A Tableau Compiled Labelled Deductive System for Hybrid Logic

Krysia Broda and Alessandra Russo
*Department of Computing Imperial College
South Kensington London SW7 2AZ
{k.broda,a.russo}@imperial.ac.uk

Abstract

Compiled Labelled Deductive Systems (CLDS) provide a uniform logical framework where families of different logics can be given a uniform proof system and semantics. A variety of applications of this framework have been proposed so far ranging from extensions of classical logics (e.g. normal modal logics and multi-modal logics) to non-classical logics such as resource and substructural loogics. Labelled natural deduction proof systems have been developed and proved to be a generalization of existing proof systems for each of these logics. This extended abstract gives a brief presentation of the CLDS framework and outlines how it can be applied to develop a labeled tableau system for Hybrid Logic.

1 Introduction

Labelled Deductive Systems (LDS) were introduced by Gabbay (6) as a unifying framwork for different families of logics. Logics within a given family have often similar syntax and similar sets of proof rules: they differ mainly on their semantic properties. LDS formalization of these logics provide a way of defining uniform proof systems where the semantic properties are syntactically captured by a labeling algebra and handled alongside with the object level formulae. Labelled deductive systems, based on a propositional language, for families of modal, conditional, resource logics and access control logics, have been well investigated(1; 2; 3; 4; 5). A Compiled Labelled Deductive Systems (CLDS),builds upon the LDS framework in that it provides similar labeled deductive proof system approach, but also provides a model theoretical semantic approach, based on a compilation technique into first-order semantics, that is uniformly applicable to any (first-order axiomatizable) logic. The compilation technique can be also adapted to provide an associated automated theorem prover for any of these logics. In most of the previous applications, the CLDS proof system has been presented using natural deduction. This extended abstract shows in some detail how uniform tableau systems can be developed within the CLDS framework looking in particular at the case of Hybrid Logic.

Building upon the main features of a LDS framework (6), in a CLDS the semantic features of a given logic are expressed syntactically by means of a *labelling algebra* that define properties over *labels* (i.e. terms of a labelling language $\mathcal{L}_L$). A CLDS theory, called *configuration*, consists of a set of *declarative units* and a *diagram*. Declarative units are of the form $\alpha:\lambda$, where α is a wff from a given language $\mathcal{L}_P$ and λ is a label, whereas a *diagram* is a set of ground relational properties over labels. By varying the properties of the labelling algebra different logics belonging to a given family of logics can be captured within the same CLDS framework. For example, in the family of Hybrid Logics, labels may be related by a binary relation R, analogous to an accessibilty relation, or by the equality relation. Different properties of the R relation capture different logics. In the family of resource logics, labels represent resources and the labelling algebra includes an operation for combining resources as well as a relation for comparing different resource combinations. The proof system of a CLDS manipulates configurations: it includes rules for deriving new declarative units and new diagram's elements. The semantics is, instead, defined in terms of a first-order axiomatization of the given logic, that extends the given labelling algebra. In almost all previous applications of the CLDS framework the proof system has been presented using a natural deduction style, in which each step derives a configuration $\mathcal{C}$' from a given (or previously derived) configuration $\mathcal{C}$. The framework has been applied to various Modal Logics, including Conditional Logics(1) and Access Control Logic(2), where labels represent accessible worlds of a Kripke-like diagram, and resource logics (5; 9), where labels represent resources. It has also been deployed in the context of clausal abduction(12), where the proof technique was based on resolution and labels were sets of abducibles. These investigations have shown that main advantages of a CLDS framework are that (i) it facilitates a *uniform* presentation of logics belonging to the same family (i.e. different logics within the same family can be captured by appropriately changing the labelling algebra), (ii) its first-order based semantics facilitates the implementation of sound theorem provers for the logic in question; and finally (iii) it enables the investigation of completely new logics. The next section gives an overview of CLDS and exemplifies the method by applying it to Hybrid Logic and using, instead, a tableau proof system.

2 CLDS – Overview of Application to Hybrid Logic

The basic syntactic entities of a CLDS are *declarative units*, of the form *formula:label*, in which formulas are written in a *formula language* $\mathcal{L}_P$ and labels in a *labelling language* $\mathcal{L}_L$, and *R-literals* as atomic formulae written in the relational signature of the given labelling language $\mathcal{L}_L$[1] A *configuration* is a set of declarative units and a *diagram* is a set of R-literals. The *proof rules* allow reasoning over declarative units only, R-literals only, or reasoning between both kinds of information. The proof system could be defined to be natural deduction, tableau or others such as sequent calculus. The *labelling algebra* is a first order theory written in $\mathcal{L}_L$ and defines properties of the relational signature of $\mathcal{L}_L$. The semantics is defined in terms of an *extended labelling algebra*, which is usually a set of first order axiom schema written in an *extended labelling language* $\mathcal{L}_L^+$, which defines the semantic meanings of sentences in the object language. This algebra makes use of a *translation function* that maps configurations into first order theories written in $\mathcal{L}_L^+$.

The above notions are illustrated via a brief description of H_{CLDS}, a CLDS for Hybrid Logic. Hybrid Logic, as described by Blackburn in (7), is about "internalising labels"; certain propositions, called *nominals*, are used to name possible worlds. For instance the formula $p : i \wedge \Diamond i$ is a formula in Hybrid Logic, where p is a (standard) proposition and i is a nominal, which can be manipulated using appropriate proof rules. Rules that apply to this formula can be used to derive the formulas $p : i$, $\Diamond i$, $i : c$, $p : c$, where c is a nominal new to the derivation. Other operators of Hybrid Logic (10) can also be captured within H_{CLDS}, but are omitted here for brevity.

In H_{CLDS}, the language $\mathcal{L}_P$ consists of propositions, various operators, such as the modal $\Diamond$ and the *transfer operator* @, and nominal atoms of the form N_i, where i is a label. This has also been proposed in (8). The idea is that a nominal atom captures the notion of a nominal in Hybrid Logic and the transfer operator captures the notion of internalisation of labels. The labelling language is a first-order language with equality and the binary predicate R, corresponding to modal world accessibility. A *configuration* might be for instance the set of declarative units $\{p : i,\ \ N_i : j,\ \ @_j \neg p : k\}$. Using appropriate proof rules, the new declarative unit $\neg p : j$ and R-literal $i = j$ can be derived from the given one, then using $p : i$ and $i = j$ the declarative unit $p : j$ and then $\perp$ can be derived, showing that the initial configuration is inconsistent. The labelling algebra includes the substitutivity properties of equality, so that, for example, $R(i, i)$ can be derived from $R(i, f_p(i))$ and $i = f_p(i)$. The configuration tableau given above is, respectively, translated into the first order formulas $\{[p]^*(i), [N_i]^*(j)$ and $[@_j \neg p]^*(k)$, in which $[p]^*$ and $[N_i]^*$ are predicates derived from the propositions p and N_i.

Monadic predicates of the form $[\alpha]^*$, where α belongs to $\mathcal{L}_P$, belong to the extended labelling language. The extended labelling algebra appropriate for the above formulas includes, for example, the first order axiom schema (1) to (4) below. More general schema (5), (6) and (7) are included for illustration, where $f_p(x)$ names the new accessible world introduced by the $\Diamond$-elimination rule.

(1)	$\forall x(N_i(x) \rightarrow i = x)$
(2)	$\forall x([\alpha]^*(i) \wedge i = x \rightarrow [\alpha]^*(x))$
(3)	$\forall x([@_j\alpha]^*(x) \rightarrow [\alpha]^*(j))$
(4)	$\forall x([\neg @_j\alpha]^*(x) \rightarrow [\neg\alpha]^*(j))$
(5)	$\forall x([\neg\alpha]^*(x) \rightarrow \neg[\alpha]^*(x))$
(6)	$\forall x([\alpha \wedge \beta]^*(x) \rightarrow [\alpha]^*(x) \wedge [\beta]^*(x))$
(7)	$\forall x([\neg(\alpha \wedge \beta)]^*(x) \rightarrow [\neg\alpha]^*(x) \vee [\neg\beta]^*(x))$
(8)	$\forall x([\Diamond p]^*(x) \rightarrow R(x, f_p(x)) \wedge [p]^*(f_p))$
(9)	$\forall x, y([\neg\Diamond\alpha]^*(x) \wedge R(x, y) \rightarrow [\alpha]^*(y))$

From these, together with the labelling algebra and the translated configuration above, it is easy to derive a contradiction. The schema in the extended labelling algebra give meaning to the translated formulas. For instance, (3) gives meaning to the transfer operator. Depending on the exact type of Hybrid Logic used, the corresponding rule in Hybrid logic is either $\neg i : @_j p \iff j : p$ or $\neg i : j : p \iff \neg j : p$, for nominals i and j.

In a tableau system it is normal to restrict rules to be *analytic*, which means that new formulas in the tableau are only introduced if they are proper sub-formulas or negations of proper sub-formulas of formulas in the trunk of the tableau, which is the initial set of formulas from which the tableau is developed. Accordingly, the semantic schema in the extended labelling algebra are *configuration relative*, meaning that they are restricted to include only analytic instances. (This contrasts with the stance taken for natural deduction CLDSs, in which such a restriction is not necessarily made.)

The semantics of a CLDS is defined in terms of its first order translation and the first order semantic and syntactic entailment relations denoted by $\models$ and $\vdash$ respectively. The translation of a configuration $\mathcal{C}$ is written as $Tr(\mathcal{C})$ and consists of the translation of each item in the configuration. The extended labelling algebra, called $\mathcal{A}_{\mathcal{C}}^+$, consists of the labelling algebra and one instance of the appropriate semantic schema for each non-atomic formula in the configuration and for each proper sub-formula in the configuration. The exact set of formulas in the extended labelling algebra depends on the configuration. For example, if a configuration consists of the declarative unit $\neg p \wedge q : 0$, there will be schema instances for $p \wedge q$ and $\neg p$ in the extended labelling algebra (p and q are atomic).

Let $\mathcal{C}$ and $\mathcal{C}'$ be two configurations, then semantic entailment between configurations in a CLDS is defined by $\mathcal{C} \models_{CLDS} \mathcal{C}'$ if, and only if, $\mathcal{A}_{\mathcal{C}}^+ \cup Tr(\mathcal{C}) \models Tr(\mathcal{C}')$For a tableau-based CLDS entailment is made with respect to refutations and is amended to $\mathcal{C} \models_{CLDS} \perp$ if, and only if, $\mathcal{A}_{\mathcal{C}}^+ \cup Tr(\mathcal{C}) \models \perp$)The soundness requirement for a CLDS then requires first to

[1] The first modal CLDS used just one relation called R, which gave rise to the terminology R-literal.

show the property that if a configuration tableau formed from a configuration $\mathcal{C}$ closes, then the translation $Tr(\mathcal{C})$ can be made to close using the extended labelling algebra $\mathcal{A}_{\mathcal{C}}^{+}$. This property and the soundness of first order logic can be used to conclude that $Tr(\mathcal{C}) \cup \mathcal{A}_{\mathcal{C}}^{+} \models \bot$ and then the definition of entailment gives the desired result $\mathcal{C} \models_{CLDS} \bot$. Of these steps, only the proof of closure is specialised to the particular CLDS.

The completeness of a tableau-based CLDS is also shown through its translation and the method follows the saturated-tableau approach. (For natural deduction-based CLDS the method used follows the Henkin approach.) If a configuration $\mathcal{C}$ is such that it cannot be closed, i.e. after all possible rules have been applied to all branches of the tableau there remains at least one branch $\mathcal{B}$ still open, then a (Herbrand) model for $\mathcal{C}$ in the extended labelling language can be extracted from $\mathcal{B}$, which will show that $\mathcal{A}_{\mathcal{C}}^{+} \cup Tr(\mathcal{C}) \not\models \bot$ and hence that $\mathcal{C} \nvdash_{\mathrm{TH}} \bot$.

In the case that a CLDS is claimed to implement a known logic $\mathcal{L}$, then it needs also to be shown that the implementation is faithful to $\mathcal{L}$. That is, that it has exactly the same theorems as $\mathcal{L}$. This is called the *correspondence property* and can be shown to hold in various ways depending on the particular CLDS and the presentation of $\mathcal{L}$.

Figure 1 shows a tableau derivation using H_{CLDS} of the formula $\Diamond p : w$ from the configuration $\{\Diamond N_i : w,\ p : i\}$. The first 5 steps are fairly standard. Line 6 makes uses the rule (Nom), which captures the property of a nominal N_i, whereby it is true only at the label it names, in this case i. Line 7 makes use of equality substitution.

1	$\Diamond N_i : w$	Given (w is an arbitrary label)	6	$i = f_{N_i}(w)$	(Nom), 5
2	$p : i$	Given	7	$p : f_{N_i}(w)$	(=sub),2,6
3	$\neg\Diamond p : w$	Negated goal	8	$\neg p : f_{N_i}(w)$	($\neg\Diamond$), 3
4	$R(w, f_{N_i}(w))$	($\Diamond$), 1	9	$\bot$	(Close), 7, 8
5	$N_i : f_{N_i}(w)$	($\Diamond$), 1			

Figure 1: Example H_{CLDS} derivation

3 Future Work

The outline application of CLDS to Hybrid Logic given in Section 2 can be completed to give a sound and complete CLDS which corresponds to Hybrid Logic. Current investigations are focused on two aspects; (i) how to exploit the additional expressiveness in CLDS achieved by the Hybrid operators, and (ii) how the tableau formulation enables decidability results to be incorporated into CLDS, as is achieved in (11).

References

[1] K. Broda, D. Gabbay, L. C. Lamb and A. Russo. Labelled natural deduction for conditional logics of normality, *Logic Journal of the IGPL*, 10(2), pp 123-163, 2002.

[2] K. Broda and A. Russo. Compiled Labelled Deductive Systems for Access Logic, in *We Will Show Them!, Essays in Honour of Dov Gabbay on his 60th Birthday*, Volume 1, Eds. S. Artemov et al, College Publications, pp 309-337, 2005.

[3] K. Broda, A. Russo and D. Gabbay. A unified compilation style natural deduction system for modal, substructural and fuzzy logics. In *Discovering World with Fuzzy Logic: Perspectives and Approaches to Formalization of Human-consistent Logical Systems*, Springer, 2000.

[4] A. Russo. *Modal Logics as Labelled Deductive Systems*, PhD Thesis, Dept. of Computing, Imperial College London, 1996.

[5] K. Broda, M. Finger and A. Russo. Labelled natural deduction for substructural logics, *Logic Journal of the IGPL*, 7(3), pp 283-318, 1999.

[6] D. Gabbay. *Labelled Deductive Systems: Volume I*, Oxford Logic Guides, Vol. 33, Oxford Clarendon Press, 1996.

[7] P. Blackburn. Internalising Labelled Deduction, *Journal of Logic and Computation*, 10(1), OUP, pp 137-168, 2000.

[8] P. Blackburn and B. ten Cate. Pure extensions, proof rules, and hybrid axiomatics. *Studia Logica* 84(2): 277-322, 2006.

[9] K. Broda. A Decidable CLDS for some Propositional Resource Logics, in *Computational Logic: Logic Programming and Beyond*, Eds. A. C. Kakas and F. Sadri, LNAI 2407, pp 135 - 159

[10] M. Tsakova. Tableau Calculi for Hybrid Logics, Tableaux99, LNAI 1617, pp 278-292, 1999

[11] T. Bolander and T. Brauner. Two Tableau-Based Decision Procedures for Hybrid Logic, in *M4M-4 Workshop* , Berlin, 2005.

[12] K. Broda and D. Gabbay. Labelled Abduction - Compiled Labelled Abductive Systems, in *Labelled Deduction*, Eds. D. Basin et al, Kluwer Academic Publishers, 2000.