singlevalued properties*.
- Properties for which several fill-ins may occur at the same time, such as complaints and signs of a patient; we call such properties *multivalued properties*.

If a singlevalued property is used to describe a physiological process we shall refer to such a property as a *parameter*. To express knowledge concerning multivalued properties of objects, we use atoms of the form $P(t_1, t_2, \ldots, t_n)$, for example

$$Sign(johnson, jaundice)$$
$$Sign(johnson, spider\_angiomas)$$

expresses that a patient named Johnson has jaundice and spider angiomas at the same time. Notice that we can enumerate specific properties of an object, signs of a patient here, simply by means of a collection of atoms. To express knowledge concerning singlevalued properties of objects, we use atoms of the form $f(t_1, t_2, \ldots, t_m) \circ t'$ where $\circ$ is the equality predicate, $=$, or one of the ordering predicates, such as $<$ (less-than ordering in number theory), specified in infix position as this is normal mathematical practice. For example, the fact that someone named Johnson is 30 years old, can be expressed by the following formula:

$$age(johnson) = 30$$

Note that adding a new fact concerning the age of Johnson, for example

$$age(johnson) = 40$$

would lead to an inconsistency. The reason for this is that equality $=$ is a reflexive, symmetric and transitive binary relation. Using these properties, it is possible to derive that the formula $30 = 40$ should hold, which, however, is inconsistent with the basic axioms of number theory. (One should observe that without the availability of these axioms to a theorem-proving program, equality is not handled in a satisfactory way.) Among others, atoms with the equality predicate will be used for the logical specification of values of parameters; such atoms will be referred to as *states*.

2.3 Logical deduction and reasoning strategy

One of the pleasant aspects of logic is that logical formulas may be manipulated to yield new formulas by syntactic operations, called *inference rules*, that can be completely understood in terms of the semantics introduced in Section 2.1. One of the inference rule frequently applied in logic-based theorem-proving programs is *binary resolution* [37]. Repeatedly applying an inference rule leads to a *derivation* or *deduction*. The formula φ being derived from a theory T is denoted by $T \vdash \varphi$. If we have that if $T \vdash \varphi$ then $T \models \varphi$, the inference rules are called *sound*. Only sound inference rules derive meaningful results. On the other hand, if we have that if $T \models \varphi$ then $T \vdash \varphi$, the collection of inference rules is called *complete*, i.e. they are capable of deriving all semantic consequences of a theory T.

Resolution is usually only applied to derive a contradiction, the empty clause $\Box$, because resolution is only complete for deriving contradictions (*refutation completeness*). This is not an important limitation, since we have that $T \models \varphi$ is equivalent to $T \cup \{\neg\varphi\} \models \Box$, i.e. any theorem that follows from the theory T may simply be added as a negation to T yielding a contradiction.

The binary resolution rule is only the most basic inference rule in theorem-proving programs. This rule can be very inefficient, since it may generate many redundant formulas, i.e. formulas not relevant with respect to the theorem to be proved. Several refinements to the original binary resolution rule have therefore been proposed. An example of such an inference rule is *hyperresolution* which carries out several binary resolution steps in one step [38]. Furthermore, dealing with equality and ordering predicates as introduced above, in logical deduction poses certain problems. Simply adding the axioms that define equality and ordering, as formulas to a logical theory, is not feasible. To deal with part of the meaning of equality several special inference rules have been designed, such as for example *demodulation* [40]. A *demodulator* is an equality axiom

$$t_1 = t_2$$

where t_1, t_2 are terms. Applying a demodulator to an atom in a formula in which the term t_1 occurs yields a new formula F' where t_1 has been replaced by t_2, possibly by substituting terms for variables appearing in t_1. Finally, ordering and equality predicates are usually dealt with by evaluating the resulting atoms to *true* or *false*. In the field of constraint logic programming currently much research is carried out in dealing with such predicates in a more declarative way [16].

In addition to a number of inference rules, theorem-proving programs provide *reasoning strategies* that impose certain restrictions on the size of the search space. A frequently employed reasoning strategy is the *set-of-support strategy* [39]. It must be stressed that it requires a lot of ingenuity on the part of the developer to find a strategy that behaves in a satisfactory way.

3 Medical reasoning models

Logical languages such as first-order predicate logic are general languages with no direct relationship to any problem area whatsoever. This lack of a direct relationship to a particular problem area, such as medicine, is in certain ways an advantage. The limitations that are encountered in the formalization of medical knowledge are restrictions of the language employed, and not of an unsatisfactory attempt of designing a special-purpose medical knowledge-representation language. Moreover, since classical logical languages such as propositional and first-order predicate logic are well-understood, all restrictions are known in advance.

However, the generality of logical languages also poses some problems, since translating medical knowledge into logical formulas is not supported by some special-purpose syntax or semantics. In the following, it will turn out that it is indeed possible to impose restrictions on the full syntax of first-order predicate logic based on features of the (medical) problem at hand.

In order to show the potentials of logic as a language for formalizing medical knowledge, we will distinguish several more or less typical medical reasoning models. Although any form of categorization of medical reasoning is arbitrary, this distinction will aid in characterizing the nature of logic-based knowledge representation in relationship with medical reasoning.

Here, we shall pay attention to three medical reasoning models:

- Heuristic, diagnostic reasoning;
- Anatomical reasoning;
- Causal reasoning.

Each reasoning model will be investigated by means of an example taken from a medical textbook or by knowledge gathered from interviewing a doctor.

3.1 Heuristic, diagnostic reasoning

There are several different languages known from the literature to formalize diagnostic reasoning. Examples of such languages, other than logic, are set theory and belief networks [33, 19]. In the present section, we focus on the logical representation of diagnostic reasoning in the spirit of MYCIN-like rule-based expert systems [4, 9]. In this formalization, diagnostic reasoning is viewed as a deductive process instead of as an abductive process, the other frequently adopted view of diagnostic reasoning [34].

We describe our attempt to reformulate the HEPAR system, a rule-based expert system for the diagnosis of disorders of the liver and biliary tract, to first-order predicate logic (actually, many-sorted predicate logic, but we disregard the sorts in this article) [25]. The problems encountered in doing so, and some of the solutions found will be discussed.

The formalization of medical diagnostic reasoning involves two different aspects. First, some suitable logical representation of patient data must be chosen. Second, we have to decide on the logical representation of diagnostic medical knowledge. In Section 2.2, we have dealt with logical data representation in medicine; the forms introduced in that section immediately can be used for the representation of patient data. For example, the data of a 12-year old patient (which has been used in the validation of the HEPAR system) can be expressed in first-order logic as follows:

$$\begin{array}{l}
\mathit{sex}(\mathit{patient1}) = \mathit{female} \\
\mathit{age}(\mathit{patient1}) = 12 \\
\mathit{Complaint}(\mathit{patient1}, \mathit{arthralgia}) \\
\mathit{time_course}(\mathit{patient1}, \mathit{illness}) = 150 \\
\qquad\qquad \vdots \\
\mathit{Signs}(\mathit{patient1}, \mathit{Kayser_Fleischer_rings}) \\
\qquad\qquad \vdots \\
\mathrm{ASAT}(\mathit{patient1}, \mathit{labresult}, \mathit{biochemistry}) = 200 \\
\mathit{urinary_copper}(\mathit{patient1}, \mathit{labresult}, \mathit{biochemistry}) = 5 \\
\qquad\qquad \vdots
\end{array}$$

As can be observed, several singlevalued and multivalued properties are expressed as formulas consisting of a single atom (unit clauses). This logical representation differs from a more database-oriented representation as sometimes employed in PROLOG-like language [1]. In this case, the representation language is primarily viewed as a term manipulation language, not as a logical language. Then, the information above would be represented as a single term, for example as follows:

$$
\begin{aligned}
&\mathit{patient}(\mathit{name} = \mathit{patient1},\\
&\qquad \mathit{sex} = \mathit{female},\\
&\qquad \mathit{age} = 12,\\
&\qquad\quad \vdots\\
&\qquad \mathit{complaint} = [\mathit{arthralgia}],\\
&\qquad\quad \vdots\\
&\qquad)
\end{aligned}
$$

similar to a record in traditional database systems.

Although the representation of patient data in logic seems straightforward, the representation of negative information (such as findings observed to be absent) is not. We shall discuss this matter further after discussing the representation of diagnostic medical knowledge.

Diagnostic medical knowledge is represented in the HEPAR system using production rules with object–attribute–value tuples. According to the declarative reading of rules, translation of most production rules is straightforward yielding logical implications [2, 24]. An example of such a logical implication concerning Wilson's disease is shown below:

$$
\begin{aligned}
&\forall x(\mathit{Duration}(x, \mathit{complab}, \mathit{chronic}) \wedge\\
&\quad (\mathit{disorder}(x) = \mathit{hepatocellular}) \wedge\\
&\quad (\mathit{age}(x) < 25) \wedge\\
&\quad (\mathit{caeruloplasmin}(x, \mathit{labresult}, \mathit{biochemistry}) > 20) \wedge\\
&\quad (\mathit{urinary\_copper}(x, \mathit{labresult}, \mathit{biochemistry}) > 1)\\
&\quad \rightarrow \mathit{Diagnosis}(x, \mathit{Wilson\_s\_disease}))
\end{aligned}
$$

Note that this formula (after translation to clausal form) conforms to the syntax of a Horn clause. However, we discovered that more than 50% of the production rules in the HEPAR system could only be translated to non-Horn clauses. This was partly due to the occurrence of negative conditions in the original production rules, and partly due to the presence of multiple (positive) conclusions in rules (which were translated to disjunctions). So, a Horn-clause theorem prover would be insufficient as an interpreter of the resulting logical theory.

Diagnostic reasoning in medicine typically involves reasoning about diagnostic categories. For example, in the domain of the HEPAR system, a clinician first tries to establish whether the patient suffers from an acute, subacute or chronic disorder, from a hepatocellular or biliary obstructive disorder, and whether features are present indicating that the disorder is malignant [25]. Where in the original version of HEPAR, the specification of knowledge concerning diagnostic categories is accomplished in production rules, in the logical version of HEPAR logical implications are used for this purpose. For example, the following implication concludes about the chronic nature of the disorder of patients x:

$$
\begin{aligned}
&\forall x((\mathit{time\_course}(x, \mathit{illness}) > 26) \wedge\\
&\quad (\mathit{Signs}(x, \mathit{spider\_angiomas}) \vee\\
&\quad\ \mathit{Signs}(x, \mathit{palmar\_erythema}) \vee\\
&\quad\ \mathit{Signs}(x, \mathit{Kayser\_Fleischer\_rings}))\\
&\quad \rightarrow \mathit{Duration}(x, \mathit{complab}, \mathit{chronic}))
\end{aligned}
$$

Note that the literal $\mathit{Duration}(x, \mathit{complab,chronic})$ occurs in the previous implication concerning Wilson's disease.

Given the data of a specific patient, represented as a collection of unit clauses D, and the diagnostic theory T, diagnostic problem solving using a resolution-based theorem prover like OTTER amounts to establising

$$D \cup T \cup \{\neg \mathit{Diagnosis}(x, y)\} \vdash \Box$$

for every patient name substituted for the variable x and every possible disorder substituted for the variable y. In the OTTER system, this can be accomplished by using a combination of binary resolution, hyperresolution, demodulation and evaluation of atoms containing numeric constants, in conjunction with the set-of-support strategy.

There remain certain aspects of HEPAR that have no suitable equivalent in classical logic. In the original version of the HEPAR system:

1. A form of closed world assumption (CWA) is taken to hold true [35]. With regard to all tests carried out for a patient, all findings not observed in the patient to be present are (implicitly) assumed to be absent.

2. A diagnostic strategy is incorporated. Even the relatively simple diagnostic strategy in HEPAR could not easily be expressed in the language facilities provided by the OTTER theorem prover.

3. Some reasoning concerning the derivability of facts using the *notknown* meta-predicate is carried out. There is no analogous notion in standard first-order logic.

4. Reasoning with uncertain knowledge is possible using the certainty factor model of Shortliffe and Buchanan [4].

To deal with unavailable patient data, for all investigations for which unit clauses were present, negative unit clauses were added if the property concerned was multivalued, thus explicitly stating the remaining test results to be negative. For example, to the clauses concerning the patient discussed above, one of the unit clauses added was the following:

$$\neg \mathit{Complaint}(\mathit{patient1,fever})$$

indicating that it was assumed that the patient did not have a fever. Note the difference with the closed world assumption as it appears in the logic programming literature; here a literal $\neg P(t_1, t_2, \ldots, t_n)$ is assumed unless the positive literal $P(t_1, t_2, \ldots, t_n)$ can be derived from the theory [22]. The choice of our more restrictive method for handling negative information was also motivated by the fact that the rule-based version of HEPAR contains several production rules with negative conclusions. Instead of completely relying on the closed world assumption, separating out the CWA and the classical logical approach to negative information seemed more appropriate.

In our experiments with the OTTER system and a resolution-based theorem prover we developed in COMMON LISP, we found that general, domain-independent reasoning methods were insufficient for imposing a clear conceptual structure on the reasoning process. As a solution to this problem, we have investigated the use of a meta-level architecture [27]. In our COMMON LISP theorem-proving program, a knowledge base consists of two levels: an *object-level* containing the declarative domain knowledge from HEPAR, and a *meta-level* containing domain-specific control primitives. The actual application of inference rules at the object-level is controlled by the meta-level primitives.

The meta-level control primitive that turned out to be particularly effective was what we called the *pattern matching ordering primitive*, or pmo primitive for short. A pmo primitive is defined as an ordered list

$$(L_1, L_2, \ldots, L_n)$$

where each L_i is a list of patterns $(p_1, p_2, \ldots, p_m)$. A pattern p_j is either a positive or negative literal. Because these literals may contain arbitrary terms, their pattern-matching capability is quite general. Patterns specified in the pmo primitive are used in the selection of relevant clauses, in conjunction with the set-of-support strategy, to which resolution can be applied. The elements L_i are processed in the order of specification, one at the time. The total order of the patterns in the list of patterns is used to sort the clauses in accordance with this order. The pmo primitive provides a level of abstraction similar to the notion of generic task as proposed in [6]. Application of a pmo primitive may lead to a loss in completeness, but refutation completeness can be preserved by careful design of the meta-level of an expert system.

3.2 Anatomical reasoning

In the previous section we have discussed the kind of diagnostic reasoning employed in expert systems primarily based on heuristic knowledge. However, in certain fields of medicine, for example neurology, knowledge concerning the anatomy of the human body is at least as important. The form of automated reasoning in which knowledge concerning the anatomy of the human body is applied, is known as *anatomical reasoning*.

The point of departure for any expert system implementing anatomical reasoning, is the axiomatization of the basic anatomical relations. The more precise the description of the anatomical structures must be, the more complex the resulting axiomatization will be. Not always is a precise three-dimensional specification of anatomical relations required. In our approach to anatomical reasoning, it suffices to indicate only that certain anatomical structures are connected to each other in a qualitative way, as axiomatized by the *Connected* predicate. This predicate is defined as a transitive, irreflexive relation, as follows:

$$\forall x \forall y \forall z (\mathit{Connected}(x, y) \land \mathit{Connected}(y, z) \to \mathit{Connected}(x, z))$$
$$\forall x (\neg \mathit{Connected}(x, x))$$

Note that the *Connected* predicate is by its transitive and irreflexive properties also antisymmetric. (So, a theorem prover is capable of detecting an inconsistency given the formulas $\mathit{Connected}(a, b)$ and $\mathit{Connected}(b, a)$ from the two axioms given above.)

As a starting point for our discussion concerning the specification of anatomical reasoning in logic, we consider an actual example (the diagnosis of lesions of the facial nerve) taken from a textbook of neurology [8]. We start by summarizing the relevant sections in this textbook.

The classical picture of *facial palsy* is well-known. These patients have a mouth that droops and may draw to the opposite side. They cannot wrinkle the forehead or close the eye at the affected side. Facial palsy is due to a lesion of the facial nerve (cranial nerve VII); this nerve can be affected by a large variety of disorders. The severity and nature of the complaints and signs that may be observed in the patient depend on the level of facial nerve lesion. Knowledge of the branching pattern of the nerve and the consequences of a

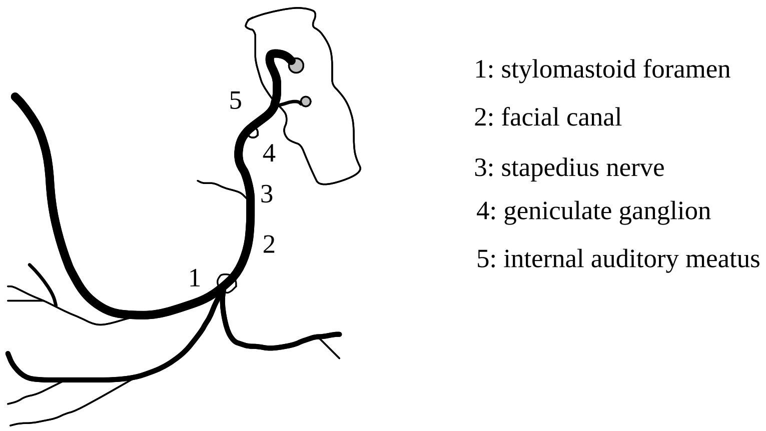

Figure 1: Levels in the facial nerve.

lesion of a particular branch is important in diagnostic problem solving. Figure 1 gives a schematic overview of the branching pattern of this nerve. The facial nerve is a mixed nerve; it contains motor fibers that supply striated muscle fibers, sensory fibers that carry taste from the anterior two-third of the tongue and some sensation, and parasympathetic fibers.

The facial nerve emerges from the brain stem and leaves the skull via the internal auditory meatus. Next, a small nerve is branched off (the stapedius nerve), that supplies the stapedius muscle (a small muscle that is attached to the ear drum, regulating its tension). The facial nerve proceeds its way through the facial canal (in the temporal bone), next branching off the chorda tympani, a nerve mostly consisting of parasympathetic fibers that supply the submandibular and sublingual glands. The facial nerve leaves the facial canal through the stylomastoid foramen. Finally, it splits up in a number of branches that supply the superficial musculature of the face and scalp (e.g. orbicularis oris et oculi, buccinator, platysma).

A bit simplified, we distinguish the following five levels of facial nerve lesions (consult again Figure 1):

Level 1: A lesion outside the stylomastoid foramen produces signs such as drooping of the mouth. The patient cannot whistle, wink, close the eye or wrinkle the forehead. When the patient attempts to close the eye, the eye bulb will turn upward (*Bell's sign*).

Level 2: A lesion of the nerve in its course through the facial canal will result in all the signs as present in a level 1 lesion, but in addition there is reduced salivation (lesion of the chorda tympani) and loss of taste in the anterior two-thirds of the tongue.

Level 3: All signs of a level 2 lesion are present, but in addition the stapedius nerve is affected, causing hyperacusis (due to paralysis of the stapedius muscle).

Level 4: A lesion of the geniculate ganglion is usually due to herpes zoster, in which case herpetic lesions are visible on the ear drum and external auditory canal (*Ramsay Hunt syndrome*). Typically, a patient will experience pain in and behind the ear.

Level 5: A lesion in the internal auditory meatus is usually associated with acoustic nerve (cranial nerve VIII) involvement, because the last nerve also runs through this canal. In addition to the signs mentioned for lower level lesions, deafness will be present.

This completes the description of the neurological knowledge involved in diagnosing facial nerve lesions.

In formalizing this knowledge using first-order predicate logic, we start by completing the axiomatization of the anatomical relationships by giving a domain-specific fill-in for the *Connected* predicate. The atom $Connected(x, y)$ is intended to mean that the facial nerve runs from level x up to level y:

$$Connected(stylomastoid\_foramen, chorda\_tympani)$$
$$Connected(chorda\_tympani, stapedius\_nerve)$$
$$Connected(stapedius\_nerve, geniculate\_ganglion)$$
$$Connected(geniculate\_ganglion, internal\_auditory\_meatus)$$

Note that we have employed anatomical terms to denote the various levels.

To relate anatomical structures and signs that may arise due to facial nerve lesion, we have to express that the signs associated with a lesion at a certain level x includes all the signs of a lesion at a lower level y:

$$\forall x \forall y (Lesion(x) \wedge Connected(y, x) \rightarrow Lesion(y))$$

This completes our axiomatization of the knowledge that forms the basis of logical anatomical reasoning.

We next specify the relationship between a lesion at a certain level and the specific anatomical structures that will be affected by this lesion, expressed by the unary predicate symbol *Affected*. We use a bi-implication, because given a lesion at a certain level we may want to know which structures will be affected by this lesion; on the other hand, the observation of malfunction of certain structures may be interpreted as evidence for a lesion at a certain level:

$$\begin{aligned} &(Lesion(stylomastoid\_foramen) \leftrightarrow \\ &\quad (Affected(orbicularis\_oris) \wedge \\ &\quad\; Affected(orbicularis\_oculi) \wedge \\ &\quad\; Affected(buccinator) \wedge \\ &\quad\; Affected(frontalis\_muscle) \wedge \\ &\quad\; Affected(platysma))) \end{aligned}$$

$$\begin{aligned} &(Lesion(chorda\_tympani) \leftrightarrow \\ &\quad (Affected(sensory\_taste\_fibers) \wedge \\ &\quad\; Affected(sublingual\_gland) \wedge \\ &\quad\; Affected(submaxillary\_gland))) \end{aligned}$$

$$(Lesion(stapedius\_nerve) \leftrightarrow Affected(stapedius\_muscle))$$

$$(Lesion(geniculate\_ganglion) \leftrightarrow Affected(sensory\_fibers\_ear))$$

$$(Lesion(internal\_auditory\_meatus) \leftrightarrow Affected(acoustic\_nerve))$$

Finally, paralysis of certain muscles and disturbed sensation will give rise to specific signs and complaints in the patient. This knowledge is again expressed using a collection of bi-implications:

$$\begin{aligned} &(Affected(orbicularis\_oris) \leftrightarrow \; (Sign(mouth\_droops) \wedge \\ &\quad Sign(cannot\_whistle))) \end{aligned}$$

$(\mathit{Affected}(\mathit{orbicularis\_oculi}) \leftrightarrow (\mathit{Sign}(\mathit{cannot\_close\_eyes}) \wedge \mathit{Sign}(\mathit{Bell})))$

$(\mathit{Affected}(\mathit{buccinator}) \leftrightarrow \mathit{Sign}(\mathit{flaccid\_cheeks}))$

$(\mathit{Affected}(\mathit{frontalis\_muscle}) \leftrightarrow \mathit{Sign}(\mathit{cannot\_wrinkle\_forehead}))$

$(\mathit{Affected}(\mathit{platysma}) \leftrightarrow \mathit{Sign}(\mathit{paresis\_superficial\_neck\_musculature}))$

$((\mathit{Affected}(\mathit{sublingual\_gland}) \wedge \mathit{Affected}(\mathit{submaxillary\_gland})) \leftrightarrow$
$\mathit{Complaint}(\mathit{dry\_mouth}))$

$(\mathit{Affected}(\mathit{sensory\_taste\_fibers}) \leftrightarrow \mathit{Complaint}(\mathit{taste\_loss\_anterior\_part\_tongue}))$

$(\mathit{Affected}(\mathit{stapedius\_muscle}) \leftrightarrow \mathit{Complaint}(\mathit{hyperacusis}))$

$(\mathit{Affected}(\mathit{sensory\_fibers\_ear}) \leftrightarrow$
$(\mathit{Complaint}(\mathit{pain\_behind\_ear}) \wedge$
$\mathit{Complaint}(\mathit{pain\_within\_ear}) \wedge$
$\mathit{Sign}(\mathit{herpetic\_lesions})))$

$(\mathit{Affected}(\mathit{acoustic\_nerve}) \leftrightarrow \mathit{Complaint}(\mathit{deafness}))$

Let T be the logical theory given above. For example, after automatic translation to clausal form, using hyperresolution the OTTER system is now capable to derive:

$$T \cup \{\mathit{Lesion}(\mathit{stapedius\_nerve})\} \cup \{\neg \mathit{Sign}(x)\} \cup \{\neg \mathit{Complaint}(y)\} \vdash \Box$$

where for x we have *mouth_droops*, *cannot_whistle*, *cannot_close_eyes*, *Bell*, *flacid_cheeks*, *cannot_wrinkle_forehead* and *paresis_superficial_neck_musculature*; for y we have *hyperacusis*, *dry_mouth* and *taste_loss_anterior_part_tongue*. Note that all results but the complaint *hyperacusis* have been derived using the anatomical axioms for the *Connected* predicate. Reasoning from signs and complaints to the level of a facial nerve lesion is also possible (essentially employing the bi-implication), but here we need a meta-level primitive that selects from the unit clause concerning *Lesion* the one specifying knowledge regarding the highest level of the lesion.

In our formalization of the anatomical reasoning model, we did not make a distinction between left- and right-sided lesions of the facial nerve. The extension of the logical theory to include this distinction is straightforward. However, this is not always the case. In our design of a logical specification of the human visual system, we obtained terms such as

$$\mathit{lateral}(\mathit{left}(\mathit{structure}))$$

where *structure* is a universally quantified variable. Presented with such terms, a theorem prover like OTTER will end up in endless search, deriving deeply nested terms such as

$$\mathit{lateral}(\mathit{left}(\mathit{left}(\mathit{left}(\mathit{retina}))))$$

Although declaratively correct, such information will generally not be very useful. Using a meta-level architecture approach in which a meta-level primitive is included that prefers clauses with flat terms may help, but we have not yet investigated this approach.

3.3 Causal reasoning

Although in the previous section we primarily focussed on the logical specification of anatomical relations, implicitly other reasoning models were also involved. Some of the formulas presented in the previous section express some relationship between cause and effect of nerve damage, thus representing *causal knowledge*. The reasoning about such cause–effect relationships is known as *causal reasoning*. In the present section, we shall study the logic of causal reasoning in medicine, taking the logical specification of physiological processes as a point of departure.

The representation of causal knowledge in logic is rather straightforward; it may be represented by means of a collection of logical implications of the following form:

$$\mathit{cause} \rightarrow \mathit{effect}$$

where both *cause* and *effect* are conjunctions of literals. Most literals refer to the *state* of some *parameter*; the states of all parameters together describe the entire physiological process. As an example of a parameter consider the level of a substance in the blood; the actual level of the substance stands for the parameter's state.

In Section 2.2 we have already introduced the logic of state representation in medicine. A state is either *numeric* or *qualitative*. An example of a numeric state (of parameter level of *sodium* in the *blood*) is expressed by the following unit clause:

$$\mathit{conc}(\mathit{blood}, \mathit{sodium}) = 125$$

In clinical practice, numeric parameters are often changed to qualitative states. In the above case, we get:

$$\mathit{conc}(\mathit{blood}, \mathit{sodium}) = \mathit{decreased}$$

There are several common types of causal reasoning in medicine. We shall study the *negative feedback process* and its logical specification in some detail.

In terms of cause–effect relationships, the global specification of a negative feedback process leads to the following logical theory T (to simplify matters, we have assumed that a cause consists of a single literal):

$$\begin{array}{l} s \rightarrow r_1 \\ r'_1 \rightarrow r_2 \\ \quad \vdots \\ r'_{n-1} \rightarrow r_n \\ r'_n \rightarrow \neg s \end{array}$$

where s, r_i, r'_i, $1 \leq i \leq n$, $n \geq 1$ are literals in first-order logic; the literals r_i, r'_i are similar, in the sense that substitution of terms for variables occurring in these literals can make them syntactically equal. Note that we have $T \models \neg s$; in words: the negative feedback is a semantic consequence of the process description.

To investigate the applicability of this approach to formalizing causal reasoning in medicine, we have chosen a particular example of a negative feedback process from the literature, viz. the renin–angiotensin–aldosterone system. The description is taken from a general textbook of physiology [14]. We start by giving a brief summary of the medical knowledge involved.

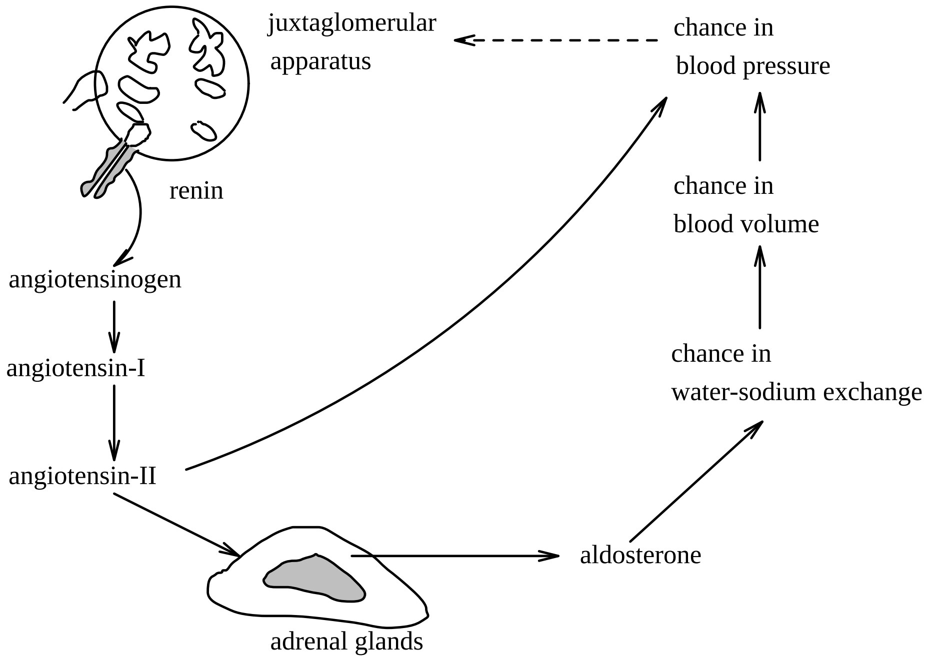

Figure 2: The renin–angiotensin–aldosterone system.

The regulation of the blood pressure is accomplished by the collaborative effort of a number of control systems in the human body. One of these control systems is the *renin–angiotensin–aldosterone system*. Figure 2 gives a pictorial overview of this control system. We review the regulatory factors involved.

The proteolytic enzyme *renin* is released by cells of the *juxtaglomerular apparatus* in the kidneys. A decrease in the mean renal arterial pressure increases the renin secretion, and an increase in the mean renal arterial pressure leads to a decrease in the renin secretion. Renin acts on an α_2-globulin (*angiotensinogen*) that circulates in the blood, liberating the decapeptid *angiotensin-I*. In turn, the octapeptid *angiotensin-II* is liberated from *angiotensin-I* by *angiotensin converting enzyme* (ACE) that is produced in the lungs.

Angiotensin-II is a powerful arteriolar constrictor; administration of angiotensin-II leads to an increase in arterial blood pressure. In addition, it stimulates the secretion of *aldosterone*, a hormone produced by the adrenal cortex; an increase in angiotensin-II levels in the blood increases aldosterone levels in the blood. Aldosterone stimulates active reabsorption of Na^+ from the urine and the secretion of K^+ to the urine. Water moves with the reabsorbed Na^+ to the blood, which causes an increase in blood volume. This in turn leads to an increase in blood pressure. Finally, an increase in blood pressure inhibits the secretion of renin. This completes our description.

When considering the physiological process described above in terms of a causal model, we have to analyse its behaviour in terms of causes and effects. We start this formalization by introducing a number of predicate and function symbols and constants that will be used to represent parameters and states. For the representation of the level of a substance in the blood we employ the binary function symbol *conc*. The unary function symbol *pressure* stands for blood pressure. The conversion of one substance into another substance by some enzyme, will be represented by the binary function symbol *conversion*. Finally, we distinguish two constants: *decreased* and *increased* to express the states of various parameters in a qualitative way. Step by step, the text given above will now be translated into a logical theory T.

A decrease in the blood pressure yields an increase in renin blood levels:

$$\mathit{pressure}(\mathit{blood}) = \mathit{decreased} \rightarrow \mathit{conc}(\mathit{blood}, \mathit{renin}) = \mathit{increased}$$

The relationship between renin levels in the blood and conversion of angiotensinogen into angiotensin-I is expressed as follows:

$$\forall v(\mathit{conc}(\mathit{blood}, \mathit{renin}) = v \rightarrow \mathit{conversion}(\mathit{angiotensinogen}, \mathit{angiotensin\_I}) = v)$$

where the universally quantified variable v stands for *increased* or *decreased*.

The relationship between decreased or increased conversion of angiotensinogen into angiotensin-I is represented by means of the following logical implication:

$$\forall v(\mathit{conversion}(\mathit{angiotensinogen}, \mathit{angiotensin\_I}) = v \rightarrow \\ \mathit{conc}(\mathit{blood}, \mathit{angiotensin\_I}) = v)$$

A change v in the blood level of angiotensin-I leads to an inverse change in the ACE levels, and a similar change in angiotensin-II levels:

$$\forall v(\mathit{conc}(\mathit{blood}, \mathit{angiotensin\_I}) = v \rightarrow \\ (\neg(\mathit{conc}(\mathit{blood}, \mathrm{ACE}) = v) \wedge \mathit{conversion}(\mathit{angiotensin\_I}, \mathit{angiotensin\_II}) = v))$$

$$\forall v(\mathit{conversion}(\mathit{angiotensin\_I}, \mathit{angiotensin\_II}) = v \rightarrow \\ \mathit{conc}(\mathit{blood}, \mathit{angiotensin\_II}) = v)$$

Angiotensin-II produces arterial vasoconstriction and an increase in aldosterone levels:

$$\mathit{conc}(\mathit{blood}, \mathit{angiotensin\_II}) = \mathit{increased} \\ \rightarrow \\ (\mathit{Vasoconstriction}(\mathit{arteries}, \mathit{peripheral}) \wedge \\ \mathit{conc}(\mathit{blood}, \mathit{aldosterone}) = \mathit{increased})$$

Arterial vasoconstriction produces an increase in bloodpressure:

$$\mathit{Vasoconstriction}(\mathit{arteries}, \mathit{peripheral}) \rightarrow \mathit{pressure}(\mathit{blood}) = \mathit{increased}$$

An increase in the aldosterone levels results in an increase of blood sodium and a decrease of the potassium levels:

$$\mathit{conc}(\mathit{blood}, \mathit{aldosterone}) = \mathit{increased} \rightarrow \\ (\mathit{conc}(\mathit{blood}, \mathit{sodium}) = \mathit{increased} \wedge \mathit{conc}(\mathit{blood}, \mathit{potassium}) = \mathit{decreased})$$

The reabsorption of sodium is accompanied by the reabsorption of water, causing an increase in blood volume; more in general, a change in sodium level causes a change in blood volume:

$$\forall v\ (\mathit{conc}(\mathit{blood}, \mathit{sodium}) = v \rightarrow \mathit{volume}(\mathit{blood}) = v)$$

The change in blood volume causes a similar change in cardiac output:

$$\forall v(\mathit{volume}(\mathit{blood}) = v \rightarrow \mathit{output}(\mathit{heart}) = v)$$

A change in cardiac output causes the same change in blood pressure:

$$\forall v(\mathit{output}(\mathit{heart}) = v \rightarrow \mathit{pressure}(\mathit{blood}) = v)$$

A change in bloodpressure causes an inverse change in renin levels:

$$\forall v(\mathit{pressure}(\mathit{blood}) = v \rightarrow \neg(\mathit{conc}(\mathit{blood}, \mathit{renin}) = v))$$

Finally, we need to express that ‘increased’ and ‘decreased’ are different notions:

$$\neg(\mathit{increased} = \mathit{decreased})$$

This completes our formalization of the causal knowledge concerning the renin–angiotensin–aldosterone system.

After (automatic) conversion of this logical theory T to clausal form, a theorem prover like OTTER is capable of deriving in six steps

$$T \cup \{\mathit{pressure}(\mathit{blood}) = \mathit{decreased}\} \vdash \Box$$

the last step being the derivation of

$$\mathit{pressure}(\mathit{blood}) = \mathit{increased}$$

among others via the intermediate derivation of

$$\mathit{Vasoconstriction}(\mathit{arteries}, \mathit{peripheral})$$

yielding a contradiction with

$$\mathit{pressure}(\mathit{blood}) = \mathit{decreased}$$

Although, we shall not go into the details, it is easy to combine the kind of causal reasoning (often called ‘deep knowledge’) with ‘heuristic rules’ (also called ‘surface knowledge’) as discussed in Section 3.1 using a logical approach. For example, the following logical implication expresses heuristic knowledge concerning renovascular hypertension, but ‘interfaces’ by two of its conditions to the causal reasoning system discussed above:

$$\begin{array}{l}((\mathit{pressure}(\mathit{blood}) = \mathit{increased}) \wedge \\ \ (\mathit{conc}(\mathit{blood}, \mathit{renin}) = \mathit{increased}) \wedge \\ \ \mathit{Investigation}(\mathit{angiogram}, \mathit{positive})) \\ \rightarrow \mathit{Diagnosis}(\mathit{patient}, \mathit{renovascular\_hypertension})\end{array}$$

4 Discussion

In this paper, we investigated the applicability of logic as a language for the representation of a number of medical reasoning models. It appeared that each of the models examined had its own characteristic logical structure. It was shown that the language of first-order predicate logic allowed for the precise, and compact, representation of these models. Translation of the medical knowledge concerned could be carried out in a simple, straightforward way. Generally, in translating domain knowledge into logic, many of the subtleties that can be expressed in natural language are lost. In our study, it appeared that this problem was less prominently present, because we took medical reasoning models as a point of departure for the formalization. Although a significant portion of medical knowledge may be accessible to formalization in logic, for many problem types in medicine, logic will not be the first language of choice. Examples of such problems are medical decision making under uncertainty and therapy planning.

In our experiments, we have refrained from using non-monotonic logics, a subject of much ongoing research. Non-monotonic logics arise, for example, when dealing with incomplete knowledge. Work on the use of non-monotonic logic in medicine has been done by Console *et al.*, who have investigated the logic of diagnosis in incomplete causal models [10]. A disadvantage of such non-monotonic logics is that they are far less well-understood than standard logic. Another approach to non-monotonic logics is to express the non-monotonicity at the meta-level and to adopt a form of meta-level reasoning. The advantage of such an approach is that the representation language remains standard logic, while at the same time gaining flexibility [18, 36]. This is what we actually have attempted to do; in our experiments we have completely remained within standard first-order logic. The use of first-order predicate logic in building medical expert systems has also been advocated in [12]. However, in this paper a significant part of the medical domain knowledge is represented by means of meta-level logical schemata. As far as we know, a clean mathematical foundation of this approach has never been provided.

Although the problems taken as examples in this paper were relatively small, the logical expert systems we developed for larger domains indicate that the techniques discussed remain applicable. Large applications will require additional machinery, such as the modularization of the logical theories. Techniques for the modularization of logical theories have been studied in the related field of algebraic specification languages [13]. Furthermore, our results indicate that when using general-purpose inference rules in logic-based medical expert systems solving real-life problems, some domain-specific forms of automated reasoning control are required. A meta-level architecture, where the meta-level consists of a number of domain-specific control primitives, as applied in our experiments concerning diagnostic reasoning, may be useful in this case.

Several problems in using logic for building medical expert systems require further study. In translating the HEPAR system to many-sorted logic, we disregarded the uncertainty that went with the medical knowledge represented. As a consequence, the advice produced by the logical version of HEPAR does not provide ordering information as to which conclusion has the strongest support. With regard to the classification of a patient into final diagnostic categories, the effect was not significant, as was to be expected, because it was previously shown in the original version of HEPAR, using a database with patient data, that on the average 4 conclusions were selected by the system out of the 80 possible diagnostic conclusions [26]. Even without the availability of ordering information, such advice is still valuable. However, the result of the translation was less successful with respect to the intermediate conclusions produced by HEPAR, because here the uncertainty attached is used as information to pursue certain final diagnoses. We think that a meta-level reasoning approach may solve part of the problem, a subject of future research.

Although a restrictive logical approach to anatomical reasoning may be applicable to many other problems than those we have experimented with, it is to be expected that when the axiomatization of the anatomical relations becomes more involved, it will be difficult to keep the logical reasoning process under control. On the other hand, it has been shown that when a theorem prover is used intelligently, even a complete Tarskian axiomatization of plane geometry can be handled.

The classical approach to the formalization of causal reasoning is control theory, which provides means for the design of flexible time-varying models. The qualitative model we developed in first-order logic has the advantage over a quantitative modeling technique that the model's structure is more clearly revealed, and that no presuppositions of the linearity of the model are made. In applications in which the structural, qualitative aspects of a model are more important than numerical detail, such as in clinical medicine, this approach may be sufficiently powerful.

Acknowledgement. Implementation of the resolution-based theorem prover in COMMON LISP was carried out together with Bob van den Berg, who developed most of the program.

References

[1] H. Aït-Kaci and R. Nasr, LOGIN: a logic programming language with built-in inheritance. *The Journal of Logic Programming*, **3** (1986) 185-215.

[2] M. Bezem, Consistency of rule-based expert systems, in: E. Lusk, R. Overbeek (eds.), *Proceedings of the 9th International Conference on Automated Deduction*, (Springer-Verlag, Berlin, 1988) 151-161.

[3] J. Breuker, B. Wielinga, Models of expertise in knowledge acquisition, in: G. Guida, C. Tasso (eds.), *Topics in expert systems design,* (North-Holland, Amsterdam, 1989) 265-295.

[4] B.G. Buchanan, E.H. Shortliffe, *Rule-based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project,* (Addison-Wesley, Reading, Massachusetts, 1984).

[5] S. Ceri, G. Gottlob, L. Tanca, *Logic programming and databases,* (Springer-Verlag, Berlin, 1989).

[6] B. Chandrasekaran and S. Mittal, Conceptual representation of medical knowledge for diagnosis by computer: MDX and related systems, in: M.C. Yovits (ed.), *Advances in Computers* **22**, (Academic Press, London, 1983).

[7] C.L. Chang, R.C.T. Lee, *Symbolic Logic and Mechanical Theorem Proving*, (Academic Press, New York, 1973).

[8] J.G. Chusid, *Correlative Neuroanatomy & Functional Neurology,* (Lange, Los Altos, 1976).

[9] W.J. Clancey, Heuristic Classification, *AI Journal*, **27** (1975) 289-350.

[10] L. Console, D.T. Dupré and P. Torasso, A theory of diagnosis for incomplete causal models, *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, (Morgan-Kaufman, Los Altos, 1990) 1311-1317.

[11] H.B. Enderton, *A Mathematical Introduction to Logic,* (Academic Press, London, 1972).

[12] J. Fox, D.A. Clark, A. Glowinski, M.J. O'Neil, Using predicate logic to integrate qualitative reasoning and classical decision theory, *IEEE Transactions on Systems, man, and Cybernetics,* **20** (1990) 347-357.

[13] K. Futatsugi, J.A. Goguen, J.P. Jouannaud, J. Meseguer, Principles of OBJ2, in: *Proceedings of the Symposium on Principles of Programming Languages* (ACM, New York, 1985) 52-66.

[14] W.F. Ganong, *Review of Medical Physiology,* (Lange Medical Publications, Los Altos, California, 1975).

[15] P. Jackson, H. Reichgelt, F. van Harmelen, *Logic-based Knowledge Representation*, (The MIT Press, Cambridge, Massachusetts, 1989).

[16] J. Jaffar and J-L. Lassez, Constraint Logic Programming, *Proceedings of the 14th POPL*, 1987, 111-119.

[17] G. Guida and C. Tasso, Building expert systems: from life cycle to development methodology, in: G. Guida, C. Tasso (eds.), *Topics in expert systems design: methodologies and tools,* (North-Holland, Amsterdam, 1989) 3-24.

[18] R.A. Kowalski, Problems and promises of computational logic, in: J.W. Lloyd (ed.), *Computational Logic,* (Springer-Verlag, Berlin, 1990) 1-35.

[19] J.H. Kim and J. Pearl, A computational model for causal and diagnostic reasoning in inference systems, *Proceedings of the 8th International Joint Conference on Artificial Intelligence* (Karlsruhe, West Germany, 1983) 190-193.

[20] C.A. Kulikowski, S.M. Weis, Representation of expert knowledge for consultation: the CASNET and EXPERT projects, in: P. Szolovits (ed.), *Artificial Intelligence in medicine,* (Westview Press, Boulder, 1982) 21-56.

[21] G. Lanzola and M. Stefanelli, A Knowledge Acquisition Tool for Medical Diagnostic Knowledge-based Systems, in: M. Stefanelli, A. Hasman, M. Fieschi, J. Talmon (eds.), *AIME91, Proceedings of the Third Conference on Artificial Intelligence in Medicine*, (Springer-Verlag, Berlin, 1991) 272-293.

[22] J.W. Lloyd, *Foundations of Logic Programming*, 2nd edition, (Springer-Verlag, Berlin, 1987).

[23] P.J.F. Lucas, R.W. Segaar, A.R. Janssens, HEPAR: an expert system for the diagnosis of disorders of the liver and biliary tract, *Liver* **9** (1989) 266-275.

[24] P.J.F. Lucas, L.C. van der Gaag, *Principles of Expert Systems*, (Addison-Wesley, Wokingham, 1991).

[25] P.J.F. Lucas, A.R. Janssens, Development and validation of HEPAR, an expert system for the diagnosis of disorders of the liver and biliary tract, *Journal of Medical Informatics* **16** (1991) 259-270.

[26] P.J.F. Lucas, A.R. Janssens, Second evaluation of HEPAR, an expert system for the diagnosis of disorders of the liver and biliary tract, *Liver* **11** (1991) 340-346.

[27] P. Maes, D. Nardi, *Metalevel Architectures and Reflection*, (North-Holland, Amsterdam, 1987).

[28] W.W. McCune, *OTTER 2.0 Users' Guide,* Report ANN-88/44, Mathematics and Computer Science Division, Argonnne National Laboratory, Argonne, Illinois, 1990.

[29] *Artificial Intelligence in Medicine*, *Medical Expert Systems: Evaluation*, forthcoming issue, 1993.

[30] *Journal of Medical Informatics* **15** (1990), Special Issue: Validation and Testing of Medical Decision Aids.

[31] E. Motta, T. Rajan, M. Eisenstadt, A methodology and tool for knowledge acquisition in KEATS-2, in: G. Guida, C. Tasso, *Topics in expert systems design,* (North-Holland, Amsterdam, 1989) 297-322.

[32] E. Motta, T. Rajan, J. Domingue, M. Eisenstadt, Methodological foundation of KEATS, the knowledge engineer's assistant, in: B. Wielinga, J. Boose, B. Gaines, G. Schreiber, M. van Someren (eds.), *Current trends in knowledge acquisition*, (IOS Press, Amsterdam, 1990) 257-275.

[33] Peng Y, Reggia JA. *Abductive inference models for diagnostic problem solving,* (Springer-Verlag, New York, 1990).

[34] D. Poole, Normality and faults in logic-based diagnosis, *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, (Morgan-Kaufman, Los Altos, 1990) 1304-1310.

[35] R. Reiter, On closed world databases, in: H. Gallaire, J. Minker (eds.), *Logic and Databases*, (Springer-Verlag, Berlin, 1978).

[36] H. Reichgelt, The place of defaults in reasoning systems, in: B. Smith and G. Kelleher (eds.), *Reasoning Maintenance Systems and their Applications*, (Ellis Horwood, New York, 1989).

[37] J.A. Robinson, A machine-oriented logic based on the resolution principle, *Journal of the ACM,* **12** (1965) 23-41.

[38] J.A. Robinson, Automatic deduction with hyperresolution, *International Journal of Computational Mathematics,* **1** (1965) 227-234.

[39] L. Wos, D. Carson, G.A. Robinson, Efficiency and completeness of the set-of-support strategy in theorem proving, *Journal of the ACM,* **12** (1965) 536-541.

[40] L. Wos, G.A. Robinson, D. Carson, L. Shalla, The concept of demodulation in theorem proving, *Journal of the ACM,* **14** (1967) 698-709.

[41] L. Wos, R. Overbeek, E. Lusk, J. Boyle, *Automated reasoning: introduction and applications,* (Prentice-Hall, Englewood Cliffs, NJ, 1984).