Automated reasoning about metric and topology

Ullrich Hustadt[1], Dmitry Tishkovsky[1], Frank Wolter[1], and
Michael Zakharyaschev[2]

[1] Department of Computer Science, University of Liverpool
{U.Hustadt,D.Tishkovsky,F.Wolter}@csc.liv.ac.uk
[2] School of Computer Science and Information Systems, Birkbeck College
michael@dcs.bbk.ac.uk

1 Introduction

In this paper we compare two approaches to automated reasoning about metric and topology in the framework of the logic $\mathcal{MT}$ introduced in [10]. *$\mathcal{MT}$-formulas* are built from *set variables* $p_1, p_2, \ldots$ (for arbitrary subsets of a metric space) using the Booleans $\wedge$, $\vee$, $\rightarrow$, and $\neg$, the *distance operators* $\exists^{<a}$ and $\exists^{\leq a}$, for $a \in \mathbb{Q}^{>0}$, and the topological *interior* and *closure operators* $\mathbf{I}$ and $\mathbf{C}$. *Intended models* for this logic are of the form $\mathfrak{I} = (\Delta, d, p_1^{\mathfrak{I}}, p_2^{\mathfrak{I}}, \ldots)$ where (Δ, d) is a metric space and $p_i^{\mathfrak{I}} \subseteq \Delta$. The *extension* $\varphi^{\mathfrak{I}} \subseteq \Delta$ of an $\mathcal{MT}$-formula φ in $\mathfrak{I}$ is defined inductively in the usual way, with $\mathbf{I}$ and $\mathbf{C}$ being interpreted as the interior and closure operators induced by the metric, and $(\exists^{<a}\varphi)^{\mathfrak{I}} = \{x \in \Delta \mid \exists y \in \varphi^{\mathfrak{I}}\ d(x,y) < a\}$. In other words, $(\mathbf{I}\varphi)^{\mathfrak{I}}$ is the interior of $\varphi^{\mathfrak{I}}$, $(\exists^{<a}\varphi)^{\mathfrak{I}}$ is the open a-neighbourhood of $\varphi^{\mathfrak{I}}$, and $(\exists^{\leq a}\varphi)^{\mathfrak{I}}$ is the closed one. A formula φ is *satisfiable* if there is a model $\mathfrak{I}$ such that $\varphi^{\mathfrak{I}} \neq \emptyset$; φ is *valid* if $\neg\varphi$ is not satisfiable.

In $\mathcal{MT}$, one can represent various basic facts about metric and topology. For example, the validity of $\exists^{<a}p \rightarrow \mathbf{I}\exists^{<a}p$ means that the open a-neighbourhood of any set is open. The non-validity of $\mathbf{C}\exists^{<a}p \rightarrow \exists^{\leq a}p$ means that there is a metric space with a subset X such that the closure of the open a-neighbourhood of X properly contains the closed a-neighbourhood of X. The logic $\mathcal{MT}$ as well as its metric fragment $\mathcal{MS}$ without the topological operators have been suggested as basic tools for reasoning about distances and similarity [9].

One obvious approach to automated reasoning with $\mathcal{MT}$ is to use the standard ε-definition of the topological interior

$$\mathbf{I}X \ = \ \{x \in X \mid \exists\varepsilon > 0\ \forall y(d(x,y) < \varepsilon \rightarrow y \in X\}$$

and translate $\mathcal{MT}$ into a two-sorted first-order language, with one sort for the real numbers and the other one for points of metric spaces. This approach allows the use of interactive systems supporting a theory of real numbers, like HOL, Isabelle, or PVS. However, it is unlikely to be a viable basis for efficient automatic reasoning about $\mathcal{MT}$ or $\mathcal{MS}$, which is our focus.

The alternative approach we present here is based on the results obtained in [10] which show that the intended metric models for $\mathcal{MT}$ can be (equivalently) replaced by relational Kripke-style models where the distance operators are interpreted (like in modal logic) by binary relations and the interior operator by a quasi-order. More precisely, suppose for simplicity that we are given an $\mathcal{MT}$-formula φ whose numerical parameters (in the operators $\exists^{<a}$, $\exists^{\leq a}$) are all natural numbers, and N is the maximal one. A *relational φ-model* is a structure

$$\mathfrak{R} \;=\; (W, R_1, \ldots, R_N, S_1, \ldots, S_N, R, p_1^{\mathfrak{R}}, p_2^{\mathfrak{R}}, \ldots)$$

where, for $0 < a \leq N$, (i) R_a and S_a are reflexive and symmetric binary relations on $W \neq \emptyset$, (ii) R is reflexive and transitive, (iii) $R \subseteq R_a \subseteq S_a$, (iv) $S_a \subseteq R_b$ for $a < b$, (v) xS_ayS_bz implies $xS_{a+b}z$ whenever $a + b \leq N$, (vi) xR_ayS_bz implies $xR_{a+b}z$ whenever $a+b \leq N$, (vii) xS_ayR_bz implies $xR_{a+b}z$ whenever $a+b \leq N$, and (viii) xR_ayRz implies xR_az. The operator $\exists^{<a}$ is interpreted in $\mathfrak{R}$ (in the standard Kripke style) by means of R_a, $\exists^{\leq a}$ by S_a, and $\mathbf{I}$ by R. Then, according to [10], we have the following theorem: *an $\mathcal{MT}$-formula φ is satisfiable in a metric model iff it is satisfiable in a relational φ-model.*

2 Reasoning

The relational semantics above enables a variety of reasoning techniques to be applied to the satisfiability and validity problem in $\mathcal{MT}$. Here we focus on just two: a tableau calculus, which forms the basis of our MetTeL system, and first-order translation, which allows the use of a range of existing theorem provers.

2.1 Tableau calculus

In our tableau calculus we use a modified version of the tableau rules for hybrid logic (see e.g. [2]) with additional rules for the metric and topology operators which follow the semantics described in Section 1. For example, we use the standard tableau rules for modal logic S4 to capture the behaviour of the interior operator plus the following rules expressing interaction between the topological and metric operators (where i, j, and k are nominals):

$$\frac{@_i\exists^{<a}j \quad @_j\mathbf{I}k}{@_i\exists^{<a}k} \qquad \frac{@_i\neg\exists^{<a}j}{@_i\neg\mathbf{I}j} \qquad \frac{@_i\neg\exists^{<a}\varphi \quad @_i\mathbf{I}j}{@_j\neg\varphi} \qquad \frac{@_i\neg\exists^{\leq a}\varphi \quad @_i\mathbf{I}j}{@_j\neg\varphi}$$

Note that nominals are not part of $\mathcal{MT}$ and only serve as a technical tool in the calculus. The part of the tableau calculus related to metric operators and nominals is actually equivalent to the labelled tableau algorithm in [9]. The MetTeL system implements this tableau calculus and provides a decision procedure for $\mathcal{MT}$. MetTeL is implemented in Java 1.5.

2.2 First-order translation

The intuition behind the first-order encoding is to use the standard relational translation for modal logics [4] including a representation of the semantic conditions (i)–(viii) presented in Section 1. For example, occurrences of $\exists^{\leq a}\varphi$ are translated according to

$$\pi_r(\exists^{\leq a}\varphi, x) \;=\; \exists y\, (S_a(x, y) \wedge \pi_r(\varphi, y)),$$

while conditions (i) and (v), for S_a, are represented by (1) $\forall x\,(S_a(x,x))$, (2) $\forall x,y\,(S_a(x,y) \rightarrow S_a(y,x))$, and (3) $\forall x,y,z\,(S_a(x,y) \wedge S_b(y,z) \rightarrow S_c(x,z))$, for $c = a + b \leq N$. Note that the number of formulae of the form (1) and (2) which we need to include in the translation is linear in N, while for formulae of the form (3) it is quadratic in N. Our implementation of the translation also allows for the application of structural transformation and the application of alternatives to the relational translation, e.g. the axiomatic translation, [6], but in the following we restrict ourselves to the approach described above.

3 Comparisons

To establish whether the techniques presented in Section 2 provide viable means for reasoning about metric formulae with topology operators, we have devised a set of sample formulae, divided into the following groups of formulae (plus the single formula metric-axioms given by the negated conjunction of all metric axioms). (a) The formulae in the textbook group generalise the examples of interaction between the topological and metric operators from Section 1. (b) Let $\forall^{<a}\varphi = \neg\exists^{<a}\neg\varphi$. Then path-box-p.$n$ and path-box-u.n are parametric series of formulae of the shape

$$\underbrace{\forall^{<1}\cdots\forall^{<1}}_{n} p \rightarrow \forall^{<n} p \qquad \text{and} \qquad \neg(\forall^{<n} p \rightarrow \underbrace{\forall^{<1}\cdots\forall^{<1}}_{n} p),$$

respectively, with $n = 4, 8, 12, 20, 24, 32$, which test the general triangle inequality while we increase the nesting depth of $\forall^{<1}$. (c) Groups symm-box-p.n and symm-box-u.n are parametric series of formulae of the following shape:

$$p \rightarrow \forall^{<n}\neg\underbrace{\forall^{<1}\cdots\forall^{<1}}_{n}\neg p \qquad \text{and} \qquad \neg(p \rightarrow \underbrace{\forall^{<1}\cdots\forall^{<1}}_{n}\neg\forall^{<n}\neg p),$$

respectively, with $n = 4, 8, 12, 20, 24, 32$ which test the interaction of symmetry and the general triangle inequality while we increase the nesting depth of $\forall^{<1}$.

For the first-order translation approach we have used a range of state-of-the-art first-order theorem provers, Darwin 1.1 [1], DCTP 1.31 [3], E 0.91 [7], SPASS 2.2 [8], Vampire 7.0 [5]. While the last three are based on resolution calculi, Darwin and DCTP are based on the model evolution and the disconnection calculus, respectively. Each prover was executed on each sample formula with a timelimit of 1000 CPU seconds. In the case of first-order translation, the time required to perform the translation has been included. All tests were performed on a 2.8GHz Pentium 4 PC with 1024MB main memory under RedHat Linux 9. Figure 1 shows a summary of the results. Time is measured in user CPU seconds. A 'Timeout' entry indicates that the CPU timelimit was exceeded while a 'Fail' entry indicates that the reasoner failed before the timelimit was reached, e.g. because it was running out of memory.

Concerning the four parametric series of formulae, we see that increasing the parameter n increases the time required to solve a formula. The difference between the various approaches and provers is the extent to which it does so. Overall, DCTP on the relational translation performs best. The three resolution

Sample problem	Status	MetTeL	Darwin	DCTP	E	SPASS	Vampire
textbook.00	unsat	0.28	0.14	0.15	0.15	0.12	0.16
textbook.01	sat	0.22	0.16	0.14	0.16	0.13	0.16
textbook.02	unsat	6.63	5.16	0.16	2.34	26.68	33.42
textbook.03	sat	0.22	0.18	0.14	0.15	0.14	0.25
metric-axioms	unsat	0.88	32.01	0.68	*T/O*	*T/O*	*Fail*
path-box-p.04	sat	0.29	0.10	0.14	0.16	0.12	0.17
path-box-p.08	sat	2.86	0.28	0.14	0.20	0.39	48.03
path-box-p.12	sat	17.54	1.11	0.20	0.35	5.74	*T/O*
path-box-p.20	sat	236.63	17.57	0.55	2.25	312.68	*T/O*
path-box-p.24	sat	633.02	47.23	0.99	7.59	*T/O*	*T/O*
path-box-p.32	sat	*T/O*	230.76	3.51	45.05	*T/O*	*T/O*
path-box-u.04	unsat	0.43	0.15	0.14	0.18	0.13	0.16
path-box-u.08	unsat	2.34	0.20	0.14	1.42	0.23	0.16
path-box-u.12	unsat	14.61	0.92	0.23	14.49	1.85	0.43
path-box-u.20	unsat	194.87	14.14	0.92	977.59	73.53	3.42
path-box-u.24	unsat	575.69	35.54	1.39	*T/O*	273.85	9.87
path-box-u.32	unsat	*T/O*	171.98	4.14	*T/O*	*T/O*	408.76
symm-box-p.04	sat	0.15	0.32	0.13	1.32	0.27	0.36
symm-box-p.08	sat	0.18	104.66	0.17	*T/O*	*T/O*	*T/O*
symm-box-p.12	sat	0.19	*T/O*	0.19	*T/O*	*T/O*	*T/O*
symm-box-p.20	sat	0.22	*Fail*	0.56	*T/O*	*T/O*	*T/O*
symm-box-p.24	sat	0.20	*Fail*	1.04	*T/O*	*T/O*	*T/O*
symm-box-p.32	sat	0.23	*Fail*	3.50	*T/O*	*T/O*	*T/O*
symm-box-u.04	unsat	0.30	0.15	0.14	0.16	0.14	0.17
symm-box-u.08	unsat	2.30	0.26	0.17	1.89	0.23	0.18
symm-box-u.12	unsat	12.41	1.06	0.28	18.73	1.89	0.81
symm-box-u.20	unsat	191.92	16.36	1.24	996.37	73.66	4.62
symm-box-u.24	unsat	517.35	42.04	2.67	*T/O*	272.38	13.99
symm-box-u.32	unsat	*T/O*	191.54	7.02	*T/O*	*T/O*	480.36

Fig. 1. Performance of various provers on sample metric formulae

provers have more difficulty on satisfiable formulae compared to all other provers, but not uniformly so: on the path-box-p.n series, E performs better than the tableau system MetTeL. On the other hand, MetTeL performs better than E on the path-box-u.n and the symm-box-u.n series of unsatisfiable formulae, which might also be surprising. SPASS and Vampire seem roughly on par.

References

1. P. Baumgartner, A. Fuchs, and C. Tinelli. Implementing the model evolution calculus. *International Journal of Artificial Intelligence Tools*, 15(1), 2005.
2. P. Blackburn and M. Marx. Tableaux for quantified hybrid logic. In *Proc. TABLEAUX 2002*, LNAI 2381, pages 38–52, 2002.
3. R. Letz and G. Stenz. DCTP: A disconnection calculus theorem prover. In *Proc. IJCAR 2001*, LNAI 2083, pages 381–385. Springer, 2001.
4. H. J. Ohlbach, A. Nonnengart, M. de Rijke, and D. M. Gabbay. Encoding two-valued nonclassical logics in classical logic. In A. Robinson and A. Voronkov, editors, *Handbook of Automated Reasoning*, pages 1403–1485. Elsevier, 2001.
5. A. Riazanov and A. Voronkov. The design and implementation of VAMPIRE. *AI Comm.*, 15(2-3):91–110, 2002.
6. R. A. Schmidt and U. Hustadt. A principle for incorporating axioms into the first-order translation of modal formulae. In *Proc. CADE-19*, LNAI 2741, pages 412–426. Springer, 2003.
7. S. Schulz. E: A Brainiac theorem prover. *AI Comm.*, 15(2/3):111–126, 2002.
8. C. Weidenbach, U. Brahm, T. Hillenbrand, E. Keen, C. Theobald, and D. Topic. SPASS version 2.0. In *Proc. CADE-18*, LNAI 2392, pages 275–279. Springer, 2002.
9. F. Wolter and M. Zakharyaschev. Reasoning about distances. In *Proc. of IJCAI 2003*, pages 1275–1280. Morgan Kaufmann, 2003.
10. F. Wolter and M. Zakharyaschev. A logic for metric and topology. *Journal of Symbolic Logic*, 70:795–828, 2005.