Syddansk Universitet

Review of Areces, C.; Figueira, D.; Figueira, S.; Mera, S. "The Expressive Power of Memory Logics"

Robering, Klaus

Published in:
Mathematical Reviews

Publication date:
2012

Document version
Accepted author manuscript

Citation for pulished version (APA):
Robering, K. (2012). Review of Areces, C.; Figueira, D.; Figueira, S.; Mera, S. "The Expressive Power of Memory Logics". Mathematical Reviews, (Review MR2841607).

MR2841607 03B45 03B60 03B25
Areces, Carlos; Figueira, Diego; Figueira, Santiago; Mera, Sergio
The Expressive Power of Memory Logics.
Rev. Symb. Log. 4 (2011), 290–317
DOI 10.1017/s1755020310000389

The article provides an overview over a new class of modal logics, namely "memory logics", it investigates the expressive power of such logics, and surveys some results concerning the decidability problems arising with them. A memory logic is formulated within a language extending propositional logic by adding two novel memory operators ⓡ (unary, "remember") and ⓚ (0-ary, "know"). In addition to these operators there may be a family of possibility operators $\langle r\rangle$ (with r from some fixed non-empty set REL of binary relation symbols) or a family of corresponding "memorizing" possibility operators $\langle\langle r\rangle\rangle$. Finally, there may be additional memory operators such as the unary operators ⓔ ("erase") and ⓕ ("forget"). In order to interpret such a language, extended Kripke-structures $\mathcal{M} = \langle D, \mathcal{I}, M\rangle$ are used. D is the set of possible worlds and the interpretation function $\mathcal{I}$ assigns subsets of D to propositional variables and binary relations $\mathcal{I}(r) \subseteq D \times D$ to the relation symbols $r \in$ REL. The component M of $\mathcal{M}$ is a "memory" which can store worlds in the course of the evaluation of formulas. In the simplest case M is just a set of possible worlds, however the authors also consider the case that the memory is a stack. In that case operators (`push`) and (`pop`) corresponding to the stack operations are used and a further operator (`top`) checks whether the current world of evaluation is the top-most element of the stack. In the simpler case of "set memories", the ⓡ-operator stores the current world of evaluation into the memory: $\langle D, \mathcal{I}, M\rangle, w \models$ ⓡ φ iff $\langle D, \mathcal{I}, M \cup \{w\}\rangle, w \models \varphi$. The ⓚ-operator checks whether the current world is in the memory: $\langle D, \mathcal{I}, M\rangle, w \models$ ⓚ iff $w \in M$. ⓔ empties the entire memory whereas ⓕ removes the current world from it. The $\langle r\rangle$-operators are interpreted in the standard way and the $\langle\langle r\rangle\rangle$-operators differ from them in that they evaluate modal formulas and simultaneously remember the world of evaluation: $\langle D, \mathcal{I}, M\rangle, w \models \langle\langle r\rangle\rangle\, \varphi$ iff there is a $w' \in D$ such that $(w, w') \in \mathcal{I}(r)$ and $\langle D, \mathcal{I}, M \cup \{w\}\rangle, w' \models \varphi$.

Different systems $\mathcal{ML}$ of memory logic (formulated in different languages) are defined as sets of formulas valid in certain classes of models. If the models in such a class are required to have an empty memory (which only becomes filled in the course of an evaluation), this is marked by adding the index "$\emptyset$" to the name of the logic. The operators present in a logic beside the obligatory ⓡ and ⓚ are marked in the logic's name by adding a corresponding list. Thus, for instance, $\mathcal{ML}_\emptyset(\langle r\rangle)$ is the logic of those formulas which besides ⓡ and ⓚ contain only operators $\langle r\rangle$ (if at all) and which are valid in the corresponding models with empty memory. In contrast to this, $\mathcal{ML}(\langle r\rangle)$ admits also models whose initial memory is not void. If a stack is used as memory instead of a simple set, this is marked by the superscript "st", e.g.: $\mathcal{ML}^{st}_\emptyset(\langle r\rangle)$. Besides memory logics proper the basic normal logic $\mathcal{K}$ and the hybrid logic $\mathcal{HL}(\downarrow)$ are considered by the authors as limiting cases. A logic $\mathcal{L}$ is defined to be at most as expressive as a logic $\mathcal{L}'$ ($\mathcal{L} \leq \mathcal{L}'$) if there is a translation function Tr from $\mathcal{L}$ to $\mathcal{L}'$ such that for each formula φ of $\mathcal{L}$ it holds true that $\mathcal{M} \models_{\mathcal{L}} \varphi$ iff $\mathcal{M} \models_{\mathcal{L}'} \mathsf{Tr}(\varphi)$. $\mathcal{L}$ is strictly less expressive than $\mathcal{L}'$ iff $\mathcal{L} \leq \mathcal{L}'$ but $\mathcal{L}' \not\leq \mathcal{L}$. The authors show the following results: **(1)** The logics $\mathcal{K}$, $\mathcal{ML}_\emptyset(\langle\langle r\rangle\rangle)$, $\mathcal{ML}_\emptyset(\langle r\rangle)$, $\mathcal{ML}_\emptyset(\langle r\rangle$, ⓔ$)$, and

$\mathcal{ML}_\emptyset(\langle r\rangle, ⓔ, ⓕ))$ are strictly increasing in expressive power. **(2)** $\mathcal{ML}_\emptyset(\langle r\rangle) < \mathcal{ML}_\emptyset(\langle r\rangle, ⓕ) \leq \mathcal{ML}_\emptyset(\langle r\rangle, ⓔ, ⓕ)$. **(3)** $\mathcal{ML}_\emptyset(\langle r\rangle, ⓕ) \not\leq \mathcal{ML}_\emptyset(\langle r\rangle, ⓔ)$. **(4)** The memory logic $\mathcal{ML}^{st}_\emptyset(\langle r\rangle)$ is expressively equivalent to the hybrid logic $\mathcal{HL}(\downarrow)$ and $\mathcal{ML}_\emptyset(\langle r\rangle, ⓔ, ⓕ)$ is at most as expressive as these two equivalent logics. **(5)** As regards decidability, it is shown that $\mathcal{ML}(\langle\langle r\rangle\rangle)$ is decidable whereas $\mathcal{ML}_\emptyset(\langle\langle r\rangle\rangle)$ and $\mathcal{ML}(\langle r\rangle)$ are not. Furthermore it is shown that $\mathcal{ML}(\langle r\rangle)$ lies on the edge of decidability. Adding one single nominal it—i.e., a propositional symbol true in just one single possible world—renders this logic undecidable.

Reviewed by Klaus Robering