Resolution for mixed Post logic

Vladimir Komendantsky

Institute of Philosophy of Russian Academy of Science, Volkhonka 14, 119992 Moscow, Russia

vycom@pochtamt.ru

Abstract. In this paper we present a resolution method for many-valued mixed Post logic *PostL* introduced in [6]. This logic is not widely known, so we give the definition of its sequent calculus and define a 2-valued interpretation of many-valued sequents of *PostL*. It is a logic based on linear inequalities, its predicates are finitely-valued and take values from fixed partitions of rational numbers, the set of truth values is linearly ordered. *PostL* naturally generalizes the class of logics that occur in real-world applications. Compared to related approaches, our resolution method provides a more compact representation of many-valued logic formulas when a set of truth values is linearly ordered. We prove soundness and completeness of resolution for *PostL*.

1 Introduction

In this paper we consider the many-valued mixed Post logic *PostL* introduced in [6]. Mixed Post logic is a logic of many-sorted language, its predicates are of the kind $P^{a,b,c}$, where a, b, c are natural numbers. We provide a resolution rule for this logic, it is essential for the purpose of automated theorem proving. As far as *PostL* is a logic of finite-valued predicates based on inequalities, the resolution principle for this logic can be applied to computations in the systems of inequalities expressed in terms of *PostL*. The logic *PostL* is defined in Section 2, where we present a sequent calculus for it. The sequential definition is convenient to consider sets of formulas as finite disjunctions for the clausal resolution. We define internal and external languages. In Section 3 we give the translation function δ that maps many-valued formulas of *PostL* to their representations in 2-valued formulas of classical first-order logic. So *PostL* is definable in classical first-order logic (this result was proved in theorem 8.3.2 in [6]). In Section 3 we also give the proof of soundness and completeness of many-valued binary resolution on inequalities of *PostL*.

The calculus of *PostL* elegantly deals with multiple complex axiom systems. If one assume different degrees of truth to the predicates from different sciences then it will be possible to join a number of theories (which may contradict one another in the classical sense) inside a single axiom system.

Proceedings of the Seventh ESSLLI Student Session

Malvina Nissim (editor)

For example, consider arithmetics and hyperarithmetics. The statements of arithmetics are assigned maximal positive value and the statements of hyperarithmetics are assigned non-negative value from the set of truth values. So we can distinguish facts from different theories and get rid of contradictions inside the whole axiom system. Here we will use rational numbers from partitions of rationals as logical values for knowledge representation in complex systems of theories.

There exists a relative approach to many-valued resolution that has many features in common with the approach we present here. In [4] the notion of signed clause was introduced. A signed clause is a disjunction of many-valued literals labeled with non-empty sets of truth-values. It was pointed out there that signed clauses are independent of the many-valued logic they originated from. In fact, signed clauses do not contain any many-valued connective and are simply generic language for denoting many-valued interpretations. This approach to many-valued resolution was exploited in [1], [5] and [7]. The approach of linear inequalities (cf. [6]) we used in this paper is very close to the approach of signed formulas, except that the inequalities approach is a very natural formalization of linear orderings defined on the sets of truth-values.

2 Mixed Post logic

The logic $PostL$ has two levels. At the first level the formulas of many-valued logic are considered. The set of truth values and an interpretation of logical connectives are defined in terms of internal signature. The object variables are many-sorted. At the second level there is a predicate of comparison with zero for internal formulas. External language doesn't contain function symbols. External object variables are internal object variables. So formulas of this level are formulas of two-valued logic.

Definition 1 (Internal syntax). An *internal language* $\mathbf{L}^i$ contains the set $\boldsymbol{L}^i = \{\pm k/c \mid k \in [a,b), \quad a,b,k,c \in \mathbf{Z}^+, \quad a < b, \quad c > 0\}$ of of logical constants for the non-empty set $\boldsymbol{P}^i$ of predicate symbols indexed with positive integers a, b, c, the set $\boldsymbol{O}^i$ of object variables, the set $\boldsymbol{F}^i$ of function symbols, the set of logical connectives $\boldsymbol{C}^i = \{\neg, \vee, \&, \supset, \equiv\}$ and the set of quantifiers $\boldsymbol{Q}^i = \{\forall, \exists\}$. An *internal term* t is made up of function symbols from $\boldsymbol{F}^i$ and object variables from $\boldsymbol{O}^i$. An *internal atom* $P^{a,b,c}(t_1, \dots, t_n)$ contains internal terms and predicate symbols from $\boldsymbol{P}^i$. An *internal predicate formula* is an internal atom or it is made up of atoms with quantifiers from $\boldsymbol{Q}^i$ and logical connectives from $\boldsymbol{C}^i$.

We will say that formula A is of the sort a, b, c if its valuation is in this sort. Formula A is build upon the set of sorts S if sorts of all variables in A are all in S.

Definition 2 (External syntax). An *external language* $\mathbf{L}^e$ contains $\boldsymbol{C}^e = \{\neg, \vee, \&, \supset, \equiv, ?:\}$, where $?:$ is a ternary conditional connective "if-then-else", $\boldsymbol{Q}^e = \{\forall, \exists\}$ and single predicate symbol of comparison with zero. An *inequality* of $PostL$ is an expression of the kind $(0 \leq A)$, where A is an internal predicate formula, or an expression $A_1 + A_2$, where A_1 and A_2 are internal predicate formulas. *External predicate formulas* of $PostL$ are formulas made up from inequalities of $PostL$ with connectives from $\boldsymbol{C}^e$ and quantifiers from $\boldsymbol{Q}^e$.

For the formulas of the kind $\neg(0 \leq A)$ will be used compressed notation $(0 < \neg A)$. So it will be possible to notate formulas as $(0 \prec A)$, where $\prec \in \{\leq, <\}$.

Definition 3 (Sequent). A *sequent* of $PostL$ is an expression of the kind $\rightarrow \Delta$, where Δ is a finite set of external predicate formulas. An interpretation of a sequent $\rightarrow \Delta$ is a disjunction of predicate formulas in Δ. An interpretation of the sequent with empty Δ is not valid. Instead of speaking that an interpretation of a sequent is valid we will say that a sequent is valid itself. More precisely, a sequent is valid if its interpretation is valid under every propositional and object variable assignment.

Definition 4. *Axiom schemata* of sequent calculus of two-valued logic are $(\rightarrow \Delta\ A\ \neg A)$, $(\rightarrow \Delta\ \top)$ and $(\rightarrow \Delta\ \neg\bot)$, where $\top$ means *truth* and $\bot$ means *false*.

External inference rules are following:

$$\frac{\rightarrow \Delta_1\ A\ B\ \Delta_2}{\rightarrow \Delta_1\ B\ A\ \Delta_2}\ (Split) \qquad \frac{\rightarrow \Delta\ A}{\rightarrow \Delta\ \neg\neg A}\ (\neg\neg)$$

$$\frac{\rightarrow \Delta\ A \quad \rightarrow \Delta\ B}{\rightarrow \Delta\ A \& B}\ (\&) \qquad \frac{\rightarrow \Delta\ \neg A\ \neg B}{\rightarrow \Delta\ \neg(A \& B)}\ (\neg\&)$$

$$\frac{\rightarrow \Delta\ A\ B}{\rightarrow \Delta\ A \vee B}\ (\vee) \qquad \frac{\rightarrow \Delta\ \neg A \quad \rightarrow \Delta\ \neg B}{\rightarrow \Delta\ \neg(A \vee B)}\ (\neg\vee)$$

$$\frac{\rightarrow \Delta\ \neg A, B \quad \rightarrow \Delta\ \neg B, A}{\rightarrow \Delta\ A \equiv B}\ (\equiv) \qquad \frac{\rightarrow \Delta\ A\ B \quad \rightarrow \Delta\ \neg B\ \neg A}{\rightarrow \Delta\ \neg(A \equiv B)}\ (\neg \equiv)$$

$$\frac{\rightarrow \Delta\ A(y \setminus x)}{\rightarrow \Delta\ \forall x A}\ (\forall) \qquad \frac{\rightarrow \Delta\ \neg A(t \setminus x)\ \neg\forall x A}{\rightarrow \Delta\ \neg\forall x A}\ (\neg\forall)$$

$$\frac{\rightarrow \Delta\ A(t \setminus x)\ \exists x A}{\rightarrow \Delta\ \exists x A}\ (\exists) \qquad \frac{\rightarrow \Delta\ \neg A(y \setminus x)}{\rightarrow \Delta\ \neg\exists x A}\ (\neg\exists)$$

where $A(t \backslash x)$ means that term t is substituted in atoms A instead of variable x; object variable y and term t are free for substitution in A in place of x, variable x has not free occurrences in the conclusion of the rules $(\exists)$ and $(\neg\forall)$. Implication $A \supset B$ is an abbreviation for $\neg A \vee B$.

Interior rules are following, expressions in square brackets may be absent simultaneously:

$$\frac{\to \Delta(0 \leq A + B)}{\to \Delta(0 \leq B + A)}\ (\leq Split)$$

$$\frac{\to \Delta(0 \leq A[+B])}{\to \Delta(0 \leq \neg\neg A[+B])}\ (\leq \neg\neg)$$

$$\frac{\to \Delta(0 \leq A[+C]) \quad \to \Delta(0 \leq B[+C])}{\to \Delta(0 \leq (A\&B)[+C])}\ (\leq \&)$$

$$\frac{\to \Delta(0 \leq \neg A[+C])(0 \leq \neg B[+C])}{\to \Delta(0 \leq \neg(A\&B)[+C])}\ (\leq \neg\&)$$

$$\frac{\to \Delta(0 \leq A[+C])(0 \leq B[+C])}{\to \Delta(0 \leq (A \vee B)[+C])}\ (\leq \vee)$$

$$\frac{\to \Delta(0 \leq \neg A[+C]) \quad \to \Delta(0 \leq \neg B[+C])}{\to \Delta(0 \leq \neg(A \vee B)[+C])}\ (\leq \neg\vee)$$

$$\frac{\to \Delta(0 \leq A(y \setminus x)[+B])}{\to \Delta(0 \leq \forall x A[+B])}\ (\leq \forall)$$

$$\frac{\to \Delta(0 \leq \neg A(t \setminus x)[+B])(0 \leq \neg\forall x A[+B])}{\to \Delta(0 \leq \neg\forall x A[+B])}\ (\leq \neg\forall)$$

$$\frac{\to \Delta(0 \leq A(t \setminus x)[+B])(0 \leq \exists x A[+B])}{\to \Delta(0 \leq \exists x A[+B])}\ (\leq \exists)$$

$$\frac{\to \Delta(0 \leq \neg A(y \setminus x)[+B])}{\to \Delta(0 \leq \neg\exists x A[+B])}\ (\leq \neg\exists)$$

all rules above are valid if $\leq$ is replaced with $<$. Quantificational rules are restricted in the same kind as external rules, variable y has not free occurrences in the conclusions of $(\prec \exists)$ and $(\prec \neg\forall)$.

In addition to above rules we define rules for conditional statement:

$$\frac{\to \Delta(0 \leq A[+E]^2)(0 \leq [\neg]^1 C[+D]^3) | \to \Delta(0 < \neg(A[+\neg E]^2)(0 \leq [\neg]^1 B[+D]^3)}{\to \Delta(0 \leq ([\neg]^1 A[+E]^2)?B : C[+D^3])}$$

$$(\leq ? :)$$

$$\frac{\to \Delta(0 \leq A[+E]^2)(0 < [\neg]^1 C[+D]^3) | \to \Delta(0 < \neg A[+\neg E]^2)(0 < [\neg]^1 B[+D]^3)}{\to \Delta(0 < ([\neg]^1 A[+E]^2)?B : C[+D^3])}$$

$$(< ? :)$$

where expressions in square brackets with the same index can be dropped simultaneously.

Conditional statement lets us introduce Post cyclic negation into the language of $PostL$. For any internal formula A which takes n different values from a set $\{a_1, \ldots, a_n\}$, where $a_1 < \cdots < a_n$, we define Post negation $\neg_{Post} A$ as an abbreviation for

$$(\neg A + a_1 ? a_n : (\neg A + a_2 ? a_1 : \quad \ldots \quad (\neg A + a_{n-1} ? a_{n-2} : a_{n-1}) \ldots)).$$

3 Resolution

We adopt the concepts of resolution method from [3]. For the resolution in many-valued logic see cf. [2].

Now the translation function δ will be defined. The translation δ maps sequents and external predicate formulas of $PostL$ to sequents and formulas of two-valued logic respectively, i.e. δ assigns interpretations to sequents of $PostL$. Correctness of 2-valued interpretation δ introduced in [6] on page 165 is based on the theorem 7.4.1 and lemma 8.3.2 from [6].

$$
\begin{aligned}
&\delta(A\&B) \rightleftharpoons \delta(A)\&\delta(B),\\
&\delta(A \vee B) \rightleftharpoons \delta(A) \vee \delta(B),\\
&\delta(\neg A) \rightleftharpoons \neg\delta(A),\\
&\delta(\forall x A) \rightleftharpoons \forall x \delta(A),\\
&\delta(\exists x A) \rightleftharpoons \exists x \delta(A),\\
&\delta(0 \prec (A\&B)[+C]) \rightleftharpoons \delta(0 \prec A[+C])\&\delta(0 \prec B[+C]),\\
&\delta(0 \prec (A \vee B)[+C]) \rightleftharpoons \delta(0 \prec A[+C]) \vee \delta(0 \prec B[+C]),\\
&\delta(0 \prec \neg(A\&B)[+C]) \rightleftharpoons \delta(0 \prec \neg A[+C]) \vee \delta(0 \prec \neg B[+C]),\\
&\delta(0 \prec \neg(A \vee B)[+C]) \rightleftharpoons \delta(0 \prec \neg A[+C])\&\delta(0 \prec \neg B[+C]),\\
&\delta(0 \prec (A?B : C)[+D]) \rightleftharpoons \delta(0 \prec (\neg A \vee B)[+D])\&\delta((A \vee C)[+D])\\
&\delta(0 \prec \neg(A?B : C)[+D]) \rightleftharpoons \delta(0 \prec (\neg A \vee \neg B)[+D])\&\delta((A \vee \neg C)[+D])\\
&\delta(0 \prec \forall x A[+C]) \rightleftharpoons \forall y \delta(0 \prec A(y \setminus x)[+C]),\\
&\delta(0 \prec \neg\forall x A[+C]) \rightleftharpoons \exists y \delta(0 \prec \neg A(y \setminus x)[+C]),\\
&\delta(0 \prec \exists x A[+C]) \rightleftharpoons \exists y \delta(0 \prec A(y \setminus x)[+C]),\\
&\delta(0 \prec \neg\exists x A[+C]) \rightleftharpoons \forall y \delta(0 \prec \neg A(y \setminus x)[+C]),
\end{aligned}
$$

where in the last four lines y is neither in A nor in C.

$$\delta(0 \prec P + C) \rightleftharpoons \delta(0 \prec C + P),$$

where P is negated or unnegated atom.

Using rules for translation δ one can for any $PostL$-formula build a skolemized CNF $(L_{1,1} \vee \cdots \vee L_{1,n_1})\& \ldots \&(L_{m,1} \vee \cdots \vee L_{m,n_m})$, where each of $L_{i,j} \quad (1 \leq i \leq m, 1 \leq j \leq n_i)$, is a literal of the form $\delta(0 \prec A[+B])$, A and B are negated or unnegated atoms. Formula $\delta(0 \prec A[+B])$ is two-valued, it is valid iff the inequality $(0 \prec A[+B])$ is valid. The method of constructing of such formulas $\delta(0 \prec A[+B])$ is following. Any inequality with quantifiers eliminated to the external level containing only internal atoms is replaced by equivalent formula with finite set of pairwise different two-valued atoms. For any given atom of the sort a, b, c it would be enough $b - a$ pairwise different two-valued atoms each of which is assigned a value from $\{-1/c, 1/c\}$. Then rational values of two-valued formulas are summarized. To prove the next results it is convenient to use only 2-valued atoms with values from $\{-1, 1\}$ instead of $\{-1/c, 1/c\}$ (we can multiply a linear combination of atoms by any integer). Since all sorts are finite it is possible to build a classical CNF for any given $PostL$-formula. As far as classical CNF is constructed it has to be skolemized by eliminating quantifiers from the external level.

Since for any $PostL$-formula there exists a skolemized classical CNF (we will also call it a set of clauses of $PostL$), the resolution rule schema is straightforward. Here we will assume that any 2-valued literal $(0 \prec A[+B])$, possibly with negation, occurring in a clause of $PostL$ is a skolemized formula resulted by finite number of applications of δ to some $PostL$-formula, A and B are negated or unnegated internal atoms.

Definition 5 (Resolution). Let Δ_1 and Δ_2 be sets of clauses containing skolemized literals of the form $(0 \prec A[+B])$, then *binary resolution* is the rule

$$\frac{\Delta_1 \cap \{(0 \prec A_1 + B)\} \qquad \Delta_2 \cap \{(0 \prec \neg A_2 + C)\}}{(\Delta_1 \cap \Delta_2 \cap \{(0 \prec B + C)\})\sigma},$$

where the sign $\prec$ in the conclusion is $\leq$ iff $\prec$ is $\leq$ in both premises; σ is the most common unifier of the formulas A_1 and A_2. B and C are atoms, possibly empty. In the case if both of them are empty we have the disjunct $(0 \prec \emptyset)$.

Together with binary resolution we admit to use ($\prec$ *Split*) rule that allows to rearrange internal formulas in premises. External formulas in clauses can be arbitrarily rearranged since we consider only unordered resolution. Also there is *a rule of contraction*:

$$\frac{(0 \prec A + A)}{(0 \prec A)}$$

.

Lemma 1 (Ground clauses). *Let Φ be an arbitrary set of ground clauses of PostL. If Φ is unsatisfiable then the empty clause $(0 \prec \emptyset)$ is derivable from Φ by finite number of applications of binary resolution.*

Proof. The case when $(0 \prec \emptyset) \in \Phi$ is trivial. Therefore let us assume that $(0 \prec \emptyset) \notin \Phi$. Since all the clauses in Φ are ground, the Herbrand basis of Φ coincides with the set of all literals of Φ. The lemma will be proved by induction on the difference $diff(\Phi)$ between the total number of literals in Φ and the total number of clauses in Φ. There are two cases:

1. $diff(\Phi) = 0$. In this case the number of literals equals to the number of disjuncts. Since $(0 \prec \emptyset) \notin \Phi)$, Φ consists of one-element disjuncts only. Let us construct a sequence of sets $\Phi_0 \subseteq \Phi_1 \subseteq \cdots \subseteq \Phi_n$ such that

$\Phi_0 = \Phi$, and for each $i \geq 1$ Φ_i is closed under the ($\prec$ *Split*) and contraction rules; Φ_i contains Φ_{i-1}, and also all immediate consequences by binary resolution from Φ_{i-1}. We will show that using the procedure explained above, any unsatisfiable set of disjuncts can be extended to the set which contains the empty disjunct. Any unsatisfiable set of disjuncts contains a subset $\Lambda = \{(0 \prec A[+B]), (0 \prec \neg A[+\neg B])\}$. Let Λ is obtained at the i-th step of extension. Then $(0 \prec \emptyset)$ can be inferred at most at the step $i+3$. In fact, $\Lambda \subseteq \Phi_i$, $\{(0 \prec A+\neg A), (0 \prec B+\neg B)\} \subseteq \Phi_{i+1}$, $\{(0 \prec A+A), (0 \prec \neg A+\neg A), (0 \prec A), (0 \prec \neg A)\} \subseteq \Phi_{i+2}$. Thus, $\{(0 \prec \emptyset)\} \subseteq \Phi_{i+3}$.

2. $diff(\Phi) > 0$. Φ contains disjuncts having more than one element. Let Ψ be a non-empty proper subset of Φ such that $\Phi - \Psi$ is consistent. Suppose that there are no such disjuncts in $\Phi - \Psi$ which can be used to infer the empty disjunct by a finite number of applications of binary resolution, the rule ($\prec$ *Split*), and the contraction rule to the set Ψ. Then there is a Herbrand interpretation in which both $\Phi - \Psi$ and Ψ are satisfied. We have a contradiction with the unsatisfiability of Φ. Hence, $(0 \prec \emptyset)$ can be inferred from Φ.

□

For the proof of this lemma for the case of signed many-valued clauses see cf. [7], where signed negative hyperresolution was considered.

Lemma 2 (Lifting Lemma). *If Δ'_1 and Δ'_2 are instances of clauses Δ_1 and Δ_2 respectively, and if Δ' is a resolvent of Δ'_1 and Δ'_2, then there is a resolvent Δ of Δ_1 and Δ_2 such that Δ' is an instance of Δ.*

Proof. The proof of this lemma is the proof of the classical lifting lemma (cf. [3]), where the notion of many-valued resolvent is taken into account. □

Theorem 1 (Completeness and soundness). *The set Φ of clauses of PostL is unsatisfiable iff the empty clause $(0 \prec \emptyset)$ is derivable from Φ by finite number of applications of binary resolution rule.*

Proof. ($\Rightarrow$) It follows from Lemma 1 and Lemma 2 exactly as in the case of classical logic. Since Φ is unsatisfiable then by Herbrand theorem (cf. [3]) there is a finite unsatisfiable set Φ' of ground clauses from Φ. By Lemma 1 there is a proof D' of $(0 \prec \emptyset)$ from Φ'. The proof D' can be transformed by Lemma 2 to the proof D of $(0 \prec \emptyset)$ from Φ.

($\Leftarrow$) It follows from the fact that a Herbrand interpretation that satisfies a set of clauses also satisfies all their resolvents. But no Herbrand interpretation satisfies the empty clause. □

As an easy corollary of Theorem 1, Theorem 8.3.1 and Theorem 6.1.2 (which states that satisfiability problem of a system of linear inequalities with variables defined on rational numbers is in PTIME) from [6] we get

Theorem 2. *The satisfiability problem in $PostL$ is NP-complete.*

4 Conclusion

In the present paper we considered only single logic defined in [6]. But the resolution principle presented here can be applied with slight remarks to the whole class of logics of finite-valued predicates based on inequalities, such as pluralistic logic of n-dimensional sequences and mixed Lukasiewicz's logic.

Further investigations must show how much the many-valued resolution on inequalities differs from the signed many-valued resolution introduced in [2], [4] and [7]. The complexity of resolution with respect to the method of computing the clause normal form must be precisely defined. It seems that linear ordered sets of truth values are treated very effectively with our resolution, in addition, we get the ability of automated reasoning in hypertheories which contain multiple various theories and protected from contradictory propositions. One of the most important topics that also must be studied is the representation of cyclic Post negation in $PostL$. In the present text it is represented by conditional operator "if-then-else", but in this case we might have a sufficient time and space expenditures. The natural formalization of Post negation with respect to many-sorted origin of $PostL$ is necessary.

Bibliography

[1] S. Aguzzoli and A. Ciabattoni. Finiteness in infinite-valued logic. *Journal of Logic, Language and Information*, 9(1):5–29, 2000.

[2] M. Baaz and C. G. Fermüller. Resolution based theorem proving for many-valued logics. *Journal of Symbolic Computation*, 19(4):353–391, 1995.

[3] C. Chang and R. C. Lee. *Symbolic logic and mechanical theorem proving.* Academic Press, 1973.

[4] R. Hähnle. *Automated theorem proving in multiple-valued logics.* International Series of Monographs on Computer Science. Oxford University Press, 1993.

[5] R. Hähnle. Commodious axiomatization of quantifiers in multiple-valued logic. In *Proc. 26th International Symposium on Multiple-Valued Logics*, pages 118–123, Los Alamitos, May 1996. IEEE Press.

[6] N. K. Kosovsky and A. V. Tishkov. *Logics of finite-valued predicates based on inequalities.* St. Petersburg University Press, 2000. In Russian.

[7] V. Sofronie-Stokkermans. Automated theorem proving by resolution for finitely-valued logics based on distributive lattices with operators. *Multiple-Valued Logic - An International Journal*, 5(4):289–344, 2001.