# Efficient Theorem-Proving for Modal Logics

Cláudia Nalon [1]

*Department of Computer Science, University of Brasília, Brazil*

**Abstract**

KSP is a resolution-based theorem prover for multimodal normal logics that implements decision procedures for the local and global satisfiability problems for the multimodal logic $\mathsf{K}_n$, all logics in the cube and logics of confluence (with parameters 1 or 0). In this talk, we discuss two different resolution-based calculi for those logics that are implemented in the prover with particular focus on the characteristics that have impact on the efficiency of the implementation

*Keywords:* Modal Logics, Resolution, Automated Reasoning.

Modal logics have long been used in Computer Science for describing and reasoning about complex systems, with applications to programming languages, knowledge representation and reasoning, terminological reasoning, and verification. The basic normal multimodal modal logic is $\mathsf{K}_n$, which extends the classical language with new operators, $[a]$ and $\langle a\rangle$, with $a \in \mathcal{A}_n = \{1, \ldots, n\}$, a fixed finite set of indexes. The local satisfiability problem for the multimodal propositional case is PSPACE-complete [2]. The global satisfiability and the local satisfiability under global constraints problems for $\mathsf{K}_n$ are EXPTIME-complete [13]. Some applications might require specific extensions of modal logics that better represent constraints for a particular given domain, which can be achieved by adding axioms to $\mathsf{K}_n$. Here, we consider the logics in the modal cube (the ones obtained by adding one or more of the axioms $\mathbf{B}_a$, $\mathbf{D}_a$, $\mathbf{T}_a$, $\mathbf{4}_a$, and $\mathbf{5}_a$) and a subset of logics of confluence (the ones obtained by adding one or more of the instances of the parameterised axiom $\mathbf{G}_a^{p,q,r,s}$: $\langle a\rangle^p[a]^q\varphi \rightarrow [a]^r\langle a\rangle^s\varphi$, with $p, q, r, s \in \{0, 1\}$). Those axioms and their first-order characterisation are given in Fig. 1.

Given the wide range of applications for which those logics can be used for both representation and reasoning, and considering the complexity of their corresponding satisfiability problems, the development of automatic, efficient tools is desirable. Several proof methods and tools for modal reasoning exist, either in the form of methods applied direct to the modal language or obtained by translation into more expressive languages (first-order or high-order logic,

[1] Email: *nalon@unb.br*. Joint work with Ullrich Hustadt (University of Liverpool), Clare Dixon (University of Manchester) and Fabio Papacchini (Lancaster University in Leipzig).

| (p,q,r,s) | Name | Axioms | Property | Condition on Frames |
|---|---|---|---|---|
| $(0,0,1,1)$<br>$(1,1,0,0)$ | $\mathbf{B}_a$ | $\varphi \rightarrow [a]\langle a\rangle\varphi$<br>$\langle a\rangle[a]\varphi \rightarrow \varphi$ | symmetric | $wR_aw_1 \rightarrow w_1R_aw$ |
| $(0,0,1,0)$<br>$(1,0,0,0)$ | $\mathbf{Ban}_a$ | $\varphi \rightarrow [a]\varphi$<br>$\langle a\rangle\varphi \rightarrow \varphi$ | modally banal | $wR_aw_1 \rightarrow w = w_1$ |
| $(0,1,0,1)$ | $\mathbf{D}_a$ | $[a]\varphi \rightarrow \langle a\rangle\varphi$ | serial | $\exists w_1(wR_aw_1)$ |
| $(1,0,1,0)$ | $\mathbf{F}_a$ | $\langle a\rangle\varphi \rightarrow [a]\varphi$ | functional | $wR_aw_1 \wedge wR_aw_2 \rightarrow w_1 = w_2$ |
| $(0,0,0,1)$<br>$(0,1,0,0)$ | $\mathbf{T}_a$ | $\varphi \rightarrow \langle a\rangle\varphi$<br>$[a]\varphi \rightarrow \varphi$ | reflexive | $wR_aw$ |
| $(1,0,1,1)$<br>$(1,1,1,0)$ | $\mathbf{5}_a$ | $\langle a\rangle\varphi \rightarrow [a]\langle a\rangle\varphi$<br>$\langle a\rangle[a]\varphi \rightarrow [a]\varphi$ | Euclidean | $wR_aw_1 \wedge wR_aw_2 \rightarrow w_1R_aw_2$ |
| $(1,1,1,1)$ | $\mathbf{G1}_a$ | $\langle a\rangle[a]\varphi \rightarrow [a]\langle a\rangle\varphi$ | convergent | $wR_aw_1 \wedge wR_aw_2 \rightarrow$<br>$\exists w_3(w_1R_aw_3 \wedge w_2R_aw_3)$ |
| $(0,1,1,1)$<br>$(1,1,0,1)$ | $\mathbf{G}_a^{0,1,1,1}$ | $[a]\varphi \rightarrow [a]\langle a\rangle\varphi$<br>$\langle a\rangle[a]\varphi \rightarrow \langle a\rangle\varphi$ | 0,1,1,1<br>convergent | $wR_aw_1 \rightarrow$<br>$\exists w_2(wR_aw_2 \wedge w_1R_aw_2)$ |
| $(0,1,2,0)$<br>$(2,0,1,0)$ | $\mathbf{4}_a$ | $[a]\varphi \rightarrow [a][a]\varphi$<br>$\langle a\rangle\langle a\rangle\varphi \rightarrow \langle a\rangle\varphi$ | transitive | $wR_aw_1 \wedge w_1R_aw_2 \rightarrow w_1R_aw_2$ |

Fig. 1. Axioms and conditions on frames for logics of confluence, where $\mathbf{G}_a^{p,q,r,s}$: $\langle a\rangle^p[a]^q\varphi \rightarrow [a]^r\langle a\rangle^s\varphi$; the free variables $(w, w_1, w_2)$ are universally quantified.

for instance). Here we briefly present two different proof methods for $\mathsf{K}_n$, namely the clausal modal resolution procedures described in [3,4].

The calculus given in [3] (GMR) operates on a very simple normal form, called $\mathsf{SNF}_K$, which allows for total separation of the propositional and modal parts of a problem. However, the clausal language requires clauses to be in the scope of the universal operator [*], which might lead to the production of clauses that cannot ever be used in a refutation. The calculus given in [4] (MLR) minimises this problem by taking advantage of well-known properties of the satisfiability problem for $\mathsf{K}_n$, namely: a formula $\varphi$ is satisfiable if, and only if, it is satisfiable in a finite tree-like model; and also that checking the satisfiability of a subformula associated with a level $ml$ of the tree depends only on the subformulae occurring at levels greater or equal to $ml$.The clausal normal form, called $\mathsf{SNF}_{ml}$, introduces labels in $\mathbb{N} \cup \{*\}$ to clauses corresponding to those levels, and resolution inferences can only be applied whenever the labels unify. The special label $*$ is used if a clause is true at all depths/worlds in a model and it normally only occurs in the normal form if we want to check a formula for global satisfiability. This restriction also allows for a very efficient implementation. The inference rules for MLR are given in Figure 2 and can only be applied if the unification on labels is defined, where the function $\sigma$ on sets of labels is given by $\sigma(\{ml, *\}) = ml$; and $\sigma(\{ml\}) = ml$; otherwise, $\sigma$ is undefined. For the GMR calculus [3], the inference rules are similar to those in Figure 2, but inference rules are applied within the scope of [*] instead of using the labels. The additional inference rules are given in Figure 3, where constant **start** is meant to be true only at the root of a model.

KSP is a theorem prover for the basic multimodal logic $\mathsf{K}_n$, first presented in [5] (see also [7] for an extended presentation of its architecture and features). It is written in C and includes both calculi mentioned above. The main loop is

$$[\mathsf{LRES}]\ \frac{ml_1 :\ D \vee l \qquad ml_2 :\ D' \vee \neg l}{ml :\ D \vee D'} \qquad [\mathsf{GEN2}]\ \frac{ml_1 : l'_1 \rightarrow [a] l_1 \qquad ml_2 : l'_2 \rightarrow [a] \neg l_1 \qquad ml_3 : l'_2 \rightarrow \langle a \rangle l_2}{ml :\ \neg l'_1 \vee \neg l'_2 \vee \neg l'_3} \qquad [\mathsf{MRES}]\ \frac{ml_1 :\ l_1 \rightarrow [a] l \qquad ml_2 :\ l_2 \rightarrow \langle a \rangle \neg l}{ml : \neg l_1 \vee \neg l_2}$$

$$[\mathsf{GEN1}]\ \frac{\begin{array}{r} ml_1 :\ l'_1 \rightarrow [a] \neg l_1 \\ \vdots \\ ml_m : l'_m \rightarrow [a] \neg l_m \\ ml_{m+1} :\ l' \rightarrow \langle a \rangle \neg l \\ ml_{m+2} :\ l_1 \ \vee\ \ldots \vee l_m \vee l \end{array}}{ml :\ \neg l'_1 \vee \ldots \vee \neg l'_m \vee \neg l'} \qquad [\mathsf{GEN3}]\ \frac{\begin{array}{r} ml_1 :\ l'_1 \rightarrow [a] \neg l_1 \\ \vdots \\ ml_m : l'_m \rightarrow [a] \neg l_m \\ ml_{m+1} :\ l' \rightarrow \langle a \rangle l \\ ml_{m+2} :\ l_1 \ \vee\ \ldots \vee l_m \end{array}}{ml :\ \neg l'_1 \vee \ldots \vee \neg l'_m \vee \neg l'}$$

Fig. 2. Rules in MLR, where $ml = \sigma(\{ml_1, \ldots, ml_{m+1}, ml_{m+2} - 1\})$ in GEN1, GEN3; $ml = \sigma(\{ml_1, ml_2\})$ in LRES, MRES; and $ml = \sigma(\{ml_1, ml_2, ml_3\})$ in GEN2.

based on the given-clause algorithm, a variation of the set of support strategy [14], a refinement which restricts the set of choices of clauses participating in a derivation step. For the modal calculus, the set of clauses is partitioned according to the modal layer at which clauses are true, noting that for the GMR calculus, except for the initial clauses, all clauses are in the same level.

Previous experimental evaluations [1,5,6,7,11,12] indicate that KSP with MLR works well on problems with high nesting of modal operators where the separation of modal levels can be exploited to improve the efficiency of reasoning [5,6,11,8,12]. The evaluations also indicate that KSP's performance is comparable to other state-of-the-art provers with built-in support for $\mathsf{K}_1$.

For the remaining logics in the cube, the implementation of the MLR calculus can deal with the satisfiability problem via the reductions given in [11,8,12]. The reductions are functions that take a formula in the extended logic and produce an equisatisfiable set of $\mathsf{SNF}_{ml}$ clauses in $\mathsf{K}_n$. Some of those functions might return infinite sets. However, bounds for each of the logics and their combinations were determined so only a finite number of clauses is required. The GMR calculus can deal with any combination of the multimodal logics with axioms presented in Fig. 1, as given in [3,10], where the inference rules for each logic are added to the calculus. Evaluations with KSP and GMR show good performance for logics including the axioms $\mathbf{4}_1$ and $\mathbf{5}_1$ [8,9].

We finish by noting that all evaluations ever performed, as indicated above, considered reasoning for monomodal logics (either $\mathsf{K}_1$ or its extensions in the modal cube). However, KSP is truly multimodal, in the sense that not only multiple modalities can be used, but also more than one logic can be specified in order to represent (and reason about) a problem. With GMR, KSP implements

$$[\mathsf{IRES1}]\ \frac{[*](\ \mathbf{start} \rightarrow D \ \vee\ l) \qquad [*](\ \mathbf{start} \rightarrow D' \vee \neg l)}{[*](\ \mathbf{start} \rightarrow D \ \vee\ D')} \qquad [\mathsf{IRES2}]\ \frac{[*](\ \mathbf{start} \rightarrow D \vee\ \ l) \qquad [*](\ \ \mathbf{true} \rightarrow D' \vee \neg l)}{[*](\ \mathbf{start} \rightarrow D \vee\ \ D')}$$

Fig. 3. GMR rules for initial clauses.

decision procedures for logics which are interesting for some applications (e.g. including seriality and functionality for temporal modalities), but these features have never been extensively tested. There is a lack of provers that deal with all those logics and also a need for more challenging benchmarks. The combination of the global and local calculus is currently under implementation. We believe that with this new feature, we can further improve on the number of problems solved using the reductions given in [9] and previous works.

## References

[1] Goré, R. and C. Kikkert, *Cegar-tableaux: Improved modal satisfiability via modal clause-learning and sat*, in: A. Das and S. Negri, editors, *TABLEAUX 2021* (2021), pp. 74–91.
[2] Halpern, J. Y. and Y. Moses, *A guide to completeness and complexity for modal logics of knowledge and belief*, Artificial Intelligence **54** (1992), pp. 319–379.
[3] Nalon, C. and C. Dixon, *Clausal resolution for normal modal logics*, J. Algorithms **62** (2007), pp. 117–134.
[4] Nalon, C., C. Dixon and U. Hustadt, *Modal resolution: Proofs, layers, and refinements*, ACM Trans. Comput. Log. **20** (2019), pp. 23:1–23:38.
[5] Nalon, C., U. Hustadt and C. Dixon, *$K_S$P: A resolution-based prover for multimodal K*, in: N. Olivetti and A. Tiwari, editors, *IJCAR 2016*, LNCS **9706** (2016), pp. 406–415.
[6] Nalon, C., U. Hustadt and C. Dixon, *$K_S$P: A resolution-based prover for multimodal K, abridged report*, in: C. Sierra, editor, *Proceedings of IJCAI 2017* (2017), pp. 4919–4923.
[7] Nalon, C., U. Hustadt and C. Dixon, *$K_S$P: Architecture, refinements, strategies and experiments*, J. Autom. Reason. **64** (2020), pp. 461–484.
[8] Nalon, C., U. Hustadt, F. Papacchini and C. Dixon, *Local reductions for the modal cube*, in: *IJCAR 2022*, LNCS **13385** (2022), pp. 486–505.
[9] Nalon, C., U. Hustadt, F. Papacchini and C. Dixon, *Buy one get 14 free: Evaluating local reductions for modal logic*, in: B. Pientka and C. Tinelli, editors, *Proceedings of CADE-29*, Lecture Notes in Artificial Intelligence **14132** (2023), pp. 382–400.
[10] Nalon, C., J. Marcos and C. Dixon, *Clausal resolution for modal logics of confluence*, in: S. Demri, D. Kapur and C. Weidenbach, editors, *Proceedings of IJCAR 2014*, LNCS **8562** (2014), pp. 322–336.
[11] Papacchini, F., C. Nalon, U. Hustadt and C. Dixon, *Efficient local reductions to basic modal logic*, in: A. Platzer and G. Sutcliffe, editors, *CADE 2021*, LNCS **12699** (2021), pp. 76–92.
[12] Papacchini, F., C. Nalon, U. Hustadt and C. Dixon, *Local is best: Efficient reductions to modal logic K*, J. Autom. Reason. **66** (2022), pp. 639–666.
[13] Spaan, E., "Complexity of Modal Logics," Ph.D. thesis, University of Amsterdam (1993).
[14] Wos, L., G. A. Robinson and D. F. Carson, *Efficiency and Completeness of the Set of Support Strategy in Theorem Proving*, Journal of the ACM **12** (1965), pp. 536–541.