International Journal of Computational Intelligence Systems, Vol. 7, No. 3 (June 2014), 418-431

α-Quasi-Lock Semantic Resolution Method Based on Lattice-Valued Logic

Xiaomei Zhong, Yang Xu
School of Mathematics, Southwest Jiaotong University, Chengdu, Sichuan 610031, P.R. China
E-mail: zhongxm2013@home.swjtu.edu.cn, xuyang@home.swjtu.edu.cn

Jun Liu, Shuwei Chen
School of Computing and Mathematics, University of Ulster, Northern Ireland, UK
E-mail: j.liu@ulster.ac.uk, chensw915@gmail.com

Received 18 April 2012

Accepted 30 April 2012

Abstract

Based on the general form of α-resolution principle for a lattice-valued logic with truth-values defined in a lattice-valued logical algebra structure - lattice implication algebra, the further extended α-resolution method in this lattice-valued logic is discussed in the present paper in order to increase the efficiency of the resolution method. Firstly, α-quasi-lock semantic resolution method in lattice-valued propositional logic LP(X) is established by combining the lock and semantic resolution simultaneously, and its theorems of soundness and conditional completeness are proved. Secondly, this α-quasi-lock semantic resolution method is extended into the corresponding lattice-valued first-order logic LF(X), and its soundness and conditional completeness are also established. This extended resolution method will provide a theoretical basis for automated soft theorem proving and program verification based on lattice-valued logic.

Keywords: α-Quasi-lock semantic resolution method; resolution-based automated reasoning; general form of α-resolution principle; lattice-valued logic; lattice implication algebra.

1. Introduction

Since resolution principle based on classical logic was proposed by Robinson in 1965 [1], resolution-based automated reasoning has been widely applied to various areas, such as mathematics, biology, engineering technologies. The study about resolution-based automated reasoning methods in classical logic attracted a lot of researchers' interest and some important results about variation or extension of resolution principle have been achieved, such as, in 1965, Wos *et al*. [2] proposed the resolution strategy based on support sets, and its soundness and completeness were also obtained. In 1967, Slagle [3] established semantic resolution method and proved its soundness and completeness. Afterwards, Loveland [4] and Luckham [5] proposed linear resolution method by their respective views. In order to select the unique resolution literal during the process of resolution, Reiter [6] established ordered resolution method and ordered semantic resolution method, but the latter did not have completeness, even for ground clause sets. In order to improve the efficiency of resolution-based automated reasoning, in 1971, Boyer proposed lock resolution method in his doctoral thesis at the University of Texas, and proved its soundness and completeness theorems. In 1981, Huang [7] improved linear resolution method and established MOL resolution method, its soundness and completeness theorems were also proved. In 1979, 1985, 1987 and 1992, Liu [8-11] did in-depth research on the compatibility among semantic resolution method, linear resolution method and lock resolution method, as well as the corresponding soundness and completeness. According to the improvement of semantic resolution method, Lu *et al*. [12] proposed colored resolution method and also proved its soundness and completeness theorems. In 2005, Cai [13] discussed the realization of the resolution method based on the strategy of support sets. In 2007, Meng *et al*. [14] improved the efficiency of resolution by checking the correlation of symbols in clauses.

From the above short review, in classical logic, there are mainly three kinds of resolution-based automated reasoning methods, i.e., lock resolution method, semantic resolution method and linear resolution method. Lock resolution method improves the efficiency of automated reasoning through limiting resolution literals to each literal equipped with a lock and implement resolution by the smallest lock rule. To some extent, this method can limit the generation of redundant clauses during the process of resolution and improve the efficiency of automated reasoning. Semantic resolution method and linear resolution method improve the efficiency of automated reasoning by limiting resolution clauses, which is to say that they limit resolution clauses by certain ways respectively, so as to reduce the number of redundant clauses occurring in the process of resolution and improve the efficiency of automated reasoning. In other words, these three kinds of resolution-based automated reasoning methods improve the efficiency of automated reasoning from two different views. If we can establish another method containing the benefits of the above three kinds of resolution-based automated reasoning methods, i.e., this new method reduces the generation of redundant clauses by limiting resolution clauses and literals simultaneously, then we can further improve the efficiency of resolution-based automated reasoning to some extent.

In another aspect, in the real world, people living in the environment with much uncertainty often need to make judgment with uncertainty ("soft conclusion") based on uncertain environment, information with uncertainty ("soft premise") and knowledge with uncertainty ("soft rules"). We call this "soft causal relationship" that "soft premise" and "soft rules" draw "soft conclusion" as a "soft theorem". People often discover such "soft theorems", and also need to verify their rationality (or correctness) through practice or methods. Non-classical logic has been a considerably useful formal tool for computer science and AI during the past decade. Many-valued logic is a powerful extension and development of classical logic, which aims to establish the logical foundation for "soft" information processing. Lattice-valued logic, as one of the most important many-valued logics, extends the chain-type truth-valued field to a general lattice in which the truth-values are incompletely comparable with each other. Lattice-valued logic is thus an important and promising research direction that provides an alternative logical approach to dealing with imprecision and incomparability as well [15]. As the automated reasoning method based on resolution principle for classical logic is an important class of automated reasoning methods in the field of "theorem machine proving", in order to make machines can simulate people verifying these "soft theorems", i.e., make machines automatically verify these "soft theorems" by reasoning, similar to the academic thinking of "theorem machine proving", it is very important for us to establish an appropriate resolution principle in non-classical logics including many-valued logics even lattice-valued logics and some effective resolution methods based on them. With the progress of society, more and more uncertain information needs to be handled in the real world.

Taking the above ideas into consideration, the resolution principle based on lattice-valued logic with truth-value in a lattice-valued logical algebraic structure - lattice implication algebras was established by Xu *et al*. [16-17], which can be used to prove whether a lattice-valued logical formula is false at a truth-value level α (i.e., α-false) or not in order to characterize incomparability and fuzziness. After that, some researchers did further research on the theory of resolution-based automated reasoning for the above lattice-valued logic and obtained some important results. For example, in 2007, Xu *et al*. [18, 19] discussed the relation between α-resolution for lattice-valued propositional logic LP(X) and that for lattice-valued first-order logic LF(X), and pointed out the fact that α-resolution for LF(X) can be equivalently transformed into that for LP(X). As an application of α-resolution principle, Xu *et al*. [20] studied α-resolution-based automated reasoning for linguistic truth-valued lattice-valued propositional logic $\mathscr{L}_{V(n\times 2)}$P(X). In 2008, Li [21] obtained some properties of α-resolution fields and filter-resolution fields in lattice-valued propositional logic LP(X) respectively, as well as the relation between α-resolution and filter-resolution in linguistic truth-valued lattice-valued propositional logic $\mathscr{L}_{V(n\times 2)}$P(X). In 2010, He *et al*. [22] proposed α-lock resolution method in lattice-valued propositional logic LP(X) and established its soundness and weak completeness. To further improve the efficiency of α-resolution-based automated reasoning in lattice-valued logic, in 2010, Xu *et al*. [23] proposed the general form of α-resolution principle in lattice-valued logic with truth-value in lattice implication algebras and proved its soundness and weak completeness theorems. In the same year, Xu *et al*. [24] proposed α-generalized resolution principle based on lattice-valued propositional logic LP(X), and its soundness and weak completeness were also established.

As a continuation of the above research work, on the basis of lock resolution method and semantic resolution method in classical logic, this paper will establish a lock semantic resolution method with features of both lock resolution method and semantic resolution method for lattice-valued logic based on lattice implication algebras, which limits the generation of redundant clauses during the process of resolution-based automated reasoning by limiting resolution clauses and literals simultaneously.

This paper is organized as follows: in Section 2, some preliminary relevant concepts and conclusions about lattice-valued logic and the general form of α-resolution principle are reviewed. In Section 3, α-quasi-lock semantic resolution method based on lattice-valued propositional logic LP(X) is established, and its soundness and weak completeness are also obtained; In Section 4, this α-quasi-lock semantic resolution method is extended into the corresponding lattice-valued first-order logic LF(X) and its soundness theorem, lifting lemma and weak completeness theorem are also proved.

2. Preliminaries

In the following, we will introduce some elementary concepts and conclusions of lattice-valued logic with truth-value in lattice implication algebra and the general form of α-resolution principle. We refer the readers to [15, 23] for more details.

2.1. *Lattice implication algebra*

Definition 1. [15] *Let* $(L, \vee, \wedge, O, I)$ *be a bounded lattice with an order-reversing involution* $'$, *I and O the greatest and the smallest element of L respectively, and* $\rightarrow: L \times L \rightarrow L$ *be a mapping.* $(L, \vee, \wedge, ', \rightarrow, O, I)$ *is called a lattice implication algebra (LIA) if the following conditions hold for any* $x, y, z \in L$:

(I_1) $x \rightarrow (y \rightarrow z) = y \rightarrow (x \rightarrow z)$,

(I_2) $x \rightarrow x = I$,

(I_3) $x \rightarrow y = y' \rightarrow x'$,

(I_4) $x \rightarrow y = y \rightarrow x = I$ *implies* $x = y$,

(I_5) $(x \rightarrow y) \rightarrow y = (y \rightarrow x) \rightarrow x$,

(l_1) $(x \vee y) \rightarrow z = (x \rightarrow z) \wedge (y \rightarrow z)$,

(l_2) $(x \wedge y) \rightarrow z = (x \rightarrow z) \vee (y \rightarrow z)$.

Example 1. [15] **(Łukasiewicz implication algebra on finite chain)** *Let* $L_n = \{a_i \mid i = 1, 2, \ldots, n\}$, $a_1 < a_2 < \ldots < a_n$. *For any* $1 \leq j, k \leq n$, *define*

$a_j \vee a_k = a_{max\{j, k\}}$,

$a_j \wedge a_k = a_{min\{j, k\}}$,

$(a_j)' = a_{n-j+1}$,

$a_j \rightarrow a_k = a_{min\{n-j+k, n\}}$.

Then $(L_n, \vee, \wedge, ', \rightarrow, a_1, a_n)$ *is a LIA.*

Example 2. [25] *Let* $\mathscr{L}_n = (L_n, \vee_1, \wedge_1, '^1, \rightarrow_1, a_1, a_n)$ *be the Łukasiewicz implication algebra in Example 2.1.* $L_2 = \{b_1, b_2\}$, $b_1 < b_2$, $\mathscr{L}_2 = (L_2, \vee_2, \wedge_2, '^2, \rightarrow_2, b_1, b_2)$ *is also a Łukasiewicz implication algebra. For any* (a_i, b_j), $(a_k, b_m) \in L_n \times L_2$, *define*

$(a_i, b_j) \vee (a_k, b_m) = (a_i \vee_1 a_k, b_j \vee_2 b_m)$,

$(a_i, b_j) \wedge (a_k, b_m) = (a_i \wedge_1 a_k, b_j \wedge_2 b_m)$,

$(a_i, b_j)' = (a_i'^1, b_j'^2)$,

$(a_i, b_j) \rightarrow (a_k, b_m) = (a_i \rightarrow_1 a_k, b_j \rightarrow_2 b_m)$.

Then $(L_n \times L_2, \vee, \wedge, ', \rightarrow, (a_1, b_1), (a_n, b_2))$ *is a LIA, denoted as* $\mathscr{L}_n \times \mathscr{L}_2$.

2.2. *Lattice-valued propositional logic LP(X)*

Definition 2. [26] *Let X be the set of propositional variables,* $(L, \vee, \wedge, ', \rightarrow, O, I)$ *be a LIA,* $T = L \cup \{', \rightarrow\}$ *be a type with* $ar(') = 1$, $ar(\rightarrow) = 2$ *and* $ar(a) = 0$ *for any* $a \in L$. *The proposition algebra of the lattice-valued proposition calculus on the set X of propositional* *variables is the free T algebra on X and denoted by LP(X).*

Definition 3. [26] *The set $\mathcal{F}$ of formulas of lattice-valued propositional logic LP(X) is the least set Y satisfying the following conditions*:

(1) $X \subseteq Y$,

(2) $L \subseteq Y$,

(3) *if* $p, q \in Y$, *then* $'(p), \rightarrow (p, q) \in Y$,

where X is the set of propositional variables, L is the set of constants.

In the following, we denote $'(p)$ as p' and $\rightarrow (p, q)$ as $p \rightarrow q$.

Definition 4. [26] *A mapping v: LP(X) → L is called a valuation of lattice-valued propositional logic LP(X), if it is a T-homomorphism*.

Definition 5. [27] *Let $G \in \mathcal{F}$ and $\alpha \in L$. If $v(G) \leq \alpha$ for any valuation v of lattice-valued propositional logic LP(X), we say G is always less than or equal to α (or G is α-false), denoted by $G \leq \alpha$.*

Definition 6. [15] *A lattice-valued propositional logical formula G in lattice-valued propositional logic system LP(X) is called an extremely simple form, in short ESF, if a lattice-valued propositional logical formula G^* obtained by deleting any constant or literal or implication term occurring in G is not equivalent to G.*

Definition 7. [15] *A lattice-valued propositional logical formula G in lattice-valued propositional logic system LP(X) is called an indecomposable extremely simple form, in short IESF, if the following two conditions hold:*

(1) *G is an ESF containing connectives → and ′ at most,*

(2) *for any $H \in \mathcal{F}$, if $\overline{H} \in \overline{G}$ in $\overline{\mathrm{LP(X)}}$, then H is an ESF containing connectives → and ′ at most, where* $\overline{\mathrm{LP(X)}} = (\bar{\mathcal{F}}, \vee, \wedge, ', \rightarrow)$ *is a LIA*, $\bar{\mathcal{F}} = \mathcal{F}/_{=} = \{ \overline{p} \mid p \in \mathcal{F} \}$, $\overline{p} = \{q \mid$ *for any valuation v in LP(X)*, $v(q) = v(p)\}$, *for any* $\overline{p}, \overline{q} \in \bar{\mathcal{F}}$, $\overline{p} \vee \overline{q} = \overline{p \vee q}$, $\overline{p} \wedge \overline{q} = \overline{p \wedge q}$, $(\overline{p})' = \overline{p'}$, $\overline{p} \rightarrow \overline{q} = \overline{p \rightarrow q}$.

Definition 8. [15] *All the constants, literals and IESFs in LP(X) are called generalized literals. Here, the definition of literal is the same as that in classical logic.*

The disjunction of a finite number of generalized literals is a generalized clause.

Definition 9. [23] *Let $C_i = p_{i1} \vee \ldots \vee p_{im_i}$ be generalized clauses of LP(X), $H_i = \{ p_{i1}, \ldots, p_{im_i} \}$ the set of all disjuncts occurring in C_i, $i = 1, 2, \ldots, m$, $\alpha \in L$. For any $i \in \{1, 2, \ldots, m\}$, if there exist generalized literals $x_i \in H_i$ such that $x_1 \wedge x_2 \wedge \ldots \wedge x_m \leq \alpha$, then*

$$C_1(x_1 = \alpha) \vee C_2(x_2 = \alpha) \vee \ldots \vee C_m(x_m = \alpha)$$

is called an α-resolvent of $C_1, C_2, \ldots, C_m$, denoted by $R_{p(g\text{-}\alpha)}(C_1(x_1), C_2(x_2), \ldots, C_m(x_m))$, $x_1, x_2, \ldots, x_m$ are called an α-resolution group.

Definition 10. [22] *Let C be a generalized clause in lattice-valued propositional logic LP(X). C is called a locked generalized clause if each disjunct occurring in C is assigned a positive integer in its lower left corner (the same disjunct appearing in different locations can be labeled different positive integer). The positive integer is called a lock of the disjunct.*

2.3. *Lattice-valued first-order logic LF(X)*

Definition 11. [17] *Suppose V and F are the set of variable symbols and that of functional symbols in lattice-valued first-order logic LF(X), respectively, the set of terms of LF(X) is defined as the smallest set $\mathcal{T}$ satisfying the following conditions*:

(1) $V \subseteq \mathcal{T}$,

(2) for any $n \in \mathrm{N}$, if $f^{(n)} \in F$, then for any $t_0, t_1, \ldots, t_n \in \mathcal{T}$, $f^{(n)}(t_0, t_1, \ldots, t_n) \in \mathcal{T}$.

Remark 1. $f^{(0)}$ *is specified as a constant symbol.*

Definition 12. [17] *Suppose P is the predicate symbol set in lattice-valued first-order logic LF(X). The set of atoms of LF(X) is defined as the smallest set $\mathcal{A}_t$ satisfying the following condition*:

For any $n \in N$, if $P^{(n)} \in P$, then $P^{(n)}(t_0, t_1, \ldots, t_n) \in \mathcal{A}_t$ for any $t_0, t_1, \ldots, t_n \in \mathcal{T}$.

Remark 2. $P^{(0)}$ *is specified as a certain element in L.*

Definition 13. [17] *The set of formulas of lattice-valued first-order logic LF(X) is defined as the smallest set* $\mathscr{F}$ *satisfying the following conditions*:

(1) $\mathscr{A}_t \subset \mathscr{F}$,

(2) *if* $p, q \in \mathscr{F}$, *then* $p \to q \in \mathscr{F}$,

(3) *if* $p \in \mathscr{F}$, x *is a free variable in* p, *then* $(\forall x)\, p$, $(\exists x)\, p \in \mathscr{F}$.

Remark 3. *Note that* $p' = p \to O$, $p \vee q = (p \to q) \to q$, $p \wedge q = (p' \vee q')'$, $p \leftrightarrow q = (p \to q) \wedge (q \to p)$. *Therefore, if* $p, q \in \mathscr{F}$, *then* $p', p \vee q, p \wedge q, p \leftrightarrow q \in \mathscr{F}$.

Definition 14. [17] *Suppose* $G \in \mathscr{F}$, F_G *is the set of all functional symbols occurring in G,* P_G *is the set of all predicate symbols occurring in G, and* $D\ (\neq \phi)$ *is the domain of interpretation. An interpretation of G over D is a triple* $I_D = \langle D, \mu_D, v_D \rangle$, *where,*

$$\mu_D : F_G \to U_D = \{ f_D^{(n)} : D^n \to D \mid n \in N \}$$

$$f^{(0)} \mapsto f_D^{(0)},\ f_D^{(0)}(D^0) = \{ f_D^{(0)} \} \subseteq D,\ D^{(0)} \text{ is a non-empty set,}$$

$$f^{(n)} \mapsto f_D^{(n)}\ (n \in N^+),$$

$$v_D : P_G \to V_D = \{ P_D^{(n)} : D^n \to L \mid n \in N \}$$

$$p^{(0)} \mapsto p_D^{(0)},\ p_D^{(0)}(D^0) = \{ p_D^{(0)} \} \subseteq L$$

$$p^{(n)} \mapsto p_D^{(n)}\ (n \in N^+).$$

In lattice-valued first-order logic LF(X), the definitions of generalized literal and generalized clause are similar to those in lattice-valued propositional logic LP(X).

Definition 15. [17] *Let* $G \in \mathscr{F}$, $\alpha \in L$. *If* $v_D(G) \le \alpha$ *for any interpretation* $I_D = \langle D, \mu_D, v_D \rangle$ *in lattice-valued first-order logic LF(X), G is said to be* α*-false, denoted by* $G \le \alpha$.

In the following, the definitions of substitution, renamed substitution, ground substitution, instance, ground instance are the same as those in classical logic.

Definition 16. [23] *Let* $C_i = p_{i1} \vee \ldots \vee p_{im_i}$ *be generalized clauses without common variables in LF(X),* $H_i = \{p_{i1}, \ldots, p_{im_i}\}$ *the set of all disjuncts occurring in* C_i, $i = 1, 2, \ldots, m$, $\alpha \in L$. *If there exist generalized literals* $x_i \in H_i$ *and a substitution* σ *such that* $x_1^\sigma \wedge x_2^\sigma \wedge \ldots \wedge x_m^\sigma \le \alpha$, *then*

$$C_1^\sigma(x_1^\sigma = \alpha) \vee C_2^\sigma(x_2^\sigma = \alpha) \vee \ldots \vee C_m^\sigma(x_m^\sigma = \alpha)$$

is called an α*-resolvent of* $C_1, C_2, \ldots, C_m$, *denoted by* $R_{f(g\text{-}\alpha)}(C_1(x_1), C_2(x_2), \ldots, C_m(x_m))$. $x_1, x_2, \ldots, x_m$ *are called an* α*-resolution group.*

α occurring in the following is always less than *I*.

3. α-Quasi-Lock Semantic Resolution for Lattice-Valued Propositional Logic LP(X)

Definition 17. *Let* v_0 *be a valuation in lattice-valued propositional logic LP(X),* $\alpha \in L$. $N, E_1, \ldots, E_q$ *are sets composed of some locked generalized clauses in LP(X). The sequence* $(N, E_1, \ldots, E_q)$ *is called an* α*-quasi-lock semantic clash (*α*-QLS clash for short) w.r.t.* v_0, *if* $N, E_1, \ldots, E_q$ *satisfy the following conditions*:

(1) *for any generalized clause* $C_{i_r} \in E_i$, $v_0(C_{i_r}) \le \alpha$, *where* $i = 1, 2, \ldots, q$,

(2) *let* $R_0 = \bigvee_{C \in N} C$. *For any* $i = 1, 2, \ldots, q$, *there exists an* α*-resolvent* R_i *of* N_i *and* E_i, *where* $N_1 = N$, $N_2 = \{R_1\} \cup N_2^*$, $N_2^* \subseteq N$ *and for any* $i = 3, \ldots, q$, $N_i = \{R_{i-1}\} \cup N_i^*$, $N_i^* \subseteq N \cup \{R_1, \ldots, R_{i-2}\}$,

(3) *for any generalized clause* $C_{i_r} \in E_i$, *the* α*-resolution literal* g_{i_r} *of* C_{i_r} *is the one that has the smallest lock among disjuncts occurring in* C_{i_r}, $i = 1, 2, \ldots, q$,

(4) *for any generalized clause* $C_{j_t} \in N_j$, *the* α*-resolution literal* g_{j_t} *of* C_{j_t} *is the one which is non-*α*-false under valuation* v_0 *and has the smallest lock among non-*α*-false disjuncts (under* v_0*) occurring in* C_{j_t}, *where* $j = 1, 2, \ldots, q$,

(5) $v_0(R_q) \le \alpha$,

R_q *is called the* α*-QLS resolvent of this clash.* $E_1, \ldots, E_q$ *are called electrons and N is called the core of this clash.*

Remark 4. (1) *For any generalized clause C occurring in Definition 17, if there exists the same disjunct occurring in different locations of C, then retain the one with the smallest lock and delete others. For example, let* $C = {}_4g \vee {}_5g \vee {}_6g \vee {}_9h \vee {}_8h$ *be a locked generalized clause, if g and h are different disjuncts, then we rewrite C as* $C = {}_4g \vee {}_8h$.

(2) *For any disjunction* h_{i_r} *occurring in* E_i, $v_0(h_{i_r}) \le \alpha$, $i = 1, 2, \ldots, q$. *In fact, let generalized clause* $C_{i_r} \in E_i$

and $C_{i_r} = C^*_{i_r} \vee h_{i_r}$, *if* $v_0(h_{i_r}) \not\leq \alpha$, *then* $v_0(C_{i_r}) \not\leq \alpha$, *which is a contradict to* $v_0(C_{i_r}) \leq \alpha$.

(3) *For any generalized clause* $C \in N$, $v_0(C) \not\leq \alpha$. *In fact, if there exists a generalized clause* $C^* \in N$ *such that* $v_0(C^*) \leq \alpha$, *then there is no* α*-resolution literal occurring in* C^* *by condition (4) of Definition 17, which means there is no* α*-QLS clash.*

(4) *Since the* α*-QLS resolvent* R_q *satisfying* $v_0(R_q)\leq \alpha$, *to obtain the* α*-QLS resolvent as soon as possible, the previous* α*-QLS resolvent must be involved in the next resolution, i.e, for the i-th resolution,* $R_{i-1}\in N_i$. *As the generalized clauses* $C_1, C_2,..., C_k$ *(*$C_1, C_2,..., C_k$ *and* R_{i-1} *constitute resolution generalized clauses) not only occur in* E_i, *but also in* $N, R_1,..., R_{i-2}$, *hence* $N_i = \{R_{i-1}\} \cup N_i^*$, $N_i^* \subseteq N\cup\{R_1,..., R_{i-2}\}$.

(5) *If* i=2, *then let* $N_2^*=A_1\cup A_2$ *and* $A_1=\phi$. *Otherwise, let* $N_i^*=A_1\cup A_2$, $i = 3,..., q$. *The construction of* A_1 *and* A_2 *is as follows*:

Step 1: Let $A^*=\{R \mid R \in\{R_1,..., R_{i-2}\}\}$. *For any* $R^*\in A^*$, R^* *satisfies the following conditions*:

1> *the* α*-resolution literal of* R_{i-1} *does not occur in* R^*,

2> *the* α*-resolution literals of* R_{i-1} *and* R^* *do not come from the same generalized clause of* N.

Step 2: *For any generalized clause* $D \in A^*$, *if there exists* $D^*\in A^*$ *such that each disjunct of* D *is a disjunct of* D^*, *then delete* D^*.

Step 3: *After step 2, we can obtain a set, denoted by* A_1.

Step 4: *Let* $A_2 = \{C \mid C \in N\}$. *For any* $C^*\in A_2$, C^* *satisfies the following conditions*:

1> *the* α*-resolution literal of* R_{i-1} *does not come from* C^*,

2> *for any generalized clause* $E \in A_1$, *the* α*-resolution literal of* E *does not come from* C^*.

Example 3. *Let* $C_1 = {}_1(x \to y)'$, $C_2 = {}_2(x \to y) \vee {}_3(y \to z)' \vee {}_4(s \to a_3)'$, $C_3 = {}_5(s \to a_4) \vee {}_6(r \to t)$ *be three locked generalized clauses in lattice-valued propositional logic* $\mathscr{L}_9$P(X) *based on* $\mathscr{L}_9$ *and* $S = C_1 \wedge C_2 \wedge C_3 \wedge C_4$, *where* $\mathscr{L}_9$ *is the Łukasiewicz implication algebra with nine elements,* x, y, z, r, s, t *are propositional variables,* $a_3, a_4 \in L_9$. *Suppose* $\alpha = a_5 \in L_9$, v_0 *is the valuation of* $\mathscr{L}_9$P(X) *such that* $v_0(x) = I$, $v_0(y) = a_8$, $v_0(z) = a_3$, $v_0(s) = I$, $v_0(r) = a_7$, $v_0(t) = a_2$, *then we can obtain an* α*-QLS clash* $(N, E_1,..., E_q)$ *by Definition 17.*

In fact, since $v_0(C_1) = a_2 < \alpha$, $v_0(C_2) = a_8 > \alpha$, $v_0(C_3) = a_4 < \alpha$, so we can obtain an α-QLS clash (w.r.t. v_0) (N, E_1, E_2, E_3): $N = \{C_2\}$, $E_1 = \{C_1\}$, $E_2 = \{C_1\}$, $E_3 = \{C_3\}$ and the α-QLS resolvent R_3 of this clash is ${}_6(r \to t) \vee \alpha$, where $R_1 = {}_3(y \to z)' \vee {}_4(s \to a_3)' \vee \alpha$, $N_2 = \{R_1\}$, $R_2 = {}_4(s \to a_3)' \vee \alpha$, $N_3 = \{R_2\}$.

If we let $E_1 = \{C_3\}$, $E_2 = E_3 =\{C_1\}$, then there is no α-QLS clash. Therefore, we can obtain the fact that α-QLS clash is affected by the order of electrons.

Example 4. *Let* $C_1 = {}_1(x \to y)$, $C_2 = {}_2(x \to z)' \vee {}_6(s \to t)$, $C_3 = {}_3y' \vee {}_4(y \to z)$, $C_4 = {}_5(s \to t)' \vee {}_7(r \to (a_2, b_1))$ *be four locked generalized clauses in lattice-valued propositional logic* $(\mathscr{L}_9\times\mathscr{L}_2)$P(X) *based on* $\mathscr{L}_9\times\mathscr{L}_2$ *and* $S = C_1 \wedge C_2 \wedge C_3 \wedge C_4$, *where* $\mathscr{L}_9\times\mathscr{L}_2$ *is the same LIA with eighteen elements as that in Example 2,* x, y, z, r, s, t *are propositional variables,* $(a_2, b_1) \in L_9\times L_2$. *Suppose* $\alpha = (a_6, b_2) \in L_9\times L_2$, v_0 is the valuation of $(\mathscr{L}_9\times\mathscr{L}_2)$P(X) *such that* $v_0(x) = I$, $v_0(y) = v_0(z) = (a_2, b_2)$, $v_0(s) = (a_8, b_2)$, $v_0(t) = (a_6, b_1)$, $v_0(r) = (a_7, b_1)$, *then we can obtain an* α*-QLS clash* $(N, E_1,..., E_q)$ *by Definition 17.*

In fact, since $v_0(C_1) = (a_2, b_2) < \alpha$, $v_0(C_2) = (a_8, b_1)$ // α (here // means incomparable), $v_0(C_3) = I > \alpha$, $v_0(C_4) = (a_4, b_2) < \alpha$, so we can obtain an α-QLS clash (w.r.t. v_0) (N, E_1, E_2, E_3): $N = \{C_2, C_3\}$, $E_1 = \{C_1\}$, $E_2 = \{C_1\}$, $E_3 = \{C_4\}$ and the α-QLS resolvent R_3 of this clash is ${}_7(r \to (a_2, b_1)) \vee \alpha$, where $R_1 = {}_4(y \to z) \vee {}_6(s \to t) \vee \alpha$, $N_2 = \{C_2, R_1\}$, $R_2 = {}_6(s \to t) \vee \alpha$, $N_3 = \{R_2\}$.

Definition 18. *Suppose* $S = C_1 \wedge C_2 \wedge\ldots\wedge C_m$, *where* $C_1, C_2,..., C_m$ *are locked generalized clauses in lattice-valued propositional logic LP(X),* v_0 *is a valuation in LP(X) and* $\alpha \in L$. $\{\Phi_1, \Phi_2,..., \Phi_t\}$ *is called an* α*-quasi-lock semantic resolution deduction (w.r.t.* v_0*) (*α*-QLS resolution deduction for short) from* S *to generalized clause* Φ_t, *if it satisfies the following conditions*:

(1) Φ_i *is a generalized clause occurring in* S *or*

(2) Φ_i *is an* α*-QLS resolvent, where the core and electrons of* Φ_i *are composed of* Φ_j *(*$j < i$*) or generalized clauses occurring in* S.

Theorem 1. (Soundness) *Suppose* $S = C_1 \wedge C_2 \wedge\ldots\wedge C_m$, *where* $C_1, C_2,..., C_m$ *are locked generalized clauses in lattice-valued propositional logic* LP(X), $\alpha \in L$. v_0 *is a valuation in LP(X) and* $\{\Phi_1, \Phi_2,..., \Phi_t\}$ *is an* α*-QLS resolution deduction (w.r.t.* v_0*) from* S *to generalized* *clause Φ_t. If Φ_t is an α-false generalized clause, then S ≤ α, i.e., if $\Phi_t \le \alpha$, then $S \le \alpha$.*

Proof. According to the soundness of the general form of α-resolution principle in lattice-valued propositional logic LP(X) [23], we can obtain the result easily. □

Theorem 2. (Conditional Completeness) *Let $S = C_1 \wedge C_2 \wedge \ldots \wedge C_m$, where $C_1, C_2, \ldots, C_m$ are locked generalized clauses in lattice-valued propositional logic LP(X). v_0 is a valuation of LP(X) and $\alpha \in L$. If the following conditions hold:*

(1) $S \le \alpha$,

(2) $S^{\Diamond} \neq \phi$, *where* $S^{\Diamond} = \{C_i \mid v_0(C_i) \le \alpha, i \in \{1, 2, \ldots, m\}\}$,

(3) *there exists at least a locked generalized clause* $C_j \in \{C_1, C_2, \ldots, C_m\}$, *such that for any disjunct g of* C_j, $v_0(g) \not\le \alpha$,

then there exists an α-QLS resolution deduction (with respect to v_0) from S to an α-false generalized clause.

Proof. Suppose H_i is the set of all disjuncts occurring in C_i and $|H_i| = w_i$, $i = 1, 2, \ldots, m$. Let $K(S)$ be equal to the number of disjuncts occurring in S minus that of generalized clauses occurring in S, i.e., $K(S) = \sum_{i=1}^{m} w_i - m$. We have the following two cases.

Case 1: If $K(S) = 0$, then S is composed of unit generalized clauses, i.e., each generalized clause occurring in S includes only one generalized literal. Since $S \le \alpha$, so all generalized literals occurring in S compose an α-resolution group. As the condition (2) of Theorem 2 holds, so we have $S = S^{\Delta} \cup S^{\Delta\Delta}$ and $S^{\Delta}, S^{\Delta\Delta} \neq \phi$, where $S^{\Delta} = \{C_r \mid C_r$ is a generalized clause occurring in S, $v_0(C_r) \not\le \alpha\}$, $S^{\Delta\Delta} = \{C_t \mid C_t$ is a generalized clause occurring in S, $v_0(C_t) \le \alpha\}$. Let $N = S^{\Delta}$, $E = S^{\Delta\Delta}$, then (N, E) is an α-QLS clash and its α-QLS resolvent is an α-false generalized clause. Therefore the result holds.

Case 2: Suppose the result holds for $K(S) < n$, $n > 0$. Now we need to prove the result for $K(S) = n$.

1) Let $K(S) = n$, so S has at least one non-unit generalized clause. Suppose t is the largest lock of α-false disjuncts (under v_0) occurring in non-unit generalized clauses of S. Let $C_i = C_i^* \vee {}_t g$, where C_i^* is non-empty and $v_0({}_t g) \le \alpha$.

Suppose $S_1 = C_1 \wedge \ldots \wedge C_{i-1} \wedge C_i^* \wedge C_{i+1} \wedge \ldots \wedge C_m$, so $S_1 \le \alpha$ and $K(S_1) < n$. According to induction hypothesis, there exists an α-QLS resolution deduction D_1^* from S_1 to an α-false generalized clause.

a. If $v_0(C_i^*) \le \alpha$, then $v_0(C_i) \le \alpha$. In each α-QLS clash $(N^*, E_1^*, \ldots, E_q^*)$ of D_1^*, C_i^* can only be an element of electrons. If there exists $k \in \{1, 2, \ldots, q\}$ such that $C_i^* \in E_k^*$, then replace C_i^* with C_i. Since the lock t of disjunct ${}_t g$ is bigger than or equal to any other lock occurring in generalized clause C_i^*, so the α-QLS resolvent R_k of E_k and N_k^* is equal to $R_k^* \vee {}_t g$, where E_k is the set obtained by replacing C_i^* occurring in E_k^* with C_i, R_k^* is the α-QLS resolvent of E_k^* and N_k^*. Since $v_0({}_t g) \le \alpha$, so ${}_t g$ can not be the α-resolution generalized literal of R_j ($j = k, k+1, \ldots, q$). Hence, after changing C_i^* to C_i, the sequence $(N^*, E_1^*, \ldots, E_{k-1}^*, E_k, E_{k+1}^*, \ldots, E_q^*)$ is also an α-QLS clash and its α-QLS resolvent equals to $R_q^* \vee {}_t g$.

Since disjuncts of an α-false α-QLS resolvent (under v_0) are composed of the following two parts:

(i) disjuncts occur in the core and are α-false under v_0,

(ii) disjuncts occur in non-unit generalized clauses of electrons and are not α-resolution literals,

so, in each α-QLS clash $(N^*, E_1^*, \ldots, E_q^*)$ of D_1^*, if there exists $k \in \{1, 2, \ldots, q\}$ such that $R^* \in E_k^*$ is an original α-QLS resolvent, which is generated by the clash with C_i^* as an element of electrons, then after changing R^* to $R^* \vee {}_t g$, the sequence $(N^*, E_1^*, \ldots, E_{k-1}^*, E_k, E_{k+1}^*, \ldots, E_q^*)$ is also an α-QLS clash and its α-QLS resolvent is equal to $R_q^* \vee {}_t g$, where E_k is the set obtained by replacing R^* occurring in E_k^* with $R^* \vee {}_t g$, and R_q^* is the α-QLS resolvent of clash $(N^*, E_1^*, \ldots, E_q^*)$.

b. If $v_0(C_i^*) \not\le \alpha$, then C_i^* can only be an element occurring in the core of each α-QLS clash $(N^*, E_1^*, \ldots, E_q^*)$ of D_1^*. Since ${}_t g$ is not the α-resolution literal of each generalized clause occurring in the core N^*, so $(N, E_1^*, \ldots, E_q^*)$ is also an α-QLS clash and its α-QLS resolvent is equal to $R_q^* \vee {}_t g$, where N is the set obtained by replacing C_i^* occurring in N^* with $C_i^* \vee {}_t g$, and R_q^* is the α-QLS resolvent of clash $(N^*, E_1^*, \ldots, E_q^*)$.

Hence, after changing all C_i^* occurring in each α-QLS clash of D_1^* to C_i and modifying the corresponding α-QLS resolvent, we can obtain a resolution deduction D_1. From the above discussion, we can get that D_1 is an α-QLS resolution deduction from S to an α-false generalized clause or ${}_t g$.

If D_1 is an α-QLS resolution deduction from S to an α-false generalized clause, then the result holds.

If D_1 is an α-QLS resolution deduction from S to ${}_t g$, then let $S_2 = C_1 \wedge \ldots \wedge C_{i-1} \wedge {}_t g \wedge C_{i+1} \wedge \ldots \wedge C_m$. Obviously, $S_2 \leq \alpha$ and $K(S_2) < n$. According to induction hypothesis, there exists an α-QLS resolution deduction D_2 from S_2 to an α-false generalized clause. Connecting D_1 and D_2, we can obtain an α-QLS resolution deduction from S to an α-false generalized clause.

2) If t occurring in 1) does not exist, then disjuncts occurring in non-unit generalized clauses of S are not α-false under v_0. As $S^{\diamond} \neq \phi$, so for any $G \in S^{\diamond}$, G is a unit generalized clause. Since $S \leq \alpha$, for any $(g_1, g_2,\ldots, g_m) \in H_1 \times H_2 \times\ldots\times H_m$, we have $g_1 \wedge g_2 \wedge\ldots\wedge g_m \leq \alpha$. Let $N = \{C_1, C_2,\ldots, C_m\} \not\leq S^{\diamond}$. According to Definition 17, there exists an α-QLS clash (w.r.t. v_0) $(N, E_1,\ldots, E_q)$, and its α-QLS resolvent is an α-false generalized clause, where $E_s \subseteq S^{\diamond}$, $s = 1, 2,\ldots, q$. Hence, the result holds. □

Example 5. *Let* $C_1 = (x \to y)$, $C_2 = (x \to z)' \vee (s \to t)$, $C_3 = y' \vee (y \to z) \vee (s \to (a_4, b_1))$, $C_4 = (s \to t)' \vee (r \to (a_2, b_1))'$, $C_5 = r \to (a_5, b_1)$ *be five generalized clauses in lattice-valued propositional logic* $(\mathscr{L}_9\times\mathscr{L}_2)$P(X) *and* $S = C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5$, *where* $(a_4, b_1), (a_2, b_1), (a_5, b_1) \in L_9 \times L_2$ *and* x, y, z, r, s, t *are propositional variables. If* $\alpha = (a_6, b_1) \in L_9\times L_2$, *then* $S \leq \alpha$ *and there exists an* α-*QLS resolution deduction from* S *to an* α-*false generalized clause.*

In fact, we only need to find an α-QLS resolution deduction from S to an α-false generalized clause. Let C_1, C_2, C_3, C_4, C_5 have the following locks:

$C_1 = {}_1(x \to y)$,

$C_2 = {}_2(x \to z)' \vee {}_3(s \to t)$,

$C_3 = {}_4y' \vee {}_5(y \to z) \vee {}_6(s \to (a_4, b_1))$,

$C_4 = {}_7(s \to t)' \vee {}_8(r \to (a_2, b_1))'$,

$C_5 = {}_9(r \to (a_5, b_1))$.

Suppose v_0 is the valuation in $(\mathscr{L}_9\times\mathscr{L}_2)$P(X) such that $v_0(x) = I$, $v_0(y) = (a_5, b_1)$, $v_0(z) = (a_2, b_2)$, $v_0(s) = (a_6, b_2)$, $v_0(t) = (a_3, b_1)$, $v_0(r) = (a_6, b_1)$. Hence we have $v_0(C_1) < \alpha$, $v_0(C_2) > \alpha$, $v_0(C_3) > \alpha$, $v_0(C_4) \parallel \alpha$ ($\parallel$ means incomparable) $v_0(C_5) > \alpha$. Since the conditions (2) and (3) of Theorem 2 hold, so we have the following α-QLS resolution deduction:

(1) ${}_1(x \to y)$

(2) ${}_2(x \to z)' \vee {}_3(s \to t)$

(3) ${}_4y' \vee {}_5(y \to z) \vee {}_6(s \to (a_4, b_1))$

(4) ${}_7(s \to t)' \vee {}_8(r \to (a_2, b_1))'$

(5) ${}_9(r \to (a_5, b_1))$

(6) ${}_6(s \to (a_4, b_1))' \vee {}_3(s \to t) \vee \alpha$ by (1), (2), (3)

(7) ${}_8(r \to (a_2, b_1))' \vee {}_3(s \to t) \vee \alpha$ by (1), (4), (6)

(8) ${}_8(r \to (a_2, b_1))' \vee \alpha$ by (4), (7)

(9) α by (5), (8)

Hence, there exists an α-QLS resolution deduction form S to an α-false generalized clause, i.e., $S \leq \alpha$. In fact, there exists four α-QLS clashes $(N, E_1,\ldots, E_q)$ as follows:

(1) $N_1^1 = \{C_2, C_3\}$, $E_1^1 = \{C_1\}$, $E_2^1 = \{C_1\}$ and the α-QLS resolvent R_2^1 of (N_1^1, E_1^1, E_2^1) is ${}_6(s \to (a_4, b_1)) \vee {}_3(s \to t) \vee \alpha$, where $R_1^1 = {}_5(y \to z) \vee {}_6(s \to (a_4, b_1)) \vee {}_3(s \to t) \vee \alpha$, $N_2^1 = \{R_1^1, C_2\}$.

(2) $N_1^2 = \{R_2^1, C_4\}$, $E_1^2 = \{C_1\}$ and the α-QLS resolvent R_1^2 of (N_1^2, E_1^2) is ${}_8(r \to (a_2, b_1))' \vee {}_3(s \to t) \vee \alpha$.

(3) $N_1^3 = \{C_4\}$, $E_1^3 = \{R_1^2\}$ and the α-QLS resolvent R_1^3 of (N_1^3, E_1^3) is ${}_8(r \to (a_2, b_1))' \vee \alpha$.

(4) $N_1^4 = \{C_5\}$, $E_1^4 = \{R_1^3\}$ and the α-QLS resolvent R_1^4 of (N_1^4, E_1^4) is α.

4. α-Quasi-Lock Semantic Resolution for Lattice-Valued First-Order Logic LF(X)

Generalized clauses and generalized literals occurring in this section always belong to a generalized-Skölem standard form, i.e., for any generalized clause C and generalized literal g, all variables of C and g are bound variables with the quantifier $\forall$. For any generalized clauses $C_1, C_2,\ldots, C_m$ $(m \geq 3)$ in lattice-valued first-order logic LF(X), there exists at least a renamed substitution ε such that $C_1^{\varepsilon}, C_2^{\varepsilon},\ldots, C_m^{\varepsilon}$ have no common variables. Therefore, generalized clauses $C_1, C_2,\ldots, C_m$ $(m \geq 3)$ occurring in the following always have no common variables.

Definition 19. *Let* $I_D = \langle D, \mu_D, v_D\rangle$ *be an interpretation in lattice-valued first-order logic LF(X),* $\alpha \in L$ *and g a generalized literal in LF(X). g is called a non-*α*-false generalized literal w.r.t.* I_D, *if for any instance* g^0 of g, $v_D(g^0) \not\leq \alpha$. *g is called an* α*-pure-false generalized literal w.r.t.* I_D, *if for any instance* g^0 of g, $v_D(g^0) \leq \alpha$. *g is called an* α*-para-false generalized literal w.r.t.* I_D, *if* *there exist instances* g^{01}, g^{02} *of g such that* $v_D(g^{01}) \leq \alpha$ *and* $v_D(g^{02}) \not\leq \alpha$. *Both α-pure-false generalized literal and α-para-false generalized literal w.r.t.* I_D *are called α-false generalized literal w.r.t.* I_D.

Definition 20. *Let* $I_D = \langle D, \mu_D, v_D \rangle$ *be an interpretation in lattice-valued first-order logic LF(X),* $\alpha \in L$. *N,* $E_1,\ldots, E_q$ *are sets composed of some locked generalized clauses in LF(X). The sequence* $(N, E_1,\ldots, E_q)$ *is called an α-quasi-lock semantic clash (α-QLS clash for short) w.r.t.* I_D, *if N,* $E_1,\ldots, E_q$ *satisfy the following conditions*:

(1) *for any generalized clause* $C_{i_r} \in E_i$, $v_D(C_{i_r}) \leq \alpha$, $i = 1, 2,\ldots, q$,

(2) *let* $R_0 = \vee_{C\in N} C$. *For any* $i = 1, 2,\ldots, q$, *there exists an α-resolvent* R_i *of* N_i *and* E_i, *where* $N_1 = N$, $N_2 = \{R_1\} \cup N_2^*$, $N_2^* \subseteq N$ *and for any* $i = 3,\ldots, q$, $N_i = \{R_{i-1}\} \cup N_i^*$, $N_i^* \subseteq N \cup \{R_1,\ldots, R_{i-2}\}$,

(3) *for any generalized clause* $C_{i_r} \in E_i$, *the α-resolution literal* g_{i_r} *of* C_{i_r} *is the one that has the smallest lock among disjuncts occurring in* C_{i_r}, $i = 1, 2,\ldots, q$,

(4) *for any generalized clause* $C_{j_t} \in N_j$, *the α-resolution literal* g_{j_t} *of* C_{j_t} *is the one which not only has at least a non-α-false instance (w.r.t.* I_D*), and also has the smallest lock among disjuncts with non-α-false instances (w.r.t.* I_D*) occurring in* C_{j_t}, *where* $j = 1, 2,\ldots, q$,

(5) $v_D(R_q) \leq \alpha$,

R_q *is called the α-QLS resolvent of this clash. N is called the core and* $E_1,\ldots, E_q$ *are called electrons of this clash.*

Remark 5. (1) *For any generalized clause C occurring in Definition 20, if there exists the same disjunct occurring in different locations of C, then retain the one with the smallest lock and delete others.*

(2) *For any disjunction h occurring in* E_i, $v_D(h) \leq \alpha$, $i = 1, 2,\ldots, q$.

(3) *In general, there exists at least a generalized clause* $C^* \in N$ *such that* $v_D(C^*) \not\leq \alpha$. *In fact, if for any generalized clause* $C \in N$, $v_D(C) \leq \alpha$, *then* $v_D(R_0) \leq \alpha$, *i.e., there does not exist an α-QLS clash. If* R_0 *is seen as an α-QLS resolvent, then this α-QLS clash is redundant by the construction of* R_0.

Example 6. $C_1 = {}_1(M(x_1) \to N(x_2))'$, $C_2 = {}_3(M(b) \to N(y_1)) \vee {}_2P(y_2)$, $C_3 = {}_4(Q(z_1) \to P(c))' \vee {}_5(R(z_2) \to S(c))$ *be four locked generalized clauses in lattice-valued first-order logic* $\mathscr{L}_9F(X)$ *based on* $\mathscr{L}_9$, *and* $S = C_1 \wedge C_2 \wedge C_3 \wedge C_4$, *where* $x_1, x_2, y_1, y_2, z_1, z_2$ *are variables and b, c are constants. Suppose* $\alpha = a_5$ *and* $I_D = \langle D, \mu_D, v_D \rangle$ *is an interpretation of* $\mathscr{L}_9F(X)$, *where* $D = \{b, c\}$,

$$\frac{b}{b}, \frac{c}{c}, \frac{M(b)}{a_3}, \frac{M(c)}{a_3}, \frac{N(b)}{I}, \frac{N(c)}{I}, \frac{P(b)}{a_3},$$
$$\frac{P(c)}{a_6}, \frac{Q(b)}{a_4}, \frac{Q(c)}{a_4}, \frac{R(b)}{a_8}, \frac{R(c)}{a_8}, \frac{S(c)}{a_2}.$$

Then we can obtain an α-QLS clash $(N, E_1,\ldots, E_q)$ *by Definition 20.*

In fact, since $v_D(C_1) < \alpha$, $v_D(C_2) > \alpha$, $v(C_3) < \alpha$, so we can obtain an α-QLS clash (N, E_1, E_2): $N = \{C_2\}$, $E_1 = \{C_3\}$, $E_2 = \{C_1\}$ and the α-QLS resolvent R_2 of this clash is ${}_5(R(z_2) \to S(c)) \vee \alpha$, where $R_1 = {}_3(M(b) \to N(y_1)) \vee {}_5(R(z_2) \to S(c)) \vee \alpha$, $N_2 = \{R_1\}$.

Example 7. $C_1 = {}_1(M(x_1) \to N(a)) \vee {}_2(P(a) \to Q(x_2))$, $C_2 = {}_3(M(b) \to R(y_1))' \vee {}_4(S(c) \to T(y_2))$, $C_3 = {}_5(N(z_1) \to R(d))$, $C_4 = {}_6(S(u_1) \to T(d))'$ *be four locked generalized clauses in lattice-valued first-order logic* $(\mathscr{L}_9 \times \mathscr{L}_2)$F(X) *based on LIA* $\mathscr{L}_9 \times \mathscr{L}_2$, *and* $S = C_1 \wedge C_2 \wedge C_3 \wedge C_4$, *where* $x_1, x_2, y_1, y_2, z_1, u_1$ *are variables and a, b, c, d are constants. Suppose* $\alpha = (a_6, b_2) \in L_9 \times L_2$ *and* $I_D = \langle D, \mu_D, v_D \rangle$ *is an interpretation of* $(\mathscr{L}_9 \times \mathscr{L}_2)$F(X), *where* $D = \{a, b, c, d\}$,

$$\frac{a}{a}, \frac{b}{b}, \frac{c}{c}, \frac{d}{d}, \frac{M(a)}{I}, \frac{M(b)}{I}, \frac{M(c)}{I}, \frac{M(d)}{I},$$
$$\frac{N(a)}{(a_2,b_1)}, \frac{N(b)}{(a_2,b_1)}, \frac{N(c)}{(a_2,b_1)}, \frac{N(d)}{(a_2,b_1)}, \frac{P(a)}{I},$$
$$\frac{Q(a)}{(a_2,b_2)}, \frac{Q(b)}{(a_2,b_2)}, \frac{Q(c)}{(a_2,b_2)}, \frac{Q(d)}{(a_2,b_2)},$$
$$\frac{R(a)}{(a_5,b_2)}, \frac{R(b)}{(a_3,b_2)}, \frac{R(c)}{(a_3,b_2)}, \frac{R(d)}{(a_3,b_2)},$$
$$\frac{S(a)}{(a_4,b_2)}, \frac{S(b)}{(a_4,b_2)}, \frac{S(c)}{(a_4,b_2)}, \frac{S(d)}{(a_4,b_2)},$$
$$\frac{T(a)}{(a_3,b_1)}, \frac{T(b)}{(a_3,b_1)}, \frac{T(c)}{(a_3,b_1)}, \frac{T(d)}{(a_3,b_1)}.$$

Then we can obtain an α-QLS clash $(N, E_1,\ldots, E_q)$ *by Definition 20.*

In fact, since $v_D(C_1) < \alpha$, $v_D(C_2) \,//\, \alpha$, $v(C_3) > \alpha$, $v(C_4) < \alpha$, so we can obtain an α-QLS clash (w.r.t. I_D) (N, E_1, E_2): $N = \{C_2, C_3\}$, $E_1 = \{C_1\}$, $E_2 = \{C_4\}$ and the α-QLS resolvent R_2 of this clash is ${}_2(P(a) \to Q(x_2)) \vee \alpha$, where $R_1 = {}_4(S(c) \to T(y_2)) \vee {}_2(P(a) \to Q(x_2)) \vee \alpha$, $N_2 = \{R_1\}$.

Definition 21. *Suppose $S = C_1 \wedge C_2 \wedge \ldots \wedge C_m$, where $C_1, C_2, \ldots, C_m$ are locked generalized clauses in lattice-valued first-order logic LF(X), $\alpha \in L$ and $I_D = \langle D, \mu_D, v_D\rangle$ is an interpretation in LF(X). $\{\Phi_1, \Phi_2, \ldots, \Phi_t\}$ is called an α-quasi-lock semantic resolution deduction (w.r.t. I_D) (α-QLS resolution deduction for short) from S to generalized clause Φ_t, if it satisfies the following conditions:*

(1) Φ_i is a generalized clause occurring in S or

(2) Φ_i is an α-QLS resolvent (w.r.t. I_D), where the core and electrons of Φ_i are composed of Φ_j ($j < i$) or generalized clauses occurring in S.

Theorem 3. (Soundness) *Suppose $S = C_1 \wedge C_2 \wedge \ldots \wedge C_m$, where $C_1, C_2, \ldots, C_m$ are locked generalized clauses in lattice-valued first-order logic LF(X). $\alpha \in L$, $\{\Phi_1, \Phi_2, \ldots, \Phi_t\}$ is an α-QLS resolution deduction from S to generalized clause Φ_t. If Φ_t is an α-false generalized clause, then $S \le \alpha$, i.e., if $\Phi_t \le \alpha$, then $S \le \alpha$.*

Proof. According to the soundness of the general form of α-resolution principle in LF(X)[23], we can obtain the result easily. □

Theorem 4. (Lifting Lemma) *Let $N, E_1, \ldots, E_q$ be sets composed of locked generalized clauses in lattice-valued first-order logic LF(X), $I_{D*} = \langle D^*, \mu_{D*}, v_{D*}\rangle$ an interpretation in LF(X) and $\alpha \in L$. N^* is the set obtained by replacing each locked generalized clause C occurring in N with an instance C^* of C, and E_i^* is the set obtained by replacing each locked generalized clause C_i occurring in E_i with an instance C_i^* of C_i, $i = 1, 2, \ldots, q$. If $(N^*, E_1^*, \ldots, E_q^*)$ is an α-QLS clash (w.r.t. I_{D*}) and write its α-QLS resolvent as R_q^*, then there exists an interpretation $I_D = \langle D, \mu_D, v_D\rangle$ of LF(X) such that $(N, E_1, \ldots, E_q)$ is an α-QLS clash (w.r.t. I_D), and R_q^* is an instance of R_q, where R_q is the α-QLS resolvent of $(N, E_1, \ldots, E_q)$, i.e., Fig. 1 holds.*

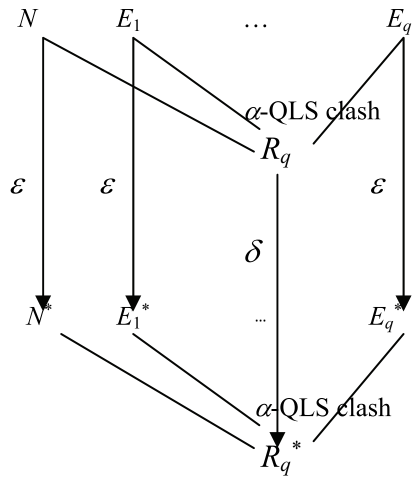

Fig. 1 Transformation Diagram

Remark 6. *In fact, $I_D = \langle D, \mu_D, v_D\rangle$ and $I_{D*} = \langle D^*, \mu_{D*}, v_{D*}\rangle$ satisfy the following conditions*:

(1) $D = D^*$.

(2) *For any function symbol $f^{(0)}$ occurring in N, $E_1, \ldots, E_q$, $f_D^{(0)} = f_{D*}^{(0)}$.*

(3) *Let $(d_1, \ldots, d_n) \in D^n (n \in \mathrm{N}^+)$. For any n-ary function symbol $f^{(n)}$ occurring in N, $E_1, \ldots, E_q$:*

1> if $f^{(n)}(d_1, \ldots, d_n)$ occurs in I_{D*}, then $\mu_D(f^{(n)}(d_1, \ldots, d_n)) = \mu_{D*}(f^{(n)}(d_1, \ldots, d_n))$,

2> if $f^{(n)}(d_1, \ldots, d_n)$ does not occur in I_{D*}, then $\mu_D(f^{(n)}(d_1, \ldots, d_n)) = \mu_{D*}(f^{(n)}(d_1^*, \ldots, d_n^*))$, where $(d_1^*, \ldots, d_n^*) \in D^n$ and $f^{(n)}(d_1^*, \ldots, d_n^*)$ occurs in I_{D*}.

(4) *Let $(d_1, \ldots, d_n) \in D^n (n \in \mathrm{N}^+)$. For any n-ary predicate symbol $p^{(n)}$ occurring in N, $E_1, \ldots, E_q$:*

1> if $p^{(n)}(d_1, \ldots, d_n)$ occurs in I_{D*}, then $v_D(p^{(n)}(d_1, \ldots, d_n)) = v_{D*}(p^{(n)}(d_1, \ldots, d_n))$,

2> if $p^{(n)}(d_1, \ldots, d_n)$ does not occur in I_{D*}, then $v_D(p^{(n)}(d_1, \ldots, d_n)) = v_{D*}(p^{(n)}(d_1^*, \ldots, d_n^*))$, where $(d_1^*, \ldots, d_n^*) \in D^n$, $p^{(n)}(d_1^*, \ldots, d_n^*)$ occurs in I_{D*}.

Proof. Since for any generalized clauses C_j^*, G_{ih}^* occurring in N^* and E_i^* respectively, there exist ground substitutions ε_j and ε_{ih} such that $C_j^* = C_j^{\varepsilon_j}$, $G_{ih}^* = G_{ih}^{\varepsilon_{ih}}$ and generalized clauses occurring in $N \cup E_1 \cup \ldots \cup E_q$ have no common variables with each other, we can obtain $C_j^* = C_j^{\varepsilon}$, $G_{ih}^* = G_{ih}^{\varepsilon}$, where $\varepsilon = \cup\{\varepsilon_{ih} \cup \varepsilon_j \mid 1 \le i \le q, h \in \Lambda_i, j$

$\in \Gamma\}$, C_j and G_{ih} are generalized clauses occurring in N and E_i respectively, Λ_i and Γ are index sets, $i = 1, 2,\ldots, q$. As $(N^*, E_1^*,\ldots, E_q^*)$ is an α-QLS clash (w.r.t. I_{D*}), hence $v_{D*}(C_j^*) \not\leq \alpha$ and $v_{D*}(G_{ih}^*) \leq \alpha$ for any generalized clauses C_j^*, G_{ih}^* occurring in N^* and E_i^* ($i =1, 2,\ldots, q$), respectively. Suppose $I_D = \langle D, \mu_D, v_D\rangle$ is an interpretation of LF(X), where I_D and I_{D*} satisfy Remark 6. For any disjunct g^* occurring in $N^* \cup E_1^* \cup\ldots\cup E_q^*$, the following results hold:

(1) if g^* is an α-para-false disjunct (w.r.t. I_{D*}), then g is an α-para-false disjunct (w.r.t. I_D),

(2) if g^* is an α-pure-false disjunct (w.r.t. I_{D*}), then g is an α-pure-false disjunct (w.r.t. I_D),

(3) if g^* is a non-α-false disjunct (w.r.t. I_{D*}), then g is a non-α-false disjunct (w.r.t. I_D),

where g^* is an instance of g, $g\in N \cup E_1 \cup\ldots\cup E_q$.

If N^*, $E_1^*,\ldots, E_q^*$ are the sets composed of the ground instances of all generalized clauses occurring in N, $E_1,\ldots, E_q$ respectively, then the above-mentioned result (1), (2) are: if g^* is an α-false disjunct (w.r.t. I_{D*}), then g is an α-false disjunct (w.r.t. I_D).

Let $N_i^* = \{C_{i1}^*, C_{i2}^*,\ldots, C_{ik}^*\}$, $E_i^* = \{G_{i1}^*, G_{i2}^*,\ldots, G_{ip}^*\}$, where $C_{ih}^* = C_{ih}^{\varepsilon}$, $G_{il}^* = G_{il}^{\varepsilon}$, $N_i = \{C_{i1}, C_{i2},\ldots, C_{ik}\}$, $E_i = \{G_{i1}, G_{i2},\ldots, G_{ip}\}$, $h = 1, 2,\ldots, k$, $l = 1, 2,\ldots, p$. Since $(N^*, E_1^*,\ldots, E_q^*)$ is an α-QLS clash (w.r.t. I_{D*}), so there exist substitution σ and disjuncts $x_1^*,\ldots, x_k^*$, $y_1^*,\ldots, y_p^*$such that $x_1^{*\sigma}\wedge\ldots\wedge x_k^{*\sigma}\wedge y_1^{*\sigma}\wedge\ldots\wedge y_p^{*\sigma} \leq \alpha$, where $x_h^{*\sigma}$ is the non-α-false disjunct (w.r.t. I_{D*}) with the smallest lock among non-α-false disjuncts (w.r.t. I_{D*}) occurring in generalized clause $C_{th}^{*\sigma}$, $y_l^{*\sigma}$ is the disjunct with the smallest lock occurring in generalized clause $G_{il}^{*\sigma}$, $x_h^* = x_h^{\varepsilon}$, $y_l^* = y_l^{\varepsilon}$, $h = 1, 2,\ldots, k$, $l = 1, 2,\ldots, p$. Hence the α-resolvent R_i^* of N_i^* and E_i^* is $C_{i1}^{*\sigma}(x_1^{*\sigma} = \alpha) \vee\ldots\vee C_{ik}^{*\sigma}(x_k^{*\sigma} = \alpha) \vee G_{i1}^{*\sigma}(y_1^{*\sigma} = \alpha) \vee\ldots\vee G_{ip}^{*\sigma}(y_p^{*\sigma} = \alpha)$, i.e., $R_i^* = C_{i1}^{\circ\varepsilon\cdot\sigma} \vee\ldots\vee C_{ik}^{\circ\varepsilon\cdot\sigma} \vee G_{i1}^{\circ\varepsilon\cdot\sigma} \vee\ldots\vee G_{ip}^{\circ\varepsilon\cdot\sigma} \vee \alpha$. According to lift lemma of the general form of α-resolution principle in LF(X) [23], there exists a most general unifier λ such that $\varepsilon\cdot\sigma = \lambda\cdot\delta$, where δ is a substitution. Therefore, $x_1^{\lambda} \wedge\ldots\wedge x_k^{\lambda} \wedge y_1^{\lambda} \wedge\ldots\wedge y_p^{\lambda} \leq \alpha$, and in C_{ih} (or G_{il}), if disjuncts $g_{i1},\ldots, g_{ir_i}$ are equal to x_i (or y_l) under substitution $\varepsilon\cdot\sigma$, then all the disjuncts, which are equal to x_i (or y_l) under substitution λ, are only $g_{i1},\ldots, g_{ir_i}$. Hence, the α-resolvent R_i of N_i and E_i is $C_{i1}^{\circ\lambda} \vee\ldots\vee C_{ik}^{\circ\lambda} \vee G_{i1}^{\circ\lambda} \vee\ldots\vee G_{ip}^{\circ\lambda} \vee \alpha$, where x_h^{λ} is the disjunct, which not only has at least a non-α-false instance (w.r.t. I_D), and also has the smallest lock among disjuncts with non-α-false instances (w.r.t. I_D) occurring in generalized clause C_{ih}^{λ}, y_l^{λ} is the disjunct with the smallest lock occurring in generalized clause G_{il}^{λ}, $h = 1, 2,\ldots, k$, $l = 1, 2,\ldots, p$. Furthermore, for any disjunct g occurring in C_{ih} (or G_{il}), if $v_{D*}(g^{\varepsilon\cdot\sigma}) \leq \alpha$, then $v_D(g^{\lambda}) \leq \alpha$. Therefore,

$$\begin{aligned} R_i^* &= C_{i1}^{\circ\varepsilon\cdot\sigma} \vee\ldots\vee C_{ip}^{\circ\varepsilon\cdot\sigma} \vee G_{i1}^{\circ\varepsilon\cdot\sigma} \vee\ldots\vee G_{ik}^{\circ\varepsilon\cdot\sigma} \vee \alpha \\ &= C_{i1}^{\circ\lambda\cdot\delta} \vee\ldots\vee C_{ip}^{\circ\lambda\cdot\delta} \vee G_{i1}^{\circ\lambda\cdot\delta} \vee\ldots\vee G_{ik}^{\circ\lambda\cdot\delta} \vee \alpha \\ &= (C_{i1}^{\circ\lambda} \vee\ldots\vee C_{ip}^{\circ\lambda} \vee G_{i1}^{\circ\lambda} \vee\ldots\vee G_{ik}^{\circ\lambda} \vee \alpha)^{\delta} \\ &= R_i^{\delta} \end{aligned}$$

Hence $(N, E_1,\ldots, E_q)$ is also an α-QLS clash (w.r.t. I_D) and R_q^* is a ground instance of R_q, where R_q is the α-QLS resolvent of $(N, E_1,\ldots, E_q)$. □

Theorem 5. (Conditional completeness) *Let $S = C_1 \wedge C_2 \wedge\ldots\wedge C_m$, where $C_1, C_2,\ldots, C_m$ are locked generalized clauses in lattice-valued first-order logic LF(X). $\alpha \in L$, $I_D = \langle D, \mu_D, v_D\rangle$ is an interpretation in LF(X) and for any disjunct g occurring in S, g is not an α-para-false disjuct (w.r.t. I_D). If the following conditions hold*:

(1) *$S \leq \alpha$,*

(2) *$S^{\Diamond} \neq \phi$, where $S^{\Diamond} = \{C_i \mid v_D(C_i) \leq \alpha, i \in\{1, 2,\ldots, m\}\}$,*

(3) *there exists at least a locked generalized clause C_j occurring in S such that $v_D(g) \not\leq \alpha$ for any disjunct g of C_j,*

then there exists an α-QLS resolution deduction (w.r.t. I_D) from S to an α-false generalized clause.

Proof. Since $S \leq \alpha$, according to Herbrand theorem [15], there exists a finite ground instance set S^0 such that $S^{0*} \leq \alpha$, where S^{0*} is the conjunction of all ground instances in S^0. As for any disjunct g of C_j, $v_D(g) \not\leq \alpha$, so for any disjunct g^0 of C_j^0, $v_{D*}(g^0) \not\leq \alpha$, where C_j^0 is a ground instance of C_j, $C_j^0 \in S^0$ and $I_{D*} = \langle D^*, \mu_{D*}, v_{D*}\rangle$ is an interpretation of LF(X), $D^* = D$, $v_{D*} = v_D$, μ_{D*} adds the interpretation of the constants only occurring in S^0 and not in S based on μ_D. Since $S^{\Diamond} \neq \phi$ and there is no α-para-false disjuct (w.r.t. I_D) occurring in S, so for any generalized clause $C_k \in S^{\Diamond}$, $v_{D*}(C_k^0) \leq \alpha$, where C_k^0 is a ground instance of C_k, $C_k^0 \in S^0$. According to Theorem 2, there exists an α-QLS resolution deduction (w.r.t. I_{D*})

ω^* from S^{0*} to an α-false generalized clause. Moreover, we can use Theorem 4 to lift ω^* to an α-QLS resolution deduction (w.r.t. I_D) from S to an α-false generalized clause. □

Example 8. *Let* $C_1 = (M(f(x_1)) \to N(x_2)) \vee (M(x_3) \to N(a))$, $C_2 = (M(f(b)) \to P(y_1))' \vee (Q(y_2) \to R(c))'$, $C_3 = (N(z_1))' \vee (N(a) \to P(z_2)) \vee (S(z_3) \to T(d))'$, $C_4 = (Q(d) \to R(u_1)) \vee (Q(u_2) \to R(c))$, $C_5 = (T(v_1) \to W(a))'$ *be five generalized clauses in lattice-valued first-order logic* $(\mathcal{L}_9 \times \mathcal{L}_2)$F(X) *and* $S = C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5$, *where* $x_1, x_2, x_3, y_1, y_2, z_1, z_2, z_3, u_1, u_2, v_1$ *are variables and a, b, c, d are constants. If* $\alpha = (a_6, b_2) \in L_9 \times L_2$, *then there exists an α-QLS resolution deduction from S to an α-false generalized clause.*

Equip S with the following locks:

$C_1 = {}_1(M(f(x_1)) \to N(x_2)) \vee {}_9(M(x_3) \to N(a))$

$C_2 = {}_2(M(f(b)) \to P(y_1))' \vee {}_3(Q(y_2) \to R(c))'$

$C_3 = {}_4(N(z_1))' \vee {}_5(N(a) \to P(z_2)) \vee {}_6(S(z_3) \to T(d))'$

$C_4 = {}_7(Q(d) \to R(u_1)) \vee {}_{10}(Q(u_2) \to R(c))$

$C_5 = {}_8(T(v_1) \to W(a))'$.

As there exists ground substitution $\theta = \{b/x_1, a/x_2, f(b)/x_3, c/y_1, d/y_2, a/z_1, c/z_2, c/z_3, c/u_1, d/u_2, d/v_1\}$ such that $C_1^\theta = M(f(b)) \to N(a)$, $C_2^\theta = (M(f(b)) \to P(c))' \vee (Q(d) \to R(c))'$, $C_3^\theta = (N(a))' \vee (N(a) \to P(c)) \vee (S(c) \to T(d))'$, $C_4^\theta = Q(d) \to R(c)$, $C_5^\theta = (T(d) \to W(a))'$, i.e., $S^\theta = C_1^\theta \wedge C_2^\theta \wedge C_3^\theta \wedge C_4^\theta \wedge C_5^\theta$, according to Theroem 4.3, we only need to prove that there exists an α-QLS resolution deduction from S^θ to an α-false generalized clause.

Suppose $I_{D*} = \langle D^*, \mu_{D*}, \nu_{D*}\rangle$ is an interpretation of $(\mathcal{L}_9 \times \mathcal{L}_2)$F(X), where $D^* = \{a, b, c, d\}$,

$$\frac{a}{a},\frac{b}{b},\frac{c}{c},\frac{d}{d},\frac{f(b)}{a},\frac{M(a)}{(a_2,b_2)},\frac{M(b)}{(a_2,b_2)},\frac{M(c)}{(a_2,b_2)},\frac{M(d)}{(a_2,b_2)},$$
$$\frac{N(a)}{(a_4,b_1)},\frac{P(c)}{(a_3,b_1)},\frac{Q(d)}{(a_2,b_2)},\frac{R(c)}{(a_4,b_1)},\frac{S(c)}{(a_4,b_1)},\frac{T(d)}{O},\frac{W(a)}{(a_3,b_2)}.$$

Hence, $\nu_{D*}(C_1^\theta) \,//\, \alpha$, $\nu_{D*}(C_2^\theta) < \alpha$, $\nu_{D*}(C_3^\theta) > \alpha$, $\nu_{D*}(C_4^\theta) \,//\, \alpha$, $\nu_{D*}(C_5^\theta) < \alpha$. Moreover, we can obtain the following α-QLS resolution deduction ω^*:

(1*) ${}_1(M(f(b)) \to N(a))$

(2*) ${}_2(M(f(b)) \to P(c))' \vee {}_3(Q(d) \to R(c))'$

(3*) ${}_4(N(a))' \vee {}_5(N(a) \to P(c)) \vee {}_6(S(c) \to T(d))'$

(4*) ${}_7(Q(d) \to R(c))$

(5*) ${}_8(T(d) \to W(a))'$

(6*) ${}_3(Q(d) \to R(c))' \vee {}_4(N(a))' \vee {}_6(S(c) \to T(d))' \vee \alpha$ by (1*), (2*), (3*)

(7*) ${}_4(N(a))' \vee {}_6(S(c) \to T(d))' \vee \alpha$ by (4*), (6*)

(8*) ${}_3(Q(d) \to R(c))' \vee {}_6(S(c) \to T(d))' \vee \alpha$ by (1*), (2*), (7*)

(9*) ${}_6(S(c) \to T(d))' \vee \alpha$ by (4*), (8*)

(10*) α by (1*), (5*), (9*)

Therefore, ω^* is an α-QLS resolution deduction (w.r.t. I_{D*}) from S^θ to an α-false generalized clause.

Let $I_D = \langle D, \mu_D, \nu_D\rangle$ be an interpretation of $(\mathcal{L}_9 \times \mathcal{L}_2)$F(X), where $D = \{a, b, c, d\}$,

$$\frac{a}{a},\frac{b}{b},\frac{c}{c},\frac{d}{d},\frac{f(a)}{a},\frac{f(b)}{a},\frac{f(c)}{a},\frac{f(d)}{a},$$
$$\frac{M(a)}{(a_2,b_2)},\frac{M(b)}{(a_2,b_2)},\frac{M(c)}{(a_2,b_2)},\frac{M(d)}{(a_2,b_2)},$$
$$\frac{N(a)}{(a_4,b_1)},\frac{N(b)}{(a_4,b_1)},\frac{N(c)}{(a_4,b_1)},\frac{N(d)}{(a_4,b_1)},$$
$$\frac{P(a)}{(a_3,b_1)},\frac{P(b)}{(a_3,b_1)},\frac{P(c)}{(a_3,b_1)},\frac{P(d)}{(a_3,b_1)},$$

$$\frac{Q(a)}{(a_2,b_2)},\frac{Q(b)}{(a_2,b_2)},\frac{Q(c)}{(a_2,b_2)},\frac{Q(d)}{(a_2,b_2)},$$
$$\frac{R(a)}{(a_4,b_1)},\frac{R(b)}{(a_4,b_1)},\frac{R(c)}{(a_4,b_1)},\frac{R(d)}{(a_4,b_1)},$$
$$\frac{S(a)}{(a_4,b_1)},\frac{S(b)}{(a_4,b_1)},\frac{S(c)}{(a_4,b_1)},\frac{S(d)}{(a_4,b_1)},$$
$$\frac{T(a)}{O},\frac{T(b)}{O},\frac{T(c)}{O},\frac{T(d)}{O},\frac{W(a)}{(a_3,b_2)}.$$

According to Theorem 4, after replacing C_i^θ ($i = 1,\ldots, 5$) occurring in ω^* with C_i, we can obtain an α-QLS resolution deduction (w.r.t. I_D) ω from S to an α-false generalized clause as follows:

(1) ${}_1(M(f(x_1)) \to N(x_2)) \vee {}_9(M(x_3) \to N(a))$

(2) ${}_2(M(f(b)) \to P(y_1))' \vee {}_3(Q(y_2) \to R(c))'$

(3) ${}_4(N(z_1))' \vee {}_5(N(a) \to P(z_2)) \vee {}_6(S(z_3) \to T(d))'$

(4) ${}_7(Q(d) \to R(u_1)) \vee {}_{10}(Q(u_2) \to R(c))$

(5) ${}_8(T(v_1) \to W(a))'$

(6) ${}_3(Q(y_2) \to R(c))' \vee {}_4(N(z_1))' \vee {}_6(S(z_3) \to T(d))' \vee \alpha$

by (1), (2), (3)

(7) $_4(N(z_1))' \vee {}_6(S(z_3) \to T(d))' \vee \alpha$ by (4), (6)

(8) $_3(Q(y_2) \to R(c))' \vee {}_6(S(z_3) \to T(d))' \vee \alpha$ by (1), (2), (7)

(9) $_6(S(z_3) \to T(d))' \vee \alpha$ by (4), (8)

(10) α by (1), (5), (9)

In fact, there are the following five α-QLS clashes $(N, E_1,\ldots, E_q)$ occurring in ω:

(1) $N_1^1 = \{C_1, C_3\}$, $E_1^1 = \{C_2\}$ and the α-QLS resolvent R_1^1 of (N_1^1, E_1^1) is $_3(Q(y_2) \to R(c))' \vee {}_4(N(z_1))' \vee {}_6(S(z_3) \to T(d))' \vee \alpha$.

(2) $N_1^2 = \{C_4\}$, $E_1^2 = \{R_1^1\}$ and the α-QLS resolvent R_1^2 of (N_1^2, E_1^2) is $_4(N(z_1))' \vee {}_6(S(z_3) \to T(d))' \vee \alpha$.

(3) $N_1^3 = \{C_1\}$, $E_1^3 = \{C_2, R_1^2\}$ and the α-QLS resolvent R_1^3 of (N_1^3, E_1^3) is $_3(Q(y_2) \to R(c))' \vee {}_6(S(z_3) \to T(d))' \vee \alpha$.

(4) $N_1^4 = \{C_4\}$, $E_1^4 = \{R_1^3\}$ and the α-QLS resolvent R_1^4 of (N_1^4, E_1^4) is $_6(S(z_3) \to T(d))' \vee \alpha$.

(5) $N_1^5 = \{C_1\}$, $E_1^5 = \{C_5, R_1^4\}$ and the α-QLS resolvent R_1^5 of (N_1^5, E_1^5) is α.

Remark 7. *The main difference between quasi-lock semantic (QLS for short) resolution in classical logic and α-QLS resolution in lattice-valued logic are the following two aspects*:

(1) *Electrons and core of QLS clash in classical logic are clauses, but electrons and core of α-QLS clash are sets composed of generalized clauses*.

(2) *Each resolution pair of QLS clash are composed of two literals, but each α-resolution group of α-QLS clash may include more than two generalized literals*.

Because of the above difference, for some false clause sets in classical logic, which do not have the completeness of QLS resolution, may be α-QLS resolved into empty clause. For example:

Example 9. Let $S = \{P(a), \sim P(x) \vee Q(y), \sim Q(b)\}$ be a clause set in classical logic, written as $S = P(a) \wedge (\sim P(x) \vee Q(y)) \wedge Q(b)$. Obviously, S is false and equip S with locks as follows:

(1) $_1P(a)$,

(2) $_3\sim P(x) \vee {}_2Q(y)$,

(3) $_4\sim Q(b)$.

Let the interpretation $I = \{\sim P(a), P(b), Q(a), \sim Q(b)\}$.

In fact, we can obtain that (1), (2) are false under I and (3) is true under I. Hence, only (3) is qualified to become the core. So we have the following two cases:

Case 1: According to QLS resolution in classical logic, we can obtain a QLS clash ((2), (3)) and the QLS resolvent of this QLS clash is $_3\sim P(x)$. Since $(_3\sim P(x), {}_4\sim Q(b))$ is not a resolution pair, so there is not other QLS clash. Hence, there does not exist a QLS resolution deduction form S to empty clause.

Case 2: According to α-QLS resolution, there exist two α-QLS clashes $(N, E_1,\ldots, E_q)$ as follows:

(1) $N_1^1 = \{(3)\}$, $E_1^1 = \{(2)\}$ and the α-QLS resolvent R_1^1 of (N_1^1, E_1^1) is $_3\sim P(x)$,

(2) $N_1^2 = \{(3)\}$, $E_1^2 = \{(1), R_1^1\}$ and the α-QLS resolvent R_1^2 of (N_1^2, E_1^2) is empty clause.

Therefore, there exists an α-QLS resolution deduction form S to empty clause.

In general, α-QLS resolution in lattice-valued first-order logic LF(X) can not be equivalently transformed into that for lattice-valued propositional logic LP(X), which means that the lifting lemma is usually not true. But we can obtain the conclusion under some special cases.

5. Conclusions

Combined with the benefits of lock resolution method and semantic resolution method in classical logic, α-quasi-lock semantic resolution method for a lattice-valued logic with truth-valued defined in a lattice-valued logical algebraic structure- lattice implication algebras (LIA) was discussed. Concretely, on the basis of the general form of α-resolution principle, α-quasi-lock semantic resolution method based on lattice-valued propositional logic LP(X) was established, and its soundness theorem and condition completeness theorem were proved. Secondly, the corresponding α-quasi-lock semantic resolution method in lattice-valued first-order logic LF(X) was proposed, its soundness and condition completeness were also established. This will become the theoretical foundation for automated reasoning in lattice-valued logic based on LIA. Meanwhile, this method can be used in areas such as automated theorem proving, program verification, and engineering technologies.

Acknowledgements

This work is partially supported by the National Natural Science Foundation of P. R. China (Grant No. 60875034, 61175055, 61100046); the project TIN-2009-0828; Sichuan Key Technology Research and Development Program of China (Grant No. 2011FZ0051) and Wireless Administration of Ministry of Industry and Information Technology of China ([2011]146).

References

1. J. P. Robinson, A machine-oriented logic based on the resolution principle, *J. ACM.* **12** (1965) 23-41.
2. L. Wos, J. A. Robinson and D. F. Carson, Efficiency and completeness of the set of support strategy in theorem proving, *J. ACM.* **12**(4) (1965) 536-541.
3. J. R. Slagle, Automatic theorem proving with renamable and semantic resolution, *J. ACM*, 14(4) (1967) 687-697.
4. D. W. Loveland, A linear format for resolution, in *Proc. IRIA Symp. Automatic Demonstration*, (Springer-Verlag, New York, 1970), pp. 147-162.
5. D. Luckham, Refinements in resolution theory, in *Proc. IRIA Symp. Automatic Demonstration*, (Springer-Verlag, New York, 1970), pp. 163-190.
6. R. Reiter, Two results on ordering for resolution with merging and linear format, *J. ACM.* **18**(4) (1971) 630-646.
7. B. C. Huang, A counterexample of OL resolution and modified MOL resolution, *J. Jilin Univ*. (*Science Edition*) **4** (1981) 91-96. (in Chinese)
8. X. H. Liu, Lock-semantic resolution principle with lemmas - LI-resolution principle, *J. Jilin Univ*. (*Science Edition*) **4** (1979) 129-136. (in Chinese)
9. X. H. Liu, Input semi-lock resolution method based on Horn sets, *Chinese Sci. Bulletin*, **16** (1985) 1201-1202. (in Chinese)
10. X. H. Liu and Y. F. Jiang, *Theorem Machine Proving*, Science Press, China, 1987. (in Chinese)
11. X. H. Liu, The compatibility among three resolution principles, *J. Software*, **2** (1992) 60-64. (in Chinese)
12. R. Z. Lu, K. Lin and Y. Q. Sun, Colored resolution, PI-Colored clash and its completeness, *Chinese J. Computers*, 1**2** (1987) 720-727. (in Chinese).
13. Z. L. Cai, Q. Huang, Q. Y. Huang and Y. D. Huang, Proving and optimizing the method of sum-up principle based on the strategy of supporting aggregate, *Modern Computer*, **5** (2005) 92-94. (in Chinese)
14. J. Meng, L.C. Paulson, Light weight relevance filtering for machine-generated resolution problems, *J. Applied Logic*, **7**(1) (2009) 41-57.
15. Y. Xu, D. Ruan, K.Y. Qin and J. Liu, *Lattice-Valued Logic: An Alternative Approach to Treat Fuzziness and Incomparability*, (Springer-Verlag, Heidelberg, 2003).
16. Y. Xu, D. Ruan, E.E. Kerre and J. Liu, α-Resolution principle based on lattice-valued propositional logic LP(X). *Inform. Sci.* **130** (2000) 195-223.
17. Y. Xu, D. Ruan, E.E. Kerre and J. Liu, α-Resolution principle based on lattice-valued first-order logic LF(X), *Inform. Sci.* **132** (2001) 221-239.
18. Y. Xu, X. B. Li, J. Liu and D. Ruan, Determination of α-resolution for lattice-valued first-order logic based on lattice implication algebra, in *Proc. the 2007 International Conference on Intelligent Systems and Knowledge Engineering*, (Chengdu, China, 2007), pp. 1567-1573.
19. Y. Xu, J. Liu, D. Ruan and X.B. Li, Determination of α-resolution in lattice-valued first-order logic LF(X), *Inform. Sci.* **181** (2011), 1836-1862.
20. Y. Xu, S. W. Chen, J. Liu and D. Ruan, Weak completeness of resolution in a linguistic truth-valued propositional logic, in *Proc. IFSA2007: Theoretical Advances and Applications of Fuzzy Logic and Soft Computing*(IFSA2007), (Cancun, Mexico, 2007), pp. 358-366.
21. X. B. Li, *The study of resolution automated reasoning for linguistic truth-valued lattice-valued logic*, Southwest Jiaotong University Doctor Degree Dissertation, 2008. (in Chinese).
22. X. X. He, Y. Xu and J. Liu, α-Satisfiability and α-Lock Resolution for a Lattice-Valued Logic LP(X), in *Proc. the 5th International Conference on Hybrid Artificial Intelligence Systems*, (San Sebastián, Spain, Jun.23-25, 2010), pp. 320-327.
23. Y. Xu, X.M. Zhong, J. Liu and S.W. Chen, General form of α-resolution principle based on lattice-valued logic with truth-value in lattice implication algebras, submitted to Information Sciences under review.
24. Y. Xu, W. T. Xu, X. M. Zhong, X. X. He, α-Generalized resolution principle based on lattice-valued propositional logic LP(X), in *Proc. the 9th International FLINS Conference* (FLINS2010), (Emei, Chengdu, China, 2010), pp. 66-71.
25. Y. Xu, S. W. Chen and J. Ma, Linguistic truth-valued lattice implication algebra and its properties, in *Proc. IMACS MuLticonference on Computational Engineering in Systems Applications* (CESA2006), (Beijing, China, 2006), pp. 1413-1418.
26. Y. Xu and K.Y. Qin, Lattice-valued propositional logic (I), *J. Southwest Jiaotong Univ*. **1**(2) (1993) 123-128.
27. K.Y. Qin, Y. Xu, Z.M. Song, Lattice-valued propositional logic LP(X) (I), *Fuzzy Sys. Math*. **11** (4) (1997) 5-10. (in Chinese).