Polynomial Approximations of Full Propositional Logic via Limited Bivalence

Marcelo Finger*

Computer Science Department
Institute of Mathematics and Statistics
University of São Paulo, Brazil
mfinger@ime.usp.br

Abstract. The aim of this paper is to study an anytime family of logics that approximates classical inference, in which every step in the approximation can be decided in polynomial time. For clausal logic, this task has been shown to be possible by Dalal [Dal96a,Dal96b]. However, Dalal's approach cannot be applied to full classical logic.
In this paper we provide a family of logics, called *Limited Bivalence Logics*, that approximates full classical logic. Our approach contains two stages. In the first stage, a family of logics parameterised by a set of formulas Σ is presented. A lattice-based semantics is given and a sound and complete tableau-based proof-theory is developed. In the second stage, the first family is used to create another approximation family, in which every approximation step is shown to be polynomially decidable.
Keywords: Approximated Reasoning, Polynomial Approximations.

1 Introduction

The computational costs associated with logical reasoning have always been a limitation to its use in the modelling of intelligent agents. Even if we restrict ourselves to classical propositional logic, deciding whether a set of formulas logically implies a certain formula is a co-NP-complete problem [GJ79].

To address this problem, researchers have proposed several ways of approximating classical reasoning. Cadoli and Schaerf have proposed the use of approximate entailment as a way of reaching at least partial results when solving a problem completely would be too expensive [SC95]. Their influential method is *parametric*, that is, a set S of atoms is the basis to define a logic. As we add more atoms to S, we get "closer" to classical logic, and eventually, when S contains all propositional symbols, we reach classical logic. This kind of approximation has been called "approximating from below" [FW04] and is useful for efficient theorem proving.

The notion of approximation is also related with the notion of an *anytime decision procedure*, that is an algorithm that, if stopped anytime during the

* Partly supported by CNPq grant PQ 300597/95-5 and FAPESP project 03/00312-0.

computation, provides an approximate answer. Such an answer is of the form "yes" or "up to logic L_i in the family, the result is not provable". To remain inside a logic framework along the approximation process, it is necessary that every approximate logic L_i have a clear semantics, so that if the anytime process is interrupted at L_i, we know exactly where we are.

Dalal's approximation method [Dal96a] was designed such that each reasoner in an approximation family can be decided in polynomial time. Dalal's initial approach was algebraic only. A model-theoretic semantics was provided in [Dal96b]. However, this approach was restricted to clausal form logic only, its semantics had an unusual format, an no proof-theoretical presentation was given.

In this work, we generalise Dalal's approach, obtaining a polynomial approximation family for full propositional logic, with a lattice-based semantics and a tableau-based proof theory. We do that in two steps. The first step develops a family of logics of *Limited Bivalence* (LB), and provide a lattice-based semantics for it. The entailment $\models_{\Sigma}^{\mathsf{LB}}$ is a parametric approximation on the set of formulas Σ that follows Cadoli and Schaerf's approximation paradigm. We also provide a tableau-based inference $\vdash_{\Sigma}^{\mathsf{KELB}}$, and prove it sound and complete with respect to $\models_{\Sigma}^{\mathsf{LB}}$. In the second step, we derive an inference $\vdash_{k}^{\mathsf{KELB}}$ based on $\vdash_{\Sigma}^{\mathsf{KELB}}$ and an entailment relation $\models_{k}^{\mathsf{LB}}$ based on $\models_{\Sigma}^{\mathsf{LB}}$, and obtain the soundness and completeness of $\vdash_{k}^{\mathsf{KELB}}$ in terms of $\models_{k}^{\mathsf{LB}}$. We then show that $\vdash_{k}^{\mathsf{KELB}}$ is polynomially decidable.

This paper proceeds as follows. Section 2 presents Dalal's approximation strategy, its semantics and discuss its limitations. In Section 3 we present the family $\mathsf{LB}(\Sigma)$; a semantics for full propositional $\mathsf{LB}(\Sigma)$ is provided and the parametric entailment $\models_{\Sigma}^{\mathsf{LB}}$ is presented; we also give a proof-theoretical characterisation based on KE-tableaux, $\vdash_{\Sigma}^{\mathsf{KELB}}$. The soundness and completeness of $\vdash_{\Sigma}^{\mathsf{KELB}}$ with respect to $\models_{\Sigma}^{\mathsf{LB}}$ is proven in Section 4. The family of inference systems $\vdash_{k}^{\mathsf{KELB}}$ and its semantics $\models_{k}^{\mathsf{LB}}$ are presented in Section 5, and $\vdash_{k}^{\mathsf{KELB}}$ is shown to be polynomially decidable.

Notation: Let $\mathcal{P}$ be a countable set of propositional letters. We concentrate on the classical propositional language $\mathcal{L}_C$ formed by the usual boolean connectives $\rightarrow$ (implication), $\wedge$ (conjunction), $\vee$ (disjunction) and $\neg$ (negation).

Throughout the paper, we use lowercase Latin letters to denote propositional letters, α, β, γ denote formulas, φ, ψ denote clauses and λ denote a literal. Uppercase Greek letters denote sets of formulas. By atoms(α) we mean the set of all propositional letters in the formula α; if Σ is a set of formulas, atoms$(\Sigma) = \bigcup_{\alpha \in \Sigma}$ atoms(α).

2 Dalal's Polynomial Approximation Strategy

Dalal [Dal96a] specifies a family of *anytime* reasoners based on an equivalence relation between formulas and on a restricted form of Cut Rule. The family is composed of a sequence of reasoners $\vdash_0, \vdash_1, \ldots$, such that each $\vdash_i$ is tractable, each $\vdash_{i+1}$ is at least as complete (with respect to classical logic) as $\vdash_i$, and for each theory there is a complete $\vdash_i$ to reason with it.

Dalal provides as an example a family of reasoners based on the classically sound but incomplete inference rule known as BCP (Boolean Constraint Propagation) [McA90], which is a variant of unit resolution [CL73]. Consider a theory as a set of clauses, where a disjunction of zero literals is denoted by $\mathbf{f}$. Let $\sim\psi$ be the *complement* of the formula ψ obtained by pushing the negation inside in the usual way using De Morgan's Laws until the atoms are reached, at which point $\sim p = \neg p$ and $\sim\neg p = p$. The equivalence relation $=_{\text{BCP}}$ is then defined as:

$$\{\mathbf{f}\} \cup \Gamma =_{\text{BCP}} \{\mathbf{f}\}$$
$$\{\lambda, \sim\lambda \vee \lambda_1 \vee \ldots \vee \lambda_n\} \cup \Gamma =_{\text{BCP}} \{\lambda, \lambda_1 \vee \ldots \vee \lambda_n\} \cup \Gamma$$

where λ, λ_i are literals. The inference $\vdash_{\text{BCP}}$ is defined as $\Gamma \vdash_{\text{BCP}} \psi$ iff $\Gamma \cup \{\sim\psi\} =_{\text{BCP}} \{\mathbf{f}\}$.

Dalal [Dal96b] presents an example in which, for the theory $\Gamma_0 = \{p \vee q, p \vee \neg q, \neg p \vee s \vee t, \neg p \vee s \vee \neg t\}$, we both have $\Gamma_0 \vdash_{\text{BCP}} p$ and $\Gamma_0, p \vdash_{\text{BCP}} s$ but $\Gamma_0 \not\vdash_{\text{BCP}} s$.

This example shows that $\vdash_{\text{BCP}}$ is unable to use a previously inferred clause p to infer s. Based on this fact comes the proposal of an anytime family of incomplete reasoners $\vdash_0^{\text{BCP}}, \vdash_1^{\text{BCP}}, \ldots$, where each $\vdash_k^{\text{BCP}}$ is given by the following:

$$1.\ \frac{\Gamma \vdash_{\text{BCP}} \varphi}{\Gamma \vdash_k^{\text{BCP}} \varphi} \qquad 2.\ \frac{\Gamma \vdash_k^{\text{BCP}} \psi \qquad \Gamma, \psi \vdash_k^{\text{BCP}} \varphi}{\Gamma \vdash_k^{\text{BCP}} \varphi} \text{ for } |\psi| \leq k$$

where $|\psi|$, the size of a clause ψ, is the number of literals it contains.

The first rule tells us that every $\vdash_{\text{BCP}}$-inference is also a $\vdash_k^{\text{BCP}}$-inference. The second rule tells us that if ψ was inferred from a theory and it can be used as a further hypothesis to infer φ, and the size of ψ is at most k, then φ is can also be inferred from the theory.

Dalal shows that this is indeed an anytime family of reasoners, that is, for each k, $\vdash_k^{\text{BCP}}$ is tractable, $\vdash_{k+1}^{\text{BCP}}$ is as complete as $\vdash_k^{\text{BCP}}$ and for each classically inferable $\Gamma \vdash \varphi$ there is a k such that $\Gamma \vdash_k^{\text{BCP}} \varphi$.

In [Dal96b], a semantics for $\vdash_k^{\text{BCP}}$ is proposed based on the notion of *k-valuations*. This semantics has a peculiar format: literals are evaluated to real values over the interval $[0, 1]$ but clauses are evaluated to real values over $[0, +\infty)$. A formula ψ is satisfied by valuation v if $v(\psi) \geq 1$. A *k-model* is a set V of k-valuations, such that if ψ, $|\psi| \leq k$, has a non-model in V, ie $v(\psi) < 1$, then it has a k-countermodel in V, ie $v(\psi) = 0$. It then defines $\Gamma \models\!\!\approx_k \psi$ iff there is no k-countermodel of ψ in any k-model of Γ. Here we simply state Dalal's main results.

Proposition 1 ([Dal96b]). *For every theory Γ and every clause ψ:*

i. $\Gamma \vdash_{\text{BCP}} \psi$ *iff* $\Gamma \models\!\!\approx_0 \psi$ *and* $\Gamma \vdash_k^{\text{BCP}} \psi$ *iff* $\Gamma \models\!\!\approx_k \psi$.
ii. $\Gamma \vdash_k^{\text{BCP}} \psi$ *can be decided in polynomial time.*

Thus the inference $\vdash_k^{\text{BCP}}$ is sound and complete with respect to $\models\!\!\approx_k$ for clausal form formulas and, for a fixed value of k, it can be decided in polynomial time.

Dalal's notion of a family of anytime reasoners has very nice properties. First, every step in the approximation is sound and can be decided in polynomial time. Second, the approximation is guaranteed to converge to classical inference. Third, every step in the approximation has a sound and complete semantics, enabling an anytime approximation process.

However, the method based on $\vdash_k^{\text{BCP}}$-approximations also has its limitations:

1. It only applies to clausal form formulas. Although every propositional formula is classically equivalent to a set of clauses, this equivalence may not be preserved in any of the approximation step. The conversion of a formula to clausal form is costly: one either has to add new propositional letters (increasing the complexity of the problem) or the number of clauses can be exponential in the size of the original formula. With regards to complexity, BCP is a form of resolution, and it is known that there are theorems that can be proven by resolution only in exponentially many steps [CS00].
2. Its non-standard semantics makes it hard to compare with other logics known in the literature, specially other approaches to approximation. Also, the semantics presented is impossible to generalise to non-clausal formulas.
3. The proof-theory for $\vdash_k^{\text{BCP}}$ is poor in computational terms. In fact, if we are trying to prove that $\Gamma \vdash_k^{\text{BCP}} \varphi$, and we have shown that $\Gamma \not\vdash_{\text{BCP}} \varphi$, then we would have to guess a ψ with $|\psi| \leq k$, so that $\Gamma \vdash_k^{\text{BCP}} \psi$ and $\Gamma, \psi \vdash_k^{\text{BCP}} \varphi$. Since the BCP-approximations provides no method to guess the formula ψ, this means that a computation would have to generate and test all the $O((2n)^k)$ possible clauses, where n is the number of propositional symbols occurring in Γ and φ.

In the following we present an approximation method that maintains all the positive aspects of $\vdash_k^{\text{BCP}}$ and avoids some of the criticism above. That is, it is applicable to all propositional formulas, whether clausal or not, and has a lattice-based semantics. This will allow non-resolution proof methods to be used in the approximation process. In particular, we present a tableaux based proof theory that is sound and complete with respect to the semantics. A family of reasoners is then built, each element of which is polynomially decidable.

3 The Family of Logics $\mathsf{LB}(\Sigma)$

We present here the family of logics of *Limited Bivalence*, $\mathsf{LB}(\Sigma)$. This is a parametric family that approximates classical logic, in which every approximation step can be decided in polynomial time. Unlike $\vdash_k^{\text{BCP}}$, $\mathsf{LB}(\Sigma)$ is parameterised by a set of formulas Σ.

The family $\mathsf{LB}(\Sigma)$ can be applied to the full language of propositional logic, and not only to clausal form formulas, with an alphabet consisting of a countable set of propositional letters (atoms) $\mathcal{P} = \{p_0, p_1, \ldots\}$, and the connectives $\neg$, $\wedge$, $\vee$ and $\rightarrow$, and the usual definition of well-formed propositional formulas; the set of all well-formed formulas is denoted by $\mathcal{L}$. The presentation of LB is made in terms of a model theoretic semantics.

We require the parameter set Σ to be *closed under formula formation*, that is, if $\alpha \in \Sigma$ then $\neg\alpha \in \Sigma$; if $\alpha, \beta \in \Sigma$ then $\alpha \circ \beta \in \Sigma$, for $\circ \in \{\wedge, \vee, \rightarrow\}$.

3.1 Semantics of $\mathsf{LB}(\Sigma)$

The semantics of $\mathsf{LB}(\Sigma)$ is based of a three-level lattice, $L = (L, \sqcap, \sqcup, 0, 1)$, where L is a countable set of elements $L = \{0, 1, \varepsilon_0, \varepsilon_1, \varepsilon_2, \ldots\}$ such that $0 \sqsubseteq \varepsilon_i \sqsubseteq 1$ for every $i < \omega$ and $\varepsilon_i \not\sqsubseteq \varepsilon_j$ for $i \neq j$. The ε_i's are called *neutral* truth values. This three-level lattice is illustrated in Figure 1(a).

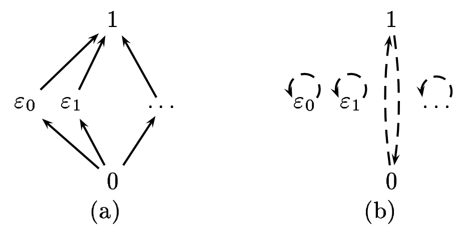

Fig. 1. The 3-Level Lattice (a) and its Converse Operation (b)

This lattice is enhanced with a *converse operation*, $\sim$, defined as: $\sim 0 = 1$, $\sim 1 = 0$ and $\sim\varepsilon_i = \varepsilon_i$ for all $i < \omega$. This is illustrated in Figure 1(b).

We next define the notion of an *unlimited valuation*, and then we limit it. An unlimited propositional valuation is a function $v_\Sigma : \mathcal{P} \rightarrow L$ that maps atoms to elements of the lattice. We extend v_Σ to all propositional formulas, $v_\Sigma : \mathcal{L} \rightarrow L$, in the following way:

$$
\begin{aligned}
&v_\Sigma(\neg\alpha) = \sim v_\Sigma(\alpha) \\
&v_\Sigma(\alpha \wedge \beta) = v_\Sigma(\alpha) \sqcap v_\Sigma(\beta) \\
&v_\Sigma(\alpha \vee \beta) = v_\Sigma(\alpha) \sqcup v_\Sigma(\beta) \\
&v_\Sigma(\alpha \rightarrow \beta) = \begin{cases} 1 & \text{if } v(\alpha) \sqsubseteq v(\beta) \\ \sim v_\Sigma(\alpha) \sqcup v_\Sigma(\beta) & \text{otherwise} \end{cases}
\end{aligned}
$$

A *limited valuation* is a valuation that satisfies the following requirements with regards to whether a formula is or is not in the parameter set Σ:

(a) if $\alpha \in \Sigma$ then $v_\Sigma(\alpha)$ must be *bivalent*, that is, $v_\Sigma(\alpha)$ must satisfy the rules above for unlimited valuations and be such that $v_\Sigma(\alpha) = 0$ or $v_\Sigma(\alpha) = 1$;
(b) if $\alpha \notin \Sigma$ then either $v_\Sigma(\alpha)$ obeys the rules of unlimited valuations *or* $v_\Sigma(\alpha) = \varepsilon_i$, for some ε_i.

These conditions above are called the Limited Bivalence Restrictions. The first conditions forces the elements of Σ to be bivalent. The second condition tells us that the truth value assigned to a formula $\alpha \notin \Sigma$ is not always *compositional*, for a neutral value may be assigned to α independently of the truth value of its components. This is the case so that the bivalence of $\alpha \in \Sigma$ can always be satisfied without forcing all α's subformulas to be bivalent.

If $\alpha \in \Sigma$ it is always possible to have $v_\Sigma(\alpha) \in \{0,1\}$ by making for every atom p in α, $v_\Sigma(p) \in \{0,1\}$. However, this is not the only possibility. For example, if $\beta, \gamma \notin \Sigma$ then we can make $v_\Sigma(\beta) = \varepsilon_i \neq \varepsilon_j = v_\Sigma(\gamma)$, so that $v_\Sigma(\beta \wedge \gamma) = 0$; similarly, we obtain $v_\Sigma(\beta \vee \gamma) = 1$ and $v_\Sigma(\beta \to \gamma) = 1$.

In the case of clausal form formulas, restriction (b) is not necessary provided we treat clauses as sets of literals [Fin04].

In the rest of this work, by a valuation v_Σ we mean a limited valuation subject to the conditions above.

A valuation v_Σ *satisfies* α if $v_\Sigma(\alpha) = 1$, and α is called *satisfiable*; a set of formulas Γ is satisfied by v_Σ if all its formulas are satisfied by v_Σ. A valuation v_Σ *contradicts* α if $v_\Sigma(\alpha) = 0$; if α is neither satisfied nor contradicted by v_Σ, we say that v_Σ is *neutral* with respect to α. A valuation is *classical* if it assigns only 0 or 1 to all proposition symbols, and hence to all formulas.

For example, consider the formula $p \to q$, and $\Sigma = \varnothing$. Then

- if $v_\Sigma(p) = 1$, then $v_\Sigma(p \to q) = v_\Sigma(q)$;
- if $v_\Sigma(p) = 0$, then $v_\Sigma(p \to q) = 1$;
- if $v_\Sigma(q) = 0$, then $v_\Sigma(p \to q) = v_\Sigma(p)$;
- if $v_\Sigma(q) = 1$, then $v_\Sigma(p \to q) = 1$;
- if $v_\Sigma(p) = \varepsilon_p$ and $v_\Sigma(q) = \varepsilon_q$, then $v_\Sigma(p \to q) = 1$;

The first four cases coincide with a classical behaviour. The last one shows that if p and q are mapped to distinct neutral values, then $p \to q$ will be satisfiable. Note that, in this case, $p \vee q$ will also be satisfiable, and that $p \wedge q$ will be contradicted.

3.2 LB-Entailment

The notion of a parameterised LB-Entailment, $\models^{\mathsf{LB}}_\Sigma$, follows the spirit of Dalal's entailment relation, namely $\Gamma \models^{\mathsf{LB}}_\Sigma \alpha$ if it is not possible to satisfy Γ and contradict α at the same time. More specifically, $\Gamma \models^{\mathsf{LB}}_\Sigma \alpha$ if no valuation v_Σ such that $v_\Sigma(\Gamma) = 1$ also makes $v_\Sigma(\alpha) = 0$. Note that since this logic is not classic, if $\Gamma \models^{\mathsf{LB}}_\Sigma \alpha$ and $v_\Sigma(\Gamma) = 1$ it is possible that α is either neutral or satisfied by v_Σ.

For example, we reconsider Dalal's example, where $\Gamma_0 = \{p \vee q, p \vee \neg q, \neg p \vee s \vee t, \neg p \vee s \vee \neg t\}$ and make $\Sigma = \varnothing$. We want to show that $\Gamma_0 \models^{\mathsf{LB}}_\Sigma p$, $\Gamma_0, p \models^{\mathsf{LB}}_\Sigma s$ but $\Gamma_0 \not\models^{\mathsf{LB}}_\Sigma s$.

To see that $\Gamma_0 \models^{\mathsf{LB}}_\Sigma p$, suppose there is a v_Σ such that $v_\Sigma(p) = 0$. Then we have $v_\Sigma(p \vee q) = v_\Sigma(q)$ and $v_\Sigma(p \vee \neg q) = \sim v_\Sigma(q)$. Since it is not possible to satisfy both, we cannot have $v_\Sigma(\Gamma_0) = 1$, so $\Gamma_0 \models^{\mathsf{LB}}_\Sigma p$.

To show that $\Gamma_0, p \models^{\mathsf{LB}}_\Sigma s$, suppose there is a v_Σ such that $v_\Sigma(s) = 0$ and $v_\Sigma(p) = 1$. Then $v_\Sigma(\neg p \vee s \vee t) = v_\Sigma(t)$ and $v_\Sigma(\neg p \vee s \vee \neg t) = \sim v_\Sigma(t)$. Again, it is not possible to satisfy both, so $\Gamma_0, p \models^{\mathsf{LB}}_\Sigma s$.

Finally, to see that $\Gamma_0 \not\models^{\mathsf{LB}}_\Sigma s$, take a valuation v_Σ such that $v_\Sigma(s) = 0, v_\Sigma(p) = \varepsilon_p, v_\Sigma(q) = \varepsilon_q, v_\Sigma(t) = \varepsilon_t$. Then $v_\Sigma(\Gamma_0) = 1$.

However, if we enlarge Σ and make $p \in \Sigma$, then we have only two possibilities for $v_\Sigma(p)$. If $v_\Sigma(p) = 1$, we have already seen that no valuation that contradicts s will satisfy Γ_0. If $v_\Sigma(p) = 0$, we have also seen that no valuation that contradicts s will satisfy Γ_0. So for $p \in \Sigma$, we obtain $\Gamma_0 \models^{\mathsf{LB}}_\Sigma s$.

This example indicates that $\models^{\mathsf{LB}}_\varnothing$ behave in a similar way to $\vdash_{\mathrm{BCP}}$, and that by adding an atom to Σ we have a behaviour similar to $\vdash^{\mathrm{BCP}}_1$. As shown in [Fin04], this is not a coincidence.

An Approximation Process. As defined in [FW04], a family of logics, parameterised with a set Σ is said to be an approximation of classical logic "from below" if, increasing size of the parameter set Σ, we get closer to classical logic. That is, for $\varnothing \subseteq \Sigma' \subseteq \Sigma'' \subseteq \ldots \subseteq \mathcal{L}$ we have that,

$$\models^{\mathsf{LB}}_\varnothing \;\subseteq\; \models^{\mathsf{LB}}_{\Sigma'} \;\subseteq\; \models^{\mathsf{LB}}_{\Sigma''} \;\subseteq\; \ldots \;\subseteq\; \models^{\mathsf{LB}}_{\mathcal{L}} \;=\; \models_{\mathrm{CL}}$$

where $\models_{\mathrm{CL}}$ is classical entailment. It is clear that the family of logics $\mathsf{LB}(\Sigma)$ is an approximation of classical logic from below.

Note that the approximation of $\Gamma \models \alpha$ can be done in a finite number of steps for finite Σ, because when Σ contains all subformulas in $\Gamma \cup \{\alpha\}$ we are in classical logic.

3.3 Tableaux for $\mathsf{LB}(\Sigma)$

We present a proof theory for $\mathsf{LB}(\Sigma)$ based on KE-tableaux [DM94,D'A99], which we call $\mathsf{KELB}(\Sigma)$-tableaux. This is a variation of Smullyan's semantic tableaux [Smu68] that is more suitable to our purposes, for it incorporates the Cut rule in its expansion rules, unlike semantic tableaux which are based on cut-free formulation of logics. In fact, both $\vdash^{\mathrm{BCP}}_k$ and $\mathsf{LB}(\Sigma)$ are approximation families based on the limited validity of the Cut inference rule. Furthermore, KE-tableaux have better computational properties than semantic tableaux [D'A92].

KE-tableaux deal with T- and F-signed formulas. So if α is a formula, $T\ \alpha$ and $F\ \alpha$ are signed formulas. $T\ \alpha$ is the *conjugate formula* of $F\ \alpha$, and vice versa.

Each connective is associated with a set of *linear expansion rules*. Linear expansion rules always have a *main premiss*; two-premissed rules also have an *auxiliary premiss*. Figure 2 shows KE-tableau linear connective expansion rules for classical logic, which are the same for KELB-tableaux.

The only branching rule in KE is the *Principle of Bivalence*, stating that a formula α must be either true or false. In $\mathsf{KELB}(\Sigma)$-tableaux, this rule is *limited* by a proviso stating that it can only occur over a formula $\alpha \in \Sigma$. This *limited principle of bivalence*, LPB(Σ) is illustrated in Figure 3.

We also require a few further linear rules which are redundant in classical KE:

$$\frac{F\ \alpha \wedge \alpha}{F\ \alpha}\,(F\wedge_{\alpha\alpha}) \qquad\qquad \frac{T\ \alpha \vee \alpha}{T\ \alpha}\,(T\vee_{\alpha\alpha})$$

$$\frac{T\ \alpha \to \beta \qquad T\ \alpha}{T\ \beta}\ (T \to_1) \qquad \frac{T\ \alpha \to \beta \qquad F\ \beta}{F\ \alpha}\ (T \to_2) \qquad \frac{F\ \alpha \to \beta}{T\ \alpha \qquad F\ \beta}\ (F \to)$$

$$\frac{F\ \alpha \wedge \beta \qquad T\ \alpha}{F\ \beta}\ (F\wedge_1) \qquad \frac{F\ \alpha \wedge \beta \qquad T\ \beta}{F\ \alpha}\ (F\wedge_2) \qquad \frac{T\ \alpha \wedge \beta}{T\ \alpha \qquad T\ \beta}\ (T\wedge)$$

$$\frac{T\ \alpha \vee \beta \qquad F\ \alpha}{T\ \beta}\ (T\vee_1) \qquad \frac{T\ \alpha \vee \beta \qquad F\ \beta}{T\ \alpha}\ (T\vee_2) \qquad \frac{F\ \alpha \vee \beta}{F\ \alpha \qquad F\ \beta}\ (F\vee)$$

$$\frac{T\ \neg\alpha}{F\ \alpha}\ (T\neg) \qquad \frac{F\ \neg\alpha}{T\ \alpha}\ (F\neg)$$

Fig. 2. KE Expansion Rules

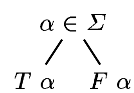

Fig. 3. Limited Principle of Bivalence LPB(Σ)

The premiss and consequence of each such rule are logically equivalent, but in classical KE the consequent can only be derived with the use of the principle of bivalence, which may not be available in KELB if $\alpha \notin \Sigma$.

An expansion of a tableau branch is allowed when the premisses of an expansion rule are present in the branch; the expansion consists of adding the conclusions of the rule to the end of all branches passing through the set of all premisses of that rule. The LPB(Σ) branching rule splits a branch into two.

A branch in a KELB-tableau is *closed* if it contains $F\ \alpha$ and $T\ \alpha$. The tableau is closed if all its branches are closed. We define the inference $\vdash_{\Sigma}^{\mathsf{KELB}}$ such that $\alpha_1, \ldots, \alpha_n \vdash_{\Sigma}^{\mathsf{KELB}} \beta$ iff there is a closed KELB(Σ)-tableau for $T\ \alpha_1, \ldots, T\ \alpha_n, F\ \beta$.

As an example, reconsider Dalal's example given above, presented using full propositional logic. $\Gamma_0 = \{p \vee q, q \to p, p \to (s \vee t), (p \wedge t) \to s\}$. Figure 4 presents three tableaux, one for $\Gamma_0 \vdash_{\varnothing}^{\mathsf{KELB}} p$, the second for $\Gamma_0, p \vdash_{\varnothing}^{\mathsf{KELB}} s$ and a third one, which contains an *incremental method* to establish whether $\Gamma_0 \vdash s$.

The tableaux in Figure 4 for $\Gamma_0 \vdash_{\varnothing}^{\mathsf{KELB}} p$ and $\Gamma_0, p \vdash_{\varnothing}^{\mathsf{KELB}} s$ close without branching. The third tableau is actually a combination of the other two. In it we try to establish whether $\Gamma_0 \vdash_{\varnothing}^{\mathsf{KELB}} s$; after a single expansion step, there are no expansion rules to apply, and since $\Sigma = \varnothing$, no branching is possible according to LPB(Σ); so we conclude that $\Gamma_0 \nvdash_{\varnothing}^{\mathsf{KELB}} s$. The set Σ is then expanded to $\Sigma' = \{p\} \supset \Sigma$ so

$T\ p \vee q$	$T\ p \vee q$	$T\ p \vee q \quad \Sigma := \varnothing$
$T\ q \rightarrow p$	$T\ q \rightarrow p$	$T\ q \rightarrow p$
$T\ p \rightarrow (s \vee t)$	$T\ p \rightarrow (s \vee t)$	$T\ p \rightarrow (s \vee t)$
$T\ (p \wedge t) \rightarrow s$	$T\ (p \wedge t) \rightarrow s$	$T\ (p \wedge t) \rightarrow s$
$F\ p$	$T\ p$	$F\ s$
$T\ q$	$F\ s$	$F\ p \wedge t$
$T\ p$	$F\ p \wedge t$	$- \quad \Sigma := \{p\}$
$\times$	$T\ s \vee t$	/ \
	$F\ t$	$F\ p \quad T\ p$
	$T\ t$	$T\ q \quad T\ s \vee t$
	$\times$	$T\ p \quad T\ t$
		$\times \quad F\ t$
		$\times$

Fig. 4. An Example of KELB-Tableaux

as to unblock the tableau, and the proof continues in the logic $\vdash^{\mathsf{KELB}}_{\{p\}}$, in which both branches close, so we conclude that $\Gamma_0 \vdash^{\mathsf{KELB}}_{\{p\}} s$.

This example indicates how KELB-tableaux present us with an incremental method to prove theorems, in which one moves from proving theorems in one logic to the next without having to start from square 0 at each move. It also indicates that KELB-tableaux approximate classical logic "from below", that is, for $\varnothing \subseteq \Sigma' \subseteq \Sigma'' \subseteq \ldots \subseteq \mathcal{L}$ we have that

$$\vdash^{\mathsf{KELB}}_{\varnothing} \;\subseteq\; \vdash^{\mathsf{KELB}}_{\Sigma'} \;\subseteq\; \vdash^{\mathsf{KELB}}_{\Sigma''} \;\subseteq\; \ldots \;\subseteq\; \vdash^{\mathsf{KELB}}_{\mathcal{L}} \;=\; \vdash_{\mathrm{KE}}$$

where $\vdash_{\mathrm{KE}}$ is KE-tableau for classical logic. Note that this process is finite if only subformulas of the original formulas are added to Σ. This is indeed the case if we follow the *Branching Heuristics*, that is a heuristic for branching which tells us to branch on a formula α such that either $T\ \alpha$ or $F\ \alpha$ is an auxiliary premiss to an unexpanded main premiss in the branch; according to [DM94], this heuristics preserves classical completeness. Next section shows that $\vdash^{\mathsf{KELB}}_{\Sigma}$ is in fact correct and complete with respect to $\models^{\mathsf{LB}}_{\Sigma}$. But before that, we comment on $\vdash^{\mathsf{KELB}}_{\varnothing}$.

It is clear that $\Gamma \vdash^{\mathsf{KELB}}_{\varnothing} \alpha$ if the tableau can close without ever branching. That is, only linear inferences are allowed in $\vdash^{\mathsf{KELB}}_{\varnothing}$. Note that $\vdash_{\mathrm{BCP}}$-inferences are one of these linear inferences, and we have the following.

Lemma 1. *Let $\Gamma \cup \{\psi\}$ be a set of clauses. Then $\Gamma \vdash_{\mathrm{BCP}} \psi$ iff $\Gamma \vdash^{\mathsf{KELB}}_{\varnothing} \psi$.*

4 Soundness and Completeness

Let Θ be a branch in a KELB-tableau. We say that Θ is *open* if it is not closed. We say that Θ is *saturated* if the following conditions are met:

(a) If the premisses of a linear rule are in Θ, so are its consequences.

(b) If the main premiss of a two-premissed rule is in Θ, and the formula α corresponding to the auxiliary premiss is in Σ, then $T\ \alpha$ or $F\ \alpha$ is in Θ.

In classical KE-tableaux, the second condition for saturation does not impose the restriction $\alpha \in \Sigma$. We extend the notion of valuations to signed formulas in the obvious way: $v_\Sigma(T\alpha) = 1$ iff $v_\Sigma(\alpha) = 1$, $v_\Sigma(F\alpha) = 1$ iff $v_\Sigma(\alpha) = 0$ and $v_\Sigma(X\alpha) = \varepsilon$ iff $v_\Sigma(\alpha) = \varepsilon$. A valuation satisfy a branch in a tableau if it simultaneously satisfy all the signed formulas in the branch.

Lemma 2. *Consider the* KELB*-tableau expansion rules.*

i. If the premisses of a linear rule are satisfied by v_Σ*, so are its consequences.*
ii. If the conjugate of an auxiliary premiss of a two-premissed linear rule is satisfied by v_Σ*, so is the main premiss.*
iii. If the consequences of a linear rule are satisfied by v_Σ*, so is the main premiss.*
iv. If a branch is satisfied by v_Σ *prior to the application of LPB(*Σ*), then one of the two generated branches is satisfied after the application of LPB(*Σ*).*

Proof. (i)–(iii) are shown by a simple inspection on the linear rules in Figure 2 and $(F\wedge_{\alpha\alpha})$ and $(T\vee_{\alpha\alpha})$. As for (iv), suppose the branching occurs over the formula α, so $\alpha \in \Sigma$. By the Limited Bivalence Restrictions, $v_\Sigma(T\ \alpha) = 1$ or $v_\Sigma(F\ \alpha) = 1$, so v_Σ satisfies one of the two branches generated by the application of LPB(Σ).

Lemma 3. *Let* Θ *be an open saturated branch in a* KELB(Σ)*-tableau. Then* Θ *is satisfiable.*

Proof. Consider the propositional valuation v_Σ such that $v(p_i) = 1$ iff $T\ p_i \in \Theta$, $v(q_j) = 0$ iff $F\ q_j \in \Theta$ and $v(r_k) = \varepsilon_k$ otherwise. Clearly v_Σ is an LB(Σ)-valuation such that no two atoms are assigned to the same neutral truth value ε.

We prove by structural induction on α that for every $X\alpha \in \Theta$, $v_\Sigma(X\alpha) = 1$, $X \in \{T, F\}$. If α is atomic, $v_\Sigma(X\alpha) = 1$ follows from the definition of v_Σ.

If $X\alpha$ is the main premiss of a one-premissed rule R, by saturation we have both consequences of R in Θ. Then, by induction hypothesis, both such consequences are satisfied by v_Σ and by Lemma 2(iii) $v_\Sigma(X\alpha) = 1$.

If $X\alpha$ is the main premiss of a two-premissed rule; we have to analyse two cases. First, let $Y\beta$ be an auxiliary premiss for $X\alpha$ in a rule R, $Y \in \{T, F\}$, such that $Y\beta \in \Theta$, in which case R's conclusion is in Θ and, by Lemma 2(iii), $v_\Sigma(X\alpha) = 1$. Second, suppose no auxiliary premiss $Y\beta \in \Theta$, in which case there are two possibilities. If $\bar{Y}\beta \in \Theta$, where $\bar{Y}$ is Y's conjugate, by Lemma 2(ii) we obtain $v_\Sigma(X\alpha) = 1$; otherwise, by saturation, we know that all possible auxiliary premisses for $X\alpha$ are not in Σ; by saturation and rules $(F\wedge_{\alpha\alpha})$ and $(T\vee_{\alpha\alpha})$, we know that α's immediate subformulas are distinct, in which case v_Σ can assign distinct neutral values to them so as to satisfy α; that is, if $X\alpha = T\beta \vee \gamma$, make $v_\Sigma(\beta) = \varepsilon_i$, $v_\Sigma(\gamma) = \varepsilon_j \neq \varepsilon_i$ so that $v_\Sigma(T\beta \vee \gamma) = 1$, and similarly for $F\beta \wedge \gamma$ and $T\ \beta \rightarrow \gamma$. For the latter, the special case we where $\beta = \gamma$ is dealt by the semantic definition of $v_\Sigma(\beta \rightarrow \beta) = 1$. This finishes the proof.

$\mathsf{KELB}(\Sigma)$-tableaux have the *soundness* property if whenever a tableau for $\Gamma \vdash_{\Sigma}^{\mathsf{KELB}} \alpha$ closes then $\Gamma \models_{\Sigma}^{\mathsf{LB}} \alpha$. Conversely, the notion of *completeness* requires that if $\Gamma \models_{\Sigma}^{\mathsf{LB}} \alpha$ then there is a closed tableau for $\Gamma \vdash_{\Sigma}^{\mathsf{KELB}} \alpha$.

Theorem 1 (Soundness and Completeness). *$\Gamma \models_{\Sigma}^{\mathsf{LB}} \alpha$ iff $\Gamma \vdash_{\Sigma}^{\mathsf{KELB}} \alpha$.*

Proof. For soundness, we prove the contrapositive, that is, assume that $\Gamma \not\models_{\Sigma}^{\mathsf{LB}} \alpha$, so that there is a v_Σ such that $v_\Sigma(\Gamma) = 1$ and $v_\Sigma(\alpha) = 0$. If there is a $\mathsf{KELB}(\Sigma)$-tableau for $\Gamma \vdash_{\Sigma}^{\mathsf{KELB}} \alpha$, we have that all initial signed formulas of the tableau are satisfied by v_Σ. By Lemma 2(i) each use of a linear expansion rule generate formulas satisfied by v_Σ. By Lemma 2(iv), each application of LPB(Σ) generates a branch satisfied by v_Σ. If this tableau closes, this means that no such v_Σ could exist, which is a contradiction, so $\Gamma \not\vdash_{\Sigma}^{\mathsf{KELB}} \alpha$.

For completeness we also prove the contrapositive, so suppose that there is a $\mathsf{KELB}(\Sigma)$-tableau for $\Gamma \vdash_{\Sigma}^{\mathsf{KELB}} \alpha$ with an open saturated branch Θ. By Lemma 3 there is a valuation v_Σ that satisfies Θ, in particular $v_\Sigma(\Gamma) = 1$ and $v_\Sigma(\alpha) = 0$, and hence $\Gamma \not\models_{\Sigma}^{\mathsf{LB}} \alpha$.

Corollary 1. *The restriction of applications of LPB(Σ) to the Branching Heuristics preserves completeness of $\vdash_{\Sigma}^{\mathsf{KELB}}$.*

Proof. The Branching Heuristics allows only the branching over subformulas of formulas occurring in the tableau. This heuristics is actually suggested by the definition of a *saturated branch*, and aims at saturating a branch. It suffices to note that nowhere in the proofs of Lemma 3 and Theorem 1 was it required the branching over a non-subformula of a formula existing in a branch. Therefore, completeness holds for KELB-tableaux restricted to the Branching Heuristics.

The Branching Heuristics reduces the search space over which formula to branch, at the price of ruling out some small proofs of complex formulas obtained by clever branching.

The approximation family $\vdash_{\Sigma}^{\mathsf{KELB}}$ is not in the spirit of Dalal's approximation, but follows the paradigm of Cadoli and Schaerf [SC95,CS96], also applied by Massacci [Mas98b,Mas98a] and Finger and Wassermann [FW04].

5 Polynomial Approximations

As mentioned before, the family of inference relation $\vdash_{\Sigma}^{\mathsf{KELB}}$ does not follow Dalal's approach to approximation. We now present a family of logics based on $\vdash_{\Sigma}^{\mathsf{KELB}}$ that is closer to that approach.

For that, let $\mathbb{S} \subseteq 2^{\mathcal{P}}$ be a set of sets of atoms and, for every $\Pi \in \mathbb{S}$, let Π^+ be the closure of Π under formula formation. We define $\Gamma \vdash_{\mathbb{S}}^{\mathsf{KELB}} \alpha$ iff there exists a set $\Pi \in \mathbb{S}$ such that $\Gamma \vdash_{\Pi^+}^{\mathsf{KELB}} \alpha$. We define

$$\mathbb{S}_k = \{\Pi \subseteq \mathcal{P} |\ \ |\Pi| = k\}.$$

That is, $\mathbb{S}_k$ is a set of sets of atoms of size k. Note that if we restrict our attention to n atoms, $|\mathbb{S}_k| = \binom{n}{k} = O(n^k)$ sets of k atoms. For a fixed k, we only have to consider a polynomial number of sets of k atoms.

We then write $\vdash_k^{\mathsf{KELB}}$ to mean $\vdash_{\mathbb{S}_k}^{\mathsf{KELB}}$. In terms of theorem proving, the approximation process using the $\vdash_k^{\mathsf{KELB}}$ family performs an *iterative depth search* over KE-tableaux.

The entailment relation $\models_k^{\mathsf{LB}}$ can be defined in a similar way: $\Gamma \models_k^{\mathsf{LB}} \alpha$ iff $\Gamma \models_{\Pi+}^{\mathsf{LB}} \alpha$ for some $\Pi \in \mathbb{S}_k$. By Theorem 1, $\vdash_k^{\mathsf{KELB}}$ is sound and complete with respect to $\models_k^{\mathsf{LB}}$.

Lemma 4. *The family of inference systems* $\vdash_k^{\mathsf{KELB}}$ *is an approximation of classical logic "from below".*

Proof. It is obvious from the definition of $\vdash_k^{\mathsf{KELB}}$ that if $\Gamma \vdash_k^{\mathsf{KELB}} \alpha$ then $\Gamma \vdash_{k+1}^{\mathsf{KELB}} \alpha$, for all possible inference in the former are also possible in the latter. And for a given pair (Γ, α), we only need to consider the atoms occurring in them, so that $\vdash_{|atoms(\Gamma,\alpha)|}^{\mathsf{KELB}}$ is actually classical KE, so $\vdash_k^{\mathsf{KELB}}$ is an approximation of classical logic "from below".

It has been shown in [Fin04] that when Γ is a set of clauses and α is a clause, Dalal's $\vdash_k^{\mathrm{BCP}}$ inference is sound and complete with respect to $\models_k^{\mathsf{LB}}$. One important property of $\vdash_k^{\mathrm{BCP}}$ is that it can be decided in polynomial time. We now prove the same result for $\vdash_k^{\mathsf{KELB}}$.

Theorem 2. *The inference* $\Gamma \vdash_k^{\mathsf{KELB}} \alpha$ *can be decided in time polynomial with respect to* $n = |atoms(\Gamma, \alpha)|$.

Proof. For a fixed k, there are at most $O(n^k)$ subsets of $atoms(\Gamma, \alpha)$ with size k, in order to decide $\Gamma \vdash_k^{\mathsf{KELB}} \alpha$ we have to test only a polynomial number of inferences $\Gamma \vdash_{\Pi+}^{\mathsf{KELB}} \alpha$. The size of each such inference under the Branching Heuristics is a function of k, which is fixed, and does not depend on n, so the whole process of deciding $\Gamma \vdash_k^{\mathsf{KELB}} \alpha$ can be done in time $O(n^k)$.

The approximation $\vdash_k^{\mathsf{KELB}}$ performs an *iterated depth search* over the space of proofs. Comparatively, the KES_3 approximation process of [FW04], which does not guarantee polynomially decidable steps, performs a *depth-first search*.

6 Conclusion

We have created a family of tableau-based inferences systems $\vdash_0^{\mathsf{KELB}}, \vdash_1^{\mathsf{KELB}}, \ldots, \vdash_k^{\mathsf{KELB}}$ that approximates classical logic, such that each step has a sound and complete lattice-based semantics and can be decided in polynomial time.

Future work involves the implementation of such an approximation system and its practical application in areas such as Belief Revision and Planning. We hope to see how well it performs in "real" situations.

References

[CL73] C. Chang and R. Lee. *Symbolic Logic and Mechanical Theorem Proving.* Academic Press, London, 1973.

[CS96] Marco Cadoli and Marco Schaerf. The complexity of entailment in propositional multivalued logics. *Annals of Mathematics and Artificial Intelligence*, 18(1):29–50, 1996.

[CS00] Alessandra Carbone and Stephen Semmes. *A Graphic Apology for Symmetry and Implicitness.* Oxford Mathematical Monographs. Oxford University Press, 2000.

[D'A92] Marcello D'Agostino. Are tableaux an improvement on truth-tables? — cut-free proofs and bivalence. *Journal of Logic, Language and Information*, 1:235–252, 1992.

[D'A99] Marcello D'Agostino. Tableau methods foc classical propositional logic. In Marcello D'Agostino, Dov Gabbay, Rainer Haehnle, and Joachim Posegga, editors, *Handbook of Tableau Methods*, pages 45–124. Kluwer, 1999.

[Dal96a] Mukesh Dalal. Anytime families of tractable propositional reasoners. In *International Symposium of Artificial Intelligence and Mathematics AI/MATH-96*, pages 42–45, 1996.

[Dal96b] Mukesh Dalal. Semantics of an anytime family of reasponers. In *12th European Conference on Artificial Intelligence*, pages 360–364, 1996.

[DM94] Marcello D'Agostino and Marco Mondadori. "the taming of the cut. classical refutations with analytic cut". *Journal of Logic and Computation*, 4(285-319), 1994.

[Fin04] Marcelo Finger. Towards polynomial approximations of full propositional logic. Technical Report RT-2004-04, Department of Computer Science, IME/USP, 2004. Available for download at `www.ime.usp.br/~mfinger/publications`.

[FW04] Marcelo Finger and Renata Wassermann. Approximate and limited reasoning: Semantics, proof theory, expressivity and control. *Journal of Logic And Computation*, 14(2):179–204, 2004.

[GJ79] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* Freeman, 1979.

[Mas98a] Fabio Massacci. Anytime approximate modal reasoning. In Jack Mostow and Charles Rich, editors, *AAAI-98*, pages 274–279. AAAIP, 1998.

[Mas98b] Fabio Massacci. *Efficient Approximate Deduction and an Application to Computer Security.* PhD thesis, Dottorato in Ingegneria Informatica, Università di Roma I "La Sapienza", Dipartimento di Informatica e Sistemistica, June 1998.

[McA90] D. McAllester. Truth maintenance. In *Proceedings of the Eighth National Conference on Artificial Intelligence (AAAI-90)*, pages 1109–1116, 1990.

[SC95] Marco Schaerf and Marco Cadoli. Tractable reasoning via approximation. *Artificial Intelligence*, 74(2):249–310, 1995.

[Smu68] Raymond M. Smullyan. *First-Order Logic.* Springer-Verlag, 1968.