Artificial Intelligence Review **17:** 1–20, 2002.

Combinations of Modal Logics *

BRANDON BENNETT[1], CLARE DIXON[2], MICHAEL FISHER[2*],
ULLRICH HUSTADT[2], ENRICO FRANCONI[3], IAN HORROCKS[3] and
MAARTEN DE RIJKE[4]
[1]*School of Computer Studies, University of Leeds, UK;* [2]*Department of Computer Science, University of Liverpool, L69 7ZF, UK;* [3]*Department of Computer Science, University of Manchester, UK;* [4]*Institute for Logic, Language and Computation, University of Amsterdam, The Netherlands (* author for correspondence, E-mail: m.fisher@csc.liv.ac.uk)*

Abstract. There is increasing use of combinations of modal logics in both foundational and applied research areas. This article provides an introduction to both the principles of such combinations and to the variety of techniques that have been developed for them. In addition, the article outlines many key research problems yet to be tackled within this callenging area of work.

Keywords: knowledge representation, logical reasoning, modal logics

1. Introduction

Combining logics for modelling purposes has become a rapidly expanding enterprise that is inspired mainly by concerns about modularity and the wish to join together different kinds of information. As any interesting real world system is a complex, composite entity, decomposing its descriptive requirements (for design, verification, or maintenance purposes) into simpler, more restricted, reasoning tasks is not only appealing but is often the only plausible way forward. It would be an exaggeration to claim that we currently have a thorough understanding of 'combined methods.' However, a core body of notions, questions and results has emerged for an important class of combined logics, and we are beginning to understand how this core theory behaves when it is applied outside this particular class.

In this paper we will consider the combination of modal (including temporal) logics, identifying leading edge research that we, and others, have carried out. Such combined systems have a wide variety of applications that

* This document was developed from a panel session in the 1999 UK Automated Reasoning Workshop, which was held as part of the AISB symposium in Edinburgh. All the authors were panelists, except for Michael Fisher who chaired the panel and edited this document.

we will describe, but also have significant problems, often concerning interactions that occur between the separate modal dimensions. However, we begin by reviewing why we might want to use modal logics at all.

2. Why Use Modal Logics?

2.1. *Motivation*

Over the past decades our perception of computers and computer programs has changed several times in quite dramatic ways, with consequences reaching far into our societies. With the rise of the personal computer we began to view the computer as an extension of our office desks and computer programs replaced traditional office tools such as typewriters, calculators, and filling cabinets with word processors, spreadsheets, and databases. Now, the advent of the electronic information age changes our view again: Computers and their programs turn into ubiquitous *digital assistants* (or digital agents). Digital agents have become necessary due to the vast extent and scattered nature of the information landscape. In addition, today's average computer user is neither able nor willing to learn how to navigate through the information landscape with the help of more traditional tools.

Digital agents have now become possible for almost the same reason. For the first time there is sufficient free information and a sufficient number of services available which can be accessed and manipulated by a computer program without direct intervention by the human computer user.

Like the personal computer, digital agents will have a substantial impact on our economy. But do they also have an impact on research in computer science? One should note that computer hardware design is still generally based on the von Neumann architecture and that computer programs are still Turing-machine equivalent. However, are techniques and results already available in computer science research that could have an impact on the way digital agents (both current and future) are developed and implemented?

Modal logics (Chellas 1980; Blackburn et al. 2001) or, more precisely, combinations of modal logics, are good candidates for a formal theory that can be helpful for the specification, development, and even the execution of digital agents. Modal logics can be used for modelling both digital agents and (aspects of) their human users. A digital agent should have an understanding of its own abilities, knowledge, and beliefs. It should also have a representation of the knowledge, beliefs, and goals of its user and of other digital agents with whom it might have to cooperate in order to achieve its goals.

Modal logics seem to be perfectly suited as a representation formalism in this setting. However, there are also some obstacles for the use of the well-studied propositional modal logics:

– propositional logic is often insufficient for more complex real world situations – a first-order, or even higher-order, language might be necessary;
– a monotonic logic might not be sufficient – in many situations our knowledge about the world is incomplete and much of our knowledge is actually only a default or only holds with a certain probability; hence, to come to useful conclusions we might have to rely on a nonmonotonic or probabilistic logic.

Thus, an appropriate representation formalism for digital agents may use combinations of (propositional or first-order) modal logics.

2.2. *Representing Agents*

Agent-based systems are a rapidly growing area of interest, in both industry and academia (Wooldridge and Jennings 1995). In particular, the characterisation of complex distributed components as *intelligent* or *rational* agents allows the system designer to analyse applications at a much higher level of abstraction. In order to reason about such agents, a number of theories of rational agency have been developed, such as the BDI (Rao and Georgeff 1991) and KARO (van Linder et al 1996) frameworks. These frameworks are usually represented as combined modal logics. In addition to their use in agent theories, where the basic representation of agency and rationality is explored, these logics form the basis for agent-based formal methods. The leading agent theories and formal methods generally share similar logical properties. In particular, the logics used have:

– an *informational* component, such as being able to represent an agent's beliefs or knowledge,
– a *dynamic* component, allowing the representation of dynamic activity, and,
– a *motivational* component, often representing the agent's desires, intentions or goals.

These aspects are typically represented as follows:

Information – modal logic of belief (**KD45**) or knowledge (**S5**);

Dynamism – temporal or dynamic logic;

Motivation – modal logic of intention (**KD**) or desire (**KD**).

Thus, the predominant approaches use relevant combinations. For example: Moore (1980) combines propositional dynamic logic and a modal logic of knowledge (**S5**); the BDI framework (Rao and Georgeff 1991; Rao 1995) uses linear or branching temporal logic, together with modal logics of belief (**KD45**), desire (**KD**), and intention (**KD**); Halpern et al. (1989; Fagin et al. 1996) use linear and branching-time temporal logics combined with a multi-modal (**S5**) logic of knowledge; and the KARO framework (van Linder et al. 1996; van der Hoek et al. 1997) uses propositional dynamic logic, together with modal logics of belief (**KD45**) or knowledge (**S5**) and wishes (**KD**).

If we assume that combinations of modal logics play an important part in modelling digital agents, it is an obvious step to consider the question whether we are able to verify specified requirements or properties of an agent using formal methods. Unfortunately, many of these combinations, particularly those using dynamic logic, become too complex (not only undecidable, but incomplete) to use in practical situations. Thus, much current research activity concerning agent theories centres around developing simpler combinations of logics that can express many of the same properties as the more complex combinations, yet are simpler to mechanise. For example, some of our work in this area has involved developing a simpler logical basis for BDI-like agents (Fisher 1997b).

2.3. *Spatial Logics*

While the traditional uses of modal logics are for representing interacting propositional attitudes such as belief, knowledge, intention, etc., recent work has investigated the representation of spatial information in combined modal logics. For example, in (Bennett 1996, 1997), Bennett uses the topological interpretation of the **S4** modality as an *interior* operator, in combination with an **S5** modality in order to encode a large class of topological relations.

2.4. *Description Logics*

Although not originally characterised in this way, one of the most successful uses of combinations of modal logics has been the development of expressive Description Logics (Sattler 1996; De Giacomo and Massacci 1996; Horrocks 1998b). Description logics have found many practical applications, for example in reasoning about database schemata and queries (Calvanese et al. 1998a, b). Since description logics have been shown to correspond directly to certain combinations of modal logics, such combinations are also useful (Schild 1991; Areces and de Rijke 2000).

The application concerning schema and query reasoning that was described above is very promising, as are ontological engineering applications (Rector et al. 1997; Baker et al. 1998). In this, and other, contexts description logics that combine transitive, non-transitive and inverse roles (Horrocks and Sattler 2000) have proved particularly useful as they enable many common conceptual data modelling formalisms (including Entity-Relationship models) to be captured while still allowing for tractable implementations (Horrocks 2000).

Another successful combination of modal logics within a description logic framework is motivated by the attempt to add a temporal dimension to the knowledge representation language (Artale and Franconi 2000, 2001). Typical applications of such temporally extended description logics have been the representation of actions and plans in Artificial Intelligence (Artale and Franconi 1998, 1999a), and reasoning with conceptual models of (federated) temporal databases (Artale and Franconi 1999b). A temporal description logic can be obtained by combining the description logic with a standard point-based tense logic (Schild 1993; Wolter and Zakharyaschev 1999, 2000; Wolter et al. 2001) or with a variant of the HS interval-based propositional temporal logic (Halpern and Shoham 1991).

Given that combinations of modal logics have a number of real and potential uses, it is important to remember the general technical problems that can occur with such combinations; this we do in the next section.

3. Problems with Combinations

Let $\mathbf{L}_1$ and $\mathbf{L}_2$ be two logics – typically, these are special purpose logics with limited expressive power, as it often does not make sense to put together logics with universal expressive power. Let P be a property that logics may have, say decidability, or axiomatic completeness. The *transfer problem* is this: if $\mathbf{L}_1$ and $\mathbf{L}_2$ enjoy the property P, does their combination $\mathbf{L}_1 \oplus \mathbf{L}_2$ have P as well? Transfer problems belong to the main mathematical questions that logicians have been concerned with in the area of combining logics.

When, and for which properties, do we have transfer or failure of transfer? As a rule of thumb, in the absence of interaction between the component logics, we do have transfer; here, absence of interaction means that the component languages do not share any symbols, except maybe the booleans and atomic symbols (we will say more about interactions in Section 5). Properties that do transfer in this restricted case include the finite model property, decidability, and (under suitable restrictions on the classes of models and the complexity class) complexity upper bounds.

The positive proofs in the area are usually based on two key intuitions: *divide and conquer* and *hide and unpack*. That is: try to split problems and delegate sub-problems to the component logics; and when working inside one of the component logics view information relating to other component logics as alien information and 'hide' it – don't unpack the hidden information until we have reduced a given problem to a sub-problem in the relevant component logic. Neither of these key intuitions continues to work in the presence of interaction. For instance, consider two modal languages $\mathcal{L}_1$ and $\mathcal{L}_2$ with modal operators $\Box$ and $\blacksquare$, respectively; there are logics $\mathbf{L}_1$ and $\mathbf{L}_2$ in $\mathcal{L}_1$ and $\mathcal{L}_2$ whose satisfiability problem is in NP, while the satisfiability problem for the combined language *plus* the interaction principle $\Box p \rightarrow \blacksquare p$ is undecidable (Hemaspaandra 1994).

4. Reasoning Methods

So, we have seen that combinations of modal logics are, or at least have the potential to be, very useful. However, we must also be careful in combining such logics. In all the application areas discussed in Section 2, the notion of proof is important. In agent theories, for example, a proof allows us to examine properties of the overall theory and, in some cases, to characterise computation within that theory. In agent-based formal methods, a proof is clearly important in developing verification techniques.

But how do we go about developing proof techniques for combined logics? Since there are a wide variety of reasoning systems for individual modal logics, it is natural to ask: "does combining work for actual reasoning systems?", i.e., can existing tools for each component logic be put together to obtain tools for combined logics? Obviously, the *re-use* of tools and procedures is one of the key motivations underlying the field.

Unfortunately, one cannot put together proof procedures for two logics in a uniform way. First, 'proving' can have different meanings in different logics: (semi-)deciding satisfiability or validity, computing an instantiation, or generating a model. Second, it is not clear where to "plug in" the proof procedure for a logic $\mathbf{L}_1$ into that for a second logic $\mathbf{L}_2$; a proof procedure may have different notions of valuations, or of proof goals.

So what can one do? One way out is to impose special conditions on the calculi that one wants to combine (Beckert and Gabbay 1998). Another possibility, in the case of modal logics, is to use a translation-based approach to theorem proving, by mapping all component logics into a common background logic (see below).

There are quite a few successful particular instances of combined logics where we have no problems whatsoever in putting together tools; see Aiello et al. (1999), kurtonina and de Rijke (2000). By and large, however, we don't have a good understanding of how to proceed. Further experiments are needed, both locally, and network based, so that at some stage we will be able to plug together tools without having to be the designer or engineer of the systems.

In the following sections we will consider the work that we are involved with concerning the development of proof techniques for combined logics.

4.1. *Tableaux Based Reasoning*

Most commonly, reasoning methods for modal logics are presented as tableau-style procedures. These semantically-based approaches are well suited to implementing proof procedures for interacting modalities because the usually explicit presence of models in the data-structures used by the algorithm helps the system to represent the interactions between modalities. Thus, tableaux systems have the advantages that:

- they have a (fairly) direct and intuitively obvious relationship with the Kripke structures of the underlying logic;
- algorithms are easy to design and extend;
- the simplicity of the algorithms facilitates optimised implementation.

We have developed a range of tableaux-based systems for description logics, for example (Horrocks 1998a, 2000), and for combinations of linear-time temporal logic with modal logics **S5** or **KD45** (Wooldridge et al. 1998).

4.2. *Resolution Based Reasoning*

An alternative approach to reasoning in combined modal logics is to use direct resolution techniques which should, in the long term have at least the performance of corresponding tableaux-based systems.

Our work in this area has focused on extending the resolution methods developed for linear-time temporal logics (Fisher 1991; Fisher et al. 2001) to particular combinations of the logics considered above. This clausal resolution method centres round three main steps:

- translation to a simple normal form (involving renaming of complex subformulae and reduction to a core set of operators);
- classical resolution between formulae that occur at the same moment in time; and
- resolution between sets of formula that make ϕ always true with constraints that ensure ϕ is false at some point in the future.

In Dixon et al. (1998), we extended this method to linear-time temporal logic combined with **S5** modal logic. During translation to the normal form, temporal formulae are separated out from modal formulae (using renaming). Reasoning in the temporal and modal components is then carried out separately and information is transferred between the two components via classical propositional logic.

Other important direct resolution methods for modal logics Include (Mints 1990) (on which the above approach was based) and our work on prefixed resolution (areces et al. 1999).

4.3. *Translation Based Reasoning*

Translation between different formal languages and different problem classes is one of the most fundamental principles in computer science. For example, a compiler for a programming language is just (the implementation of) a translation from one formal language into another formal language. In the case of logical reasoning, such a translation approach is based on the idea that given a (sound, complete, and possibly terminating) theorem prover for a logic $\mathbf{L}_2$, inference in a logic $\mathbf{L}_1$ can be carried out by translating formulae of $\mathbf{L}_1$ into $\mathbf{L}_2$. There are minimal requirements imposed on the translation from $\mathbf{L}_1$ into $\mathbf{L}_2$, namely that the translation preserves satisfiability equivalence and that it can be computed in polynomial time.

In the case of modal logics, the most straightforward translation mapping, the *relational translation*, is based on the Kripke semantics of modal logics (can Benthem 1976). But just as there are many compilers for a single programming language, there are a number of alternative translation mappings from modal logics into subclasses of first order logic which satisfy the mentioned minimal requirements. These include the *functional translation* (Auffray and Enjalbert 1992; Fariñas del Cerro and Herzig 1988; Ohlbach 1991) the *optimised functional translation* (Ohlbach and Schmidt 1997; Schmidt 1997), and the *semi-functional translation* (Nonnengart 1995).

The advantages of the translation approach include the following. First, by enabling the use of a variety of first-order and propositional theorem provers the translation approach provides access to efficient, reliable and general methods of modal theorem proving. Second, decision procedures can be obtained by suitably instantiating parameters in existing theorem proving frameworks (Bachmair and Ganzinger 1997). For the optimised functional translation this has been demonstrated in (Schmidt 1997, 1999), while the relational and semi-functional translation have been considered in (Ganzinger et al. 1999a; Hustadt 1999; Hustadat and Schmidt 1998, 1999b). Third, there are *general guidelines* for choosing these parameters in the right way to enforce termination and the hard part is to prove that termination is guaranteed.

The most common approach uses *ordering refinements* of resolution to ensure termination. Ordering refinements are very natural as they provide decision procedures for a wide range of solvable first-order fragments (Fermüller et al. 2000; Hustadt and Schmidt 1999a).

An interesting alternative is *selection refinements* (Hustadt and Schmidt 1998). Selection refinements are closely related to hyper-resolution, a well-known and commonly used refinement in first-order theorem provers. It can be shown that selection refinements plus splitting are able to polynomially simulate proof search in standard tableaux calculi for modal logic (Hustadt and Schmidt 1999b, 2000). Another alternative is based on the fact that the optimised functional translation can be used to translate modal formulae into a subclass of the Bernays-Schönfinkel class (Schmidt 1997). The number of ground instances of clauses obtained from formulae of the Bernays-Schönfinkel class is always finite. Thus, it is possible to use propositional decision procedures to test the satisfiability of the sets of ground clauses in this way.

While there are a number of other promising approaches, for example the use of equational methods (e.g. rewriting), based on the algebraic interpretation of modal logics, or constraint satisfaction techniques, it seems likely that many of the techniques developed will share a broad similarity (Hustadt and Schmidt 2000). In this sense, there are few reasoning methods that are "inappropriate" (Ohlbach et al. 2000).

5. Interactions

Once we combine modal (and temporal) logics, we must decide whether we are going to support *interactions* between the modal dimensions. Although there may be some cases of combined modal logics without interactions that can be useful, to exploit the full power of this combination technique, interactions must be handled. For example, interactions typically involve *commutative* modalities – i.e., pairs of modalities, $\Box$ and $\blacksquare$ satisfying the schema

$$\Box\blacksquare\phi \leftrightarrow \blacksquare\Box\phi \ .$$

But this apparently simple schema is surprisingly hard to incorporate into modal reasoning algorithms. Indeed, Kracht (1995) has shown that the logic containing three **S5** modalities, each pair of which commutes, is undecidable, while it is well-known that **S5** itself is NP-complete. See Section 8 later for further discussion of decidability issues.

In the following, we will re-examine some of the approaches considered in Section 4, particularly concerning how they cope with such interactions.

5.1. *Description Logics*

In the context of description logics, interactions between different kinds of role (modality) are an inherent and essential part of the work on expressive description logics. We can think of basic description logics as a syntactic variant of the normal multi-modal logic **K**, or even of a propositional dynamic logic. More expressive description logics are obtained by adding new roles with specific properties – for example they may be transitive or functional (deterministic); in this case, there is no interaction between modalities in the combined logic. However, the interesting cases are when converse modalities, or implications between modalities, e.g., $\Box\phi \rightarrow \blacksquare\phi$, are introduced. The latter axiom schema in a basic description logic combined with multi-modal **S4** is important since it allows us to encode the universal modality. As there are no description logics in which interactions such as $\Box\blacksquare\phi \leftrightarrow \blacksquare\Box\phi$ are required, the above approach leads to practical reasoning systems for quite complex description logics.

In the context of temporal description logics, interactions between temporal and non-temporal modalities are usually bad. For example, the ability to define a *global* role – i.e., invariant over time – makes the logic undecidable. However, we have identified a special case, still very useful in practice, where the combination of a basic description logic with an interval based temporal component is decidable (Artale and Franconi 1998).

5.2. *Temporal Logics of Knowledge*

Particular interactions between temporal logics and modal logics of knowledge (**S5**) have been analysed in (Halpern and Vardi 19886, 1988a, b, 1989). Notions such as *perfect recall*, *no learning*, *unique initial state* and *synchrony* are defined. The basic temporal logics of knowledge are then restricted to those where certain of the above notions hold. Halpern et al. (1989) consider the complexity of the validity problem of these logics for one or more agents, linear or branching-time temporal logic and with or without common knowledge. In general, the complexity of the validity problem is higher where interactions are involved, with some combinations of interactions leading to undecidability.

In Dixon and Fisher (2000), we consider resolution systems with synchrony and perfect recall by adding extra clauses to the clause-set to account for the application of the synchrony and perfect recall axiom to particular clauses. The former can be axiomatised by the axioms of linear-time temporal logic, plus the axioms of the modal logic **S5** (represented by the modal operator 'K') with the additional interaction axiom (Halpern and Vardi 1988b; Halpern et al. 2000),

$$K\bigcirc\phi \rightarrow \bigcirc K\phi,$$

meaning informally that if an agent knows that in the next moment ϕ will hold then in the next moment the agent will know that ϕ holds. Essentially, in systems with perfect recall, the number of timelines (or possible futures) that an agent considers possible stays the same or decreases over time. The axiom for synchrony and no learning is the converse of the above axiom.

We are also interested in looking at what interactions are actually used in application areas. For example the interaction (where $[do_i(\alpha)]$ is effectively a dynamic logic operator)

$$K_i[do_i(\alpha)]\phi \rightarrow [do_i(\alpha)]K_i\phi$$

meaning informally if agent i knows that doing action α results in ϕ then agent i doing the action α results in agent i knowing ϕ, has been highlighted as desirable for the KARO framework (van der Hoek et al. 1997). This is very similar to the synchrony and perfect recall axiom noted above.

5.3. *Translation*

From the view of the translation approach, interactions between logics in some combination of modal logics are no different from the more traditional axiom schemata which are usually provided to characterise extensions of the basic modal logic **K**, for example, the modal logics **KD**, **S4** and **S5**. To accommodate these axiom schemata in the translation approach we have to find a satisfiability or equivalence preserving first-order characterisation for them.

For example, consider a combination of two basic modal logics with modal operators $\Box$ and $\blacksquare$. A very simple interaction between the two modal logics is given by the axiom schema

$$\Box p \rightarrow \blacksquare p. \tag{1}$$

Using the relational translation approach with predicate symbols $R_\Box$ and $R_\blacksquare$ corresponding to the accessibility relation for each logic, axiom schema (1) can be characterised by the first-order formula

$$\forall xy\colon R_\Box(x, y) \rightarrow R_\blacksquare(x, y). \tag{2}$$

In some cases, the first-order formulae characterising interactions are already covered by existing decidability results for subclasses of first-order logic corresponding to the modal logics. For example, formula (2) belongs to the Skolem class, the class of DL-clauses (Fermüller et al. 2000), and various other classes. In these cases the translation approach provides us with sound, complete, and terminating decision procedures for combinations of interacting modal logics without any additional effort.

It is even possible to obtain general characterisations of the boundaries of decidability of combinations of interacting modal logics in this way. For example, results from (Ganzinger et al. 1999b) imply that if we have two modal logics which satisfy the **4** axiom, that is the accessibility relations are transitive, and the modal logics are interacting, for example, by the axiom schema (1), then the combined logic is undecidable.

6. General Frameworks

The considerations of previous sections were intended to highlight some of the potentials of a representational framework based on combinations of modal logics. However, it was also pointed out that providing such a representational framework together with the accompanying tools is a non-trivial problem. It is therefore very likely that whatever our first approach to this problem might be, it will have shortcomings which are too serious to provide a workable solution. So, research in this direction will proceed by considering a number of combinations of modal logics and assessing both their appropriateness and usefulness.

Whether or not a particular combination of modal logics provides a suitable foundation for a representational framework can only be decided if we are able to make practical use of it, for example for the purpose of verifying and executing agents. Therefore, the assessment necessarily requires the availability of implemented theorem provers for combinations of modal logics.

It is, of course, possible to develop a suitable calculus from scratch, as we described in Section 4, proving its soundness, completeness, and possibly its termination for the combination of logics under consideration, and finally to implement the calculus accompanied with the required data structures, heuristic functions, and optimisation techniques in a theorem prover. However, bearing in mind that we have to make this effort for a number of different combinations of modal logics, it is necessary to find a more general approach. Thus, we are interested in general principles and a general framework for combining modal logics.

One approach is to use standard methods for modal logics. There are various levels at which one could give a generalised account of combinations of modal logics. Semantically, both the Kripke and Tarski style semantics generalise easily to combined modal systems. Although algebraic semantics is more general and supports equational and constraint-based methods, Kripke semantics is better known, more intuitive, and is well-suited to developing tableau methods. The correspondences between these two semantic approaches provide different perspectives on the interpretation, each of which has its own advantages; see Blackburn et al. (2001) for details on the correspondences.

The proof theory of modal logics was originally developed in terms of axiom schemata (Hilbert systems). This approach can be applied equally well to combined modal systems. Axiomatic presentations are concise and the meanings of the axioms are (if not too complex) usually readily understandable. Within such a system, proofs can be carried out simply by substituting formulae into the axiom schemata and applying the *modus ponens* rule. However, these non-analytic rules do not provide practical inference systems. Rule-based presentations seem to be better suited for the development of inference algorithms, especially where they are purely analytic.

While the use of a traditional approach might be helpful in certain cases, there is clearly a need for a framework specifically designed to handle the combination of modal logics. Examples for such frameworks are fibring (Gabbay 1999), the translation approach (see above), and the SNF approach (Dixon et al 1998). Briefly, the SNF approach involves extending the normal form (called SNF) developed for representing temporal logics (Fisher 1997a) to other temporal logics, such as branching-time temporal logics (Bolotov and Fisher 1999), and modal logics (Dixon et al. 1998). The basic approach is to keep rules (clauses) separate that deal with different logics, re-use proof rules for the relevant logics and to make sure enough information is passed between each part (Dixon et al. 1998; Fisher and Ghidini 1999).

An approach that we have been interested in more recently is itself a combination, namely, a combination of the SNF and the translation approaches. For example, when combining a temporal logic with a modal logic, the temporal aspects remain as SNF clauses, while the modal clauses are translated to classical logic (Hustadt et al. 2000).

7. Tools

Given that combinations of modal logics are often quite complex, what are the prospects for having practical tools that will allow us to reason about combined modal logics incorporating interactions? One answer is that, for some particular instances of combined modal logics, namely description logics, powerful and efficient systems already exist. Good examples of these are FaCT (Horrocks 1998a), iFaCT (Horrocks 2000) and DLP (Patel-Schneider 1998).

Implementation effort is also under way to support some of the reasoning methods described in Section 4, for example clausal resolution based upon the SNF approach. Here, a resolution based theorem prover for linear-time temporal logics plus **S5** modal logic based on (Dixon et al. 1998) is being developed. This is to be extended with work on strategies for efficient application, and guidance, of proof rules (Dixon 1996, 1998, 1999) developed for temporal logics in order to help deal with interactions.

However, if combinations of modal logics are to function as practical vehicles for reasoning, we are likely to have to face the issues of complexity. There are various complexity results counting both in favour and against the feasibility of combined modal reasoning. As we have seen earlier, Kracht (1995) (amongst others) has shown that some simple combinations of modalities together with very simple interaction axioms yield undecidable systems; on the positive side, there are a number of examples of quite expressive fragments of multi-modal languages, whose decision procedures are polynomial, for example (Renz and Nebel 1997). Thus, the viability of reasoning with combined modal logics depends very much on the particular combination of modalities and interaction axioms.

An obvious way of reducing the complexity of a logical language is to restrict its syntax. We consider such an approach, called *layering* (Finger and Gabbay 1992), below.

7.1. *Layered Modal Logics*

A layered modal logic is a special kind of combined modal logic in which restrictions are placed on the nesting order of the different modalities in the language. A typical layering restriction would be to require that temporal modalities lie outside the scope of spatial modalities. This would allow one to represent tensed spatial constraints, but not specify temporal constraints which vary over space. Whereas tensed spatial relations are very common in natural language, the latter kind of constraint is not so common.

7.2. *Expressive Power of Restricted Formalisms*

One might expect that restricting combined modal formalisms to be layered would reduce their expressive power to the extent that they would lose much of their usefulness. However, the **S5**(**S4**) hybrid described in Section 2.3 is strictly layered (which is one way to account for its nice computational properties) and is also capable of representing a wide range of topological relations.

In the case of a spatio-temporal language (as above) one might restrict temporal operators to lie outside the scope of spatial operators. This would result in an intuitively natural language of tensed spatial constraints, in which we could, for example, make statements of the form "x will overlap y" but not of the form "every part of x will at some time satisfy property ϕ."

8. Decidability Problems

The result by Ganzinger et al. (1999b) and related results by Kracht (1995) show that undecidability of combinations of modal logics is imminent as soon as we allow interactions between the logics. Consequently, one of the most attractive features of modal logics is lost in these cases. Furthermore, one of the important arguments for the effort invested in the development of tableaux-based theorem provers for modal logics is lost, namely, that they provide rather straightforward decision procedures while the translation approach does not ensure termination without the use of the appropriate refinements of resolution.

This raises several questions:

1. Should we concentrate our research on identifying combinations of interacting modal logics which are still decidable? Or should we abandon the consideration of these rather simple logics, since we have to expect that most combinations of modal logic interesting for real-world applications will be undecidable?
2. If we acknowledge that we have to deal with undecidable combinations of modal logics, does the use of modal logics still make sense? Is there still an advantage compared to the use of first-order logic or some other logic for the tasks under consideration?

Note that the translation approach partly diminishes the importance of these questions as far as the development of theorem provers is concerned. Refinements of resolution which ensure termination on the translation of modal formulae are still sound and complete calculi for full first-order logic. Thus, in the translation approach we can always fall back on full first-order logic whenever we have the feeling that combinations of modal logics are too much of a strait-jacket.

In the work on description logics, the practical use of these formalisms has meant that retaining decidability is essential. Thus, one of the key aspects of research in this area has been the extension of expressive power while (just) retaining decidability.

In contrast, in the work on agent theories, there is research attempting to characterise which interactions do lead to undecidability. For decidable cases, it is important to assess how useful the interactions are in relevant application areas; if they are of use we can study decision procedures for these interactions and one of the key strategies here has been to structure information, and to separate the various reasoning tasks.

Finally, the development of heuristics and strategies to guide the proofs is essential regardless of decidability as, even if decidable fragments exist, their complexity is likely to be high if we include interactions.

9. Summary

Does the idea of combining logics actually offer anything new? Some of the possible objections can be justified. Logical combination is a relatively new idea: it has not yet been systematically explored, and there is no established body of results or techniques. Nonetheless, there is a growing body of logic-oriented work in the field, and there are explorations of their uses in AI, computational linguistics, automated deduction, and computer science. An overly critical reaction seems misguided.

In order to receive more attention from the wider community of researchers interested in knowledge representation and reasoning, the capabilities of combined modal logics need to become more accessible; and their superiority over other formalisms (such as the direct use of first-order logic) needs to be decisively demonstrated for some significant applications. A survey of the expressive power, computational properties and potential applications of a large class of combined modal logics would be very useful to AI system designers.

References

Aiello, M., C. Areces & M. de Rijke (1999). Spatial Reasoning for Image Retrieval. In *Proceedings of the International Workshop on Description Logics (DL'99)*. Sweden: Linköpings Universitet, 23–27.

Areces, C., H. de Nivelle & M. de Rijke (1999). Prefixed Resolution: A Resolution Method for Modal and Description Logics. In *Proceedings of the 16th International Conference on Automated Deduction (CADE-16)*, volume 1632 in Lecture Notes in Artificial Intelligence. Springer, 187–201.

Areces, C. and M. de Rijke (2000). Description and/or Hybrid Logics. In *Proceedings of AiML-2000 Workshop*, 1–14.

Artale, A. and E. Franconi (1998). A Temporal Description Logic for Reasoning about Actions and Plans. *Journal of Artificial Intelligence Research* **9**: 463–506.

Artale, A. and E. Franconi (1999). Representing a Robotic Domain using Temporal Description Logics. *Journal of Artificial Intelligence for Engineering Design, Analysis and Manufacturing (AIEDAM)* **2**(13): 105–117.

Artale, A. and E. Franconi (1999). Temporal Entity-Relationship Modeling with Description Logics. In *Proceedings of the 18th International Conference on Conceptual Modeling (ER'99)*, volume 1728 in Lecture Notes in Computer Science. Springer, 81–95.

Artale, A. and E. Franconi (to appear). A Survey of Temporal Extensions of Description Logics. *Annals of Mathematics and Artificial Intelligence (AMAI)*. Kluwer Academic Publishers.

Artale, A. and E. Franconi (forthcoming). Temporal Description Logics. In *Handbook of Time and Temporal Reasoning in Artificial Intelligence*.

Auffray, Y. and P. Enjalbert (1992). Modal Theorem Proving: An Equational Viewpoint. *Journal of Logic and Computation*, 2(3): 247–297.

Bachmair, L. and H. Ganzinger (to appear). A Theory of Resolution. In J.A. Robinson and A. Voronkov (eds.), *Handbook of Automated Reasoning*. MIT Press.

Baker, P. G., A. Brass, S. Bechhofer, C. Goble, N. Paton and R. Stevens (1998). Tambis: Transparent Access to Multiple Bioinformatics Information Sources: An Overview. In *Proceedings of the 6th International Conference on Intelligent Systems for Molecular Biology (ISMB'98)*.

Beckert, B. and D. Gabbay (1998). Fibring Semantic Tableaux. In *Proceedings of the International Conference on Automated Reasoning with Analytic Tableaux and Related Methods (TABLEAUX'98)*, volume 1397 in Lecture Notes in Artificial Intelligence, Springer, 77–92.

Bennett, B. (1996). Modal logics for qualitative spatial reasoning. *Bulletin of the Interest Group in Pure and Applied Logic (IGPL)* **4**(1): 23–45. `ftp://ftp.mpi-sb.mpg.de/pub/igpl/Journal/V4-1/index.html`.

Bennett, B. (1997). *Logical Representations for Automated Reasoning about Spatial Relationships*. PhD thesis, School of Computer Studies, The University of Leeds. Abstract and postscript at `http://www.scs.leeds.ac.uk/brandon/thesis.html`.

Blackburn, P., M. de Rijke and Y. Venema (2001). *Modal Logic*. Cambridge University Press (See `http://www.mlbook.org`.)

Bolotov, A. and M. Fisher (1999). A Resolution Method for CTL Branching-Time Temporal Logic. *Journal of Theoretical and Experimental Artificial Intelligence* **11**: 77–93.

Calvanese, D., G. De Giacomo and M. Lenzerini (1998). On the decidability of query containment under constraints. In *Proceedings of the 17th ACM SIGACT SIGMOD SIGART Symposium on Principles of Database Systems (PODS-98)*. ACM Press, 149–158.

Calvanese, D., G. De Giacomo, M. Lenzerini, D. Nardi and R. Rosati (1998). Description Logic Framework for Information Integration. In *Proceedings of the 6th International Conference on the Principles of Knowledge Representation and Reasoning (KR-98)*. Morgan Kaufmann, 2–13.

Chellas, B. (1980). *Modal Logic: An Introduction.* Cambridge University Press.

De Giacomo, G. and F. Massacci (1996). Tableaux and Algorithms for Propositional Dynamic Logic with Converse. In *Proceedings of the 13th International Conference on Automated Deduction (CADE-13)*, volume 1104 in Lecture Notes in Artificial Intelligence, Springer, 613–628.

Dixon, C. and M. Fisher (2000). Clausal Resolution for Logics of Time and Knowledge with Synchrony and Perfect Recall. In *Proceedings of ICTL 2000*. Leipzig, Germany, October 2000.

Dixon, C., M. Fisher and M. Wooldridge (1998). Resolution for Temporal Logics of Knowledge. *Journal of Logic and Computation* **8**(3): 345–372.

Dixon, C. (1996). Search Strategies for Resolution in Temporal Logics. In *Proceedings of the 13th International Conference on Automated Deduction (CADE-13)*, volume 1104 of *Lecture Notes in Artificial Intelligence*. Springer, 672–687.

Dixon, C. (1998). Temporal Resolution using a Breadth-First Search Algorithm. *Annals of Mathematics and Artificial Intelligence* **22**: 87–115.

Dixon, C. (1999). Removing Irrelevant Information in Temporal Resolution Proofs. *Journal of Experimental and Theoretical Artificial Intelligence* **11**: 95–121.

Fariñas del Cerro, L. and A. Herzig (1988). Quantified modal logic and unification theory. Technical Report LSI 293, University Paul Sabatier.

Fermüller, C., A. Leitsch, U. Hustadt and T. Tammet (to appear). Resolution decision procedures. In J. A. Robinson and A. Voronkov (eds.), *Handbook of Automated Reasoning*. MIT Press.

Finger, M. and D. M. Gabbay (1992). Adding a Temporal Dimension to a Logic System. *Journal of Logic, Language, and Information* **1**: 203–234.

Fisher, M., C. Dixon and M. Peim (2001). Clausal Temporal Resolution. *ACM Transactions on Computational Logic* **2**(1).

Fisher, M. (1991). A Resolution Method for Temporal Logic. In *Proceedings of the 12th International Joint Conference on Artificial Intelligence (IJCAI'91)*. Morgan Kaufman, 99–104.

Fisher, M. (1997). A Normal Form for Temporal Logic and its Application in Theorem-Proving and Execution. *Journal of Logic and Computation* **7**(4): 429–456.

Fisher, M. (1997). Implementing BDI-like Systems by Direct Execution. In *Proceedings of the 15th International joint Conference on Artificial Intelligence (IJCAI'97)*. Morgan Kaufmann, 316–321.

Fisher, M. and C. Ghidini (1999). Programming Resource-Bounded Deliberative Agents. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI'99)*. Morgan Kaufmann, 200–205.

Fagin, R., J. Halpern, Y. Moses and M. Vardi (1996). *Reasoning About Knowledge*. MIT Press.

Gabbay, D. M. (1999). *Fibring Logics*, *Oxford Logic Guides* 38. Oxford University Press.

Ganzinger, H., U. Hustadt, C. Meyer and R. A. Schmidt (to appear). A Resolution-Based Decision Procedure for Extensions Of K4. In *Advances in Modal Logic 2*. CSLI Publications.

Ganzinger, H., C. Meyer and M. Veanes (1999). The Two-Variable Guarded Fragment with Transitive Relations. In *Proceedings of the 14th Annual IEEE Symposium on Logic in Computer Science (LICS'99)*. IEEE Computer Society Press, 24–34.

Halpern, J. Y. and M. Y. Vardi (1986). The Complexity of Reasoning about Knowledge and Time: Extended Abstract. In *Proceedings of the 18th Annual ACM Symposium on Theory of Computing*. ACM Press, 304–315.

Halpern, J. Y. and M. Y. Vardi (1988). Reasoning about Knowledge and Time in Asynchronous Systems. In *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*. ACM Press, 53–65.

Halpern, J. Y. and M. Y. Vardi (1988). The Complexity of Reasoning about Knowledge and Time: Synchronous Systems. Technical Report RJ 6097, IBM Almaden Research Center.

Halpern, J. Y. and M. Y. Vardi (1989). The Complexity of Reasoning about Knowledge and Time. I Lower Bounds. *Journal of Computer and System Sciences* **38**: 195–237.

Halpern, J. Y., R. van der Meyden and M. Vardi (submitted). Complete axiomatizations for reasoning about knowledge and time.

Halpern, J. Y. and Y. Shoham (1991). A Propositional Modal Logic of Time Intervals. *Journal of ACM* **38**(4): 935–962.

Hemaspaandra, E. (1994). Complexity Transfer for Modal Logic. In *Proceedings of the 9th Annual IEEE Symposium on Logic in Computer Science (LICS'94)*. IEEE Computer Society Press, 164–175.

Horrocks, I. (1998). The FaCT system. In *Proceedings of the International Conference on Automated Reasoning with Analytic Tableaux and Related Methods (TABLEAUX'98)*, volume 1397 in Lecture Notes in Artificial Intelligence. Springer, 307–312.

Horrocks, I. (1998). Using an Expressive Description Logic: Fact or Fiction? In *Proceedings of the 6th International Conference on Principles of Knowledge Representation and Reasoning (KR'98)*. Morgan Kaufmann, 636–647.

Horrocks, I. (1999). FaCT and iFaCT. In, P. Lambrix, A. Borgida, M. Lenzerini, R. Möller and P. Patel-Schneider (eds.), *Proceedings of International Description*

Logics Workshop (DL'99). Linköpings Universitet, Sweden, July 1999, 133–135. (See http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-22/)

Horrocks, I. and U. Sattler (1999). A Description Logic with Transitive and Inverse Roles and Role Hierarchies. *Journal of Logic and Computation* **9**(3): 385–410.

Hustadt, U., C. Dixon, R. A. Schmidt and M. Fisher (2000). Normal Forms and Proofs in Combined Modal and Temporal Logics. In *Proceedings of the Third International Workshop on Frontiers of Combining Systems (FroCoS'2000)*, volume 1794 in Lecture Notes in Artificial Intelligence. Springer, 73–87.

Hustadt, U. and R. A. Schmidt (2000). Issues of Decidability for Description Logics in the Framework of Resolution. In *Automated Deduction in Classical and Non-Classical Logics: Selected Papers*, volume 1761 in Lecture Notes in Artificial Intelligence. Springer.

Hustadt, U. and R. A. Schmidt (2000). Using Resolution for Testing Modal Satisfiability and Building Models. To appear in the *SAT 2000* Special Issue of *Journal of Automated Reasoning*.

Hustadt, U. (1999). *Resolution-Based Decision Procedures for Subclasses of First-Order Logic*. PhD thesis, Universität des Saarlandes, Saarbrücken, Germany.

Hustadt, U. and R. A. Schmidt (1999). Maslov's Class K Revisited. In *Proceedings of the 16th International Conference on Automated Deduction (CADE-16)*, volume 1632 in Lecture Notes in Artificial Intelligence. Springer, 172–186.

Hustadt, U. and R. A. Schmidt (1999). On the Relation of Resolution and Tableaux Proof Systems for Description Logics. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI'99)*. Morgan Kaufmann, 110–115.

Kurtonina, N. and M. de Rijke (in preparation). Hybrid Logics and Linguistic Inference.

Kracht, M. (1995). Highway to the Danger Zone. *Journal of Logic and Computation* **5**(1): 93–109.

Kracht, M. and F. Wolter (1997). Simulation and Transfer Results in Modal Logic. *Studia Logica* **59**: 149–177.

van Linder, B., W. van der Hoek and J. J. Ch. Meyer (1996). Formalising Motivational Attitudes of Agents: On Preferences, Goals and Commitments. In *Intelligent Agents II*, volume 1037 in Lecture Notes in Artificial Intelligence. Springer, 17–32.

Mints, G. (1990). Gentzen-Type Systems and Resolution Rules, Part I: Propositional Logic. In *Proceedings of the International Conference on Computer Logic*, volume 417 in Lecture Notes in Computer Science. Springer, 198–231.

Moore, R. C. (1990). A Formal Theory of Knowledge and Action. In *Readings in Planning*. Morgan-Kaufmann.

Nonnengart, A. (1995). *A Resolution-Based Calculus For Temporal Logics*. PhD thesis, Universität des Saarlandes, Saarbrücken, Germany.

Ohlbach, H. J. (1991). Semantics-Based Translation Methods for Modal Logics. *Journal of Logic and Computation* **1**(5): 691–746.

Ohlbach, H. J., A. Nonnengart, M. de Rijke and D. M. Gabbay (to appear). Encoding two-valued non-classical logics in classical logic. In J. A. Robinson and A. Voronkov (eds.), *Handbook of Automated Reasoning*. MIT Press.

Ohlbach, H. J. and R. A. Schmidt (1997). Functional Translation and Second-Order Frame Properties of Modal Logics. *Journal of Logic and Computation* **7**(5): 581–603.

Patel-Schneider, P. F. (1998). DLP system Description. In *Collected Papers from the International Description Logics Workshop (DL'98)*. CEUR, 87–89.

Rao, A. S. and M. P. Georgeff (1991). Modeling Agents within a BDI-Architecture. In *Proceedings of the 2nd International Conference on Principles of Knowledge Representation and Reasoning*. Morgan Kaufmann, 473–484.

Rao, A. S. (1996). Decision Procedures for Propositional Linear-Time Belief-Desire-Intention Logics. In *Intelligent Agents II*, volume 1037 in Lecture Notes in Artificial Intelligence. Springer, 33–48.

Rector, A., S. Bechhofer, C. A. Goble, I. Horrocks, W. A. Nowlan and W. D. Solomon (1997). The GRAIL Concept Modelling Language for Medical Terminology. *Artificial Intelligence in Medicine* **9**: 139–171.

Renz, J. and B. Nebel (1997). On the Complexity of Qualitative Spatial Reasoning: A Maximal Tractable Fragment of the Region Connection Calculus. In *Proceedings of the 15th International Joint Conference on Artificial Intelligence (IJCAI'97)*. Morgan Kaufmann, 522–527.

Sattler, U. (1996). A Concept Language Extended with Different Kinds of Transitive Roles. In *20. Deutsche Jahrestagung für Künstliche Intelligenz*, volume 1137 in Lecture Notes in Artificial Intelligence. Springer, 333–345.

Schild, K. (1991). A Correspondence Theory for Terminological Logics. In *Proceedings of the 12th International Joint Conference on Artificial Intelligence (IJCAI'91)*. Morgan Kaufmann, 466–471.

Schmidt, R. A. (1997). *Optimised Modal Translation and Resolution*. PhD thesis, Universität des Saarlandes, Saarbrücken, Germany.

Schmidt, R. A. (1999). Decidability by Resolution for Propositional Modal Logics. *Journal of Automated Reasoning* **22**(4): 379–396.

Schild, K. D. (1993). Combining Terminological Logics with Tense Logic. In *Proceedings of the* 6^{th} *Portuguese Conference on Artificial Intelligence (EPIA'93)*. Springer, 105–120.

van Benthem, J. (1976). Modal Correspondence Theory. PhD Thesis, Mathematisch Instituut & Instituut voor Grondslagenonderzoek, University of Amsterdam, The Netherlands.

van der Hoek, W., B. van Linder and J.J. Ch. Meyer (1997). An Integrated Modal Approach to Rational Agents. Technical Report UU-CS-1997-06, Department of Computer Science, Utrecht University, The Netherlands.

Wolter, F. and M. Zakharyaschev. Modal Description Logic: Modalizing Roless. *Fundamenta Informaticae* **39**: 411–438.

Wolter, F. and M. Zakharyaschev (2000). Temporalizing Description Logics. In *Proceedings of the Second International Workshop on Frontiers of Combining Systems (FroCoS'98)*. Research Studies Press, 391–401.

Wolter, F., M. Zakharyaschev and M. Mosurovic (2001). Temporal Description Logics: a Point-based Approach. In *Italian Association for Artificial Intelligence "AI*IA" Notizie" journal*.

Wooldridge, M., C. Dixon, and M. Fisher (1998). A Tableau-Based Proof Method for Temporal Logics of Knowledge and Belief. *Journal of Applied Non-Classical Logics* **8**(3): 225–258.

Wooldridge, M. and N. R. Jennings (1995). Intelligent Agents: Theory and Practice. *The Knowledge Engineering Review* **10**(2): 115–152.