Review of Lohmann, Vollmer "Complexity Results for Modal Dependence Logic"

Robering, Klaus

Published in:
Mathematical Reviews

Publication date:
2014

Document version
Early version, also known as pre-print

Citation for published version (APA):

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.
The article gives a comprehensive survey of the computational complexity of the satisfiability problem for various modal dependence logics (MDLs); cf. MR MR2985117 as well as MR2565920. MDL studies the dependence of the truth value of some proposition \( p_n \) upon the truth values of certain other ones \( p_1, \ldots, p_{n-1} \). That \( p_n \) depends upon \( p_1, \ldots, p_{n-1} \) is expressed by \( \equiv(p_1, \ldots, p_{n-1}, p_n) \). A formula of this type is called a "dependency atom (of arity \( n - 1 \))". Besides a set \( AP \) of atomic propositions, the language of MDL comprises the following constructors: the modal operators \( \Box, \Diamond \); the connectives \( \land, \lor \) (dependency disjunction), \( \oplus \) (classical disjunction), and \( \neg \) (atomic negation); and, finally, the Boolean constants \( \top, \bot \). MDL is the set of all formulas of MDL, and \( MDL_k \) is the set of all formulas which do not contain dependency atoms of an arity greater than \( k \).

MDL is interpreted with respect to Kripkean frames \( W = (S, R, \pi) \) consisting of a set \( S \) of worlds, an accessibility relation \( R \subseteq S \times S \) and a labeling function \( \pi: S \rightarrow \mathcal{P}(AP) \) which specifies for each world \( s \in S \) the set \( \pi(s) \subseteq AP \) of atomic propositions which are, so to speak, "locally true" at \( s \). Formulas are, however, not locally evaluated with respect to single worlds but rather with respect to entire sets of worlds, so-called "evaluation sets" or "teams". A propositional letter \( p \) is true in frame \( W \) with respect to the evaluation set \( T \), i.e., \( W, T \models p \), iff \( p \in \pi(s) \) for every \( s \in T \). Similarly, \( W, T \models \neg p \) iff \( p \notin \pi(s) \) for every \( s \in T \). A proposition \( p_n \) depends in \( T \) upon the propositions \( p_1, \ldots, p_{n-1} \) iff the truth value of \( p_n \) is the same in every pair of worlds \( s_1, s_2 \in T \) which coincide with respect to the truth values of \( p_1, \ldots, p_{n-1} \). More formally, this is expressed by (1) below.

\[
W, T \models \equiv(p_1, \ldots, p_n) \iff \forall s_1, s_2 \in T. [\pi(s_1) \cap \{p_1, \ldots, p_{n-1}\} = \pi(s_2) \cap \{p_1, \ldots, p_{n-1}\} \Rightarrow (p_n \in \pi(s_1) \iff p_n \in \pi(s_2))] \tag{1}
\]

Negated dependency atoms are true in the empty evaluation set and only in it; i.e., \( W, T \models \neg \equiv(p_1, \ldots, p_n) \iff T = \emptyset \). The semantic clause for classical disjunction requires that \( \varphi \lor \psi \) is true at \( T \) if this is the case for at least one of the disjuncts. Dependency disjunction, however, is treated as in (2).

\[
W, T \models \varphi \lor \psi \iff \exists T_1, T_2 \subseteq S. [T = T_1 \cup T_2 \& W, T_1 \models \varphi \& W, T_2 \models \psi] \tag{2}
\]

The semantic clauses for the modalities in DML are given below in (3) and (4).

\[
W, T \models \Box \varphi \iff W, \{s' \in T \mid \exists s \in T, (s, s') \in R\} \models \varphi \tag{3}
\]

\[
W, T \models \Diamond \varphi \iff \exists T' \subseteq S. [W, T' \models \varphi \& \forall s \in T. \exists s' \in T'(s, s') \in R] \tag{4}
\]

Now, let \( M \) be some subset of the set of MDL-constructors, then \( MDL(M) \) is the set of formulas which can be built up by using the connectives from \( M \) only. Similarly, \( MDL_k(M) \) is the set of formulas which do not contain dependency atoms of arity
greater than $k$. The authors investigate the following two types of decision problems
MDL-SAT($M$) and MDL$_k$-SAT($M$) (for $k \geq 3$): Given a formula $\varphi$ from MDL($M$) (or
MDL$_k$($M$), respectively), is there a frame $W$ and an evaluation set $T$ with $W, T \models \varphi$? It
is shown that MDL-SAT($M$) is PSPACE-complete for $M = \{\Box, \Diamond, \land, \lor, \neg\}$ as well as
for all extensions of this set by one or more of $\top, \bot$, and $\emptyset$. However, as soon as the
dependency connective $\equiv()$ is added, the complexity rises to NEXPTIME. One of the
main technical results of the article is that this is still the complexity for the fragment
which results by dropping the disjunctive connectives. For the case of the the “pure”
modal logic $K$, this fragment has been called “Poor Man’s Logic” in MR1864605,
where it has been proved to be co-NP. Lohmann and Vollmer show that Poor Man’s
Modal Dependence Logic is NEXPTIME-complete. If, furthermore, atomic negation
is eliminated, too, (but classical disjunction retained), the complexity drops to $\Sigma^p_2$. If,
instead of dropping modal disjunction and atomic negation, the arity of dependency
atoms is restricted to an arity less or equal $k$ for some $k \geq 3$, then the complexity be-
comes $\Sigma^p_3$. The authors prove a plethora of further complexity results of a similar kind,
which in their entirety provide an exhaustive picture of the complexities of the various
subsystems of MDL. Summing up the many results achieved, one might say that the
addition of the $\equiv()$-connective to a subsystem of MDL either rises its complexity to
NEXPTIME-completeness or does neither increase its complexity nor its expressive
power.

Reviewed by Klaus Robering