Level-Ordered Q-Resolution and Tree-Like Q-Resolution are Incomparable

Meena Mahajan[a], Anil Shukla[a]

[a] *The Institute of Mathematical Sciences, Chennai 600113, India.*
{meena,anilsh}@imsc.res.in

Abstract

We show that Level-ordered Q-resolution and Tree-like Q-resolution, two restrictions of the Q-resolution system for proving false QBFs false, are incomparable. While the $\forall$Exp+Res system is known to p-simulate Tree-like Q-resolution, we observe that it cannot p-simulate Level-ordered Q-resolution.

Keywords: proof complexity, quantified Boolean formulas (QBF), resolution

1. Introduction

Resolution is a classical proof system for proving unsatisfiability of formulas in conjunctive normal form. Extending this to quantified Boolean formulas (QBFs), there are two major kinds of proof systems: those based on conflict-driven clause learning (CDCL), and those based on expansion. The most important CDCL-based proof system is Q-resolution, defined by Kleine Büning et al. [1]. An important expansion-based proof system is $\forall$Exp+Res defined by Janota and Marques-Silva [2]; this system corresponds to the expansion-based solver **RAReQS** [3]. The relative powers of both these systems are well studied, and the sytems are known to be incomparable. Looking at how incomparability was established, we see that two sub-classes of Q-resolution are significant: tree-like proofs, where the graph underlying the resolution structure is a tree, and level-ordered proofs, where at each resolution step, the variable on which resolution is performed is at the rightmost level (quantifier block) among all existential variables in the clauses involved. Level-ordered Q-resolution is practically important as well. It corresponds to QBF solvers based on the DPLL (Davis-Putnam-Logemann-Loveland) technique. One such example is the QBF Solver **Evaluate**, introduced in [4] (see also [5]); note, however, that **Evaluate** deals with variables in order of increasing level (as opposed to decreasing level in level-ordered Q-resolution).

The known results (see Figure 1) were established in the following chronological order.

1. $\forall$Exp+Res proof system cannot p-simulate Q-resolution.
   This was established by Janota and Marques-Silva in 2013 [6] (see also [7]). They defined a false QBF sentence that we denote ϕ_n, and showed that it is hard for $\forall$Exp+Res (Proposition 3, [7]) but has a polynomial size proof in Q-resolution (Proposition 2, [7]).
2. Level-ordered Q-resolution cannot p-simulate $\forall$Exp+Res.
   This too was shown by Janota and Marques-Silva in [7]. They defined a false QBF sentence CR_n and proved that CR_n is hard for level-ordered Q-resolution (Proposition 5, [7]) but has a polynomial size proof in $\forall$Exp+Res (Proposition 4, [7]).
3. Q-resolution cannot p-simulate $\forall$Exp+Res.
   This was shown by Beyersdorff et al. [8]. They showed that a formulation $\mathrm{QPARITY}_n$ of the parity function $\oplus_n$ is hard for Q-resolution (Section 4, [8]) but has a polynomial size proof in $\forall$Exp+Res. From this and (1) above, it follows that Q-resolution and $\forall$Exp+Res are incomparable.
4. $\forall$Exp+Res can p-simulate tree-like Q-resolution.
   This was shown by Janota and Marques-Silva in 2013 (Section 3, [2]). The converse direction is ruled out by the $\mathrm{QPARITY}_n$ formula. Since ϕ_n is hard for $\forall$Exp+Res, it follows that ϕ_n is hard for tree-like Q-resolution as well.

In this note, we show

Theorem 1. *Tree-like Q-resolution and level-ordered Q-resolution are incomparable.*

If we consider sentences with only existential quantifiers, then a Q-resolution proof is just a proof in general resolution. In fact, every resolution proof is level-ordered, since all variables are at the same level. Results from classical resolution thus imply that there are sentences (with only existential quantifiers) where level-ordered Q-resolution is exponentially more powerful than tree-like Q-resolution [9]. However this is not interesting since the power of

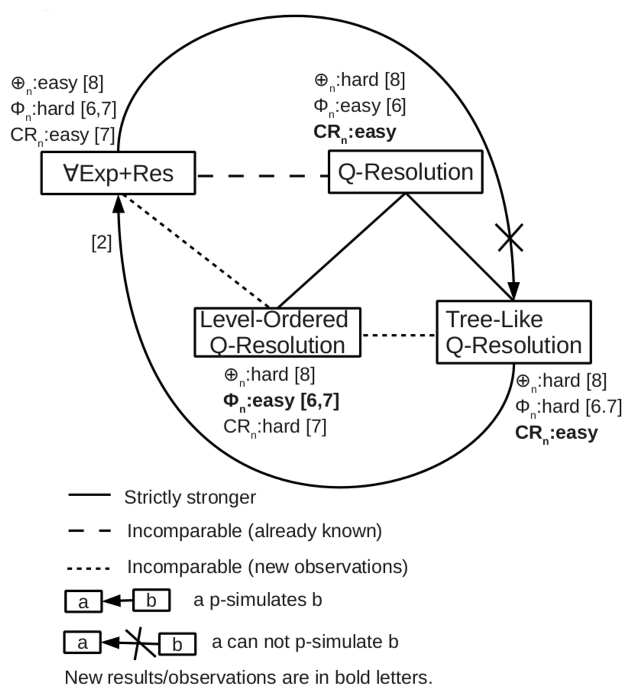

Figure 1: Relationships among some QBF resolution systems

the Q-resolution system to deal with universal quantifiers is not used. Another refinement of general resolution proofs is ordered resolution, where the variables are resolved in a specified order. This is known to be incomparable with tree-like resolution [10] (see also [9, 11]). In the context of QBFs, level-ordered is a weaker restriction (hence stronger system) than ordered, since no order is imposed on variables in the same quantifier block. Theorem 1 compares this stronger system with tree-like Q-resolution. Theorem 1 is also practically interesting because it underlines the fact that QBF solvers limit themselves greatly by assigning variables in the prefix order.

To prove Theorem 1, we proceed as follows: Firstly, we observe that the known polynomial size Q-resolution proof of ϕ_n ([6]; also in Section 6, [7]; item (1) above) is also in fact level-ordered. Therefore ϕ_n is in level-ordered Q-resolution. Since ϕ_n is hard for tree-like Q-resolution (item (4) above), we conclude that tree-like Q-resolution cannot p-simulate level-ordered Q-resolution. Furthermore, we conclude that even $\forall$Exp+Res cannot p-simulate level-ordered Q-resolution.

Next, we show that the sentences CR_n (item (2) above) have polynomial size tree-like Q-resolution proofs. As CR_n is hard for level-ordered Q-resolution, we conclude that level-ordered Q-resolution cannot p-simulate tree-like Q-resolution.

This completes the entire picture of relations among the above mentioned proof systems.

2. Definitions

For formal definitions of prenex form QBFs and of the various proof systems described above, the reader is referred to [7, 8]. Here we briefly describe the tree-like Q-resolution system, and the sentence CR_n.

Tree-like Q-resolution system: A proof in the Q-resolution system is a derivation of the empty clause (denoted $\Box$) from the initial clauses or axioms using the following rules: (1) resolve $A \vee x$ and $B \vee \bar{x}$ to get $A \vee B$, provided x is existentially quantified and $A \vee B$ is not a tautology (this is the 'Resolution' rule), and (2) replace $A \vee u$ by A provided u is universally quantified and all existential variables in A are quantified before u (this is the '$\forall$-Reduction' rule). If the underlying graph is a tree (that is, no derived clause is used more than once), then we have a tree-like Q-resolution proof.

Completion Principle and the sentence CR_n ([7]) :

Consider two sets $A = \{a_1, \ldots, a_n\}$ and $B = \{b_1, \ldots, b_n\}$, and depict their cross product $A \times B$ as in the table below.

a_1	a_1	$\ldots$	a_1	a_2	a_2	$\ldots$	a_2	$\ldots$	$\ldots$	a_n	a_n	$\ldots$	a_n
b_1	b_2	$\ldots$	b_n	b_1	b_2	$\ldots$	b_n	$\ldots$	$\ldots$	b_1	b_2	$\ldots$	b_n

The following two-player game is played on the above table. In the first round, player 1 deletes exactly one cell from each column. In the second round, player 2 chooses one of the two rows. Player 2 wins if the chosen row contains either the complete set A or the set B; otherwise player 1 wins. It is well known that player 2 has a winning strategy: suppose, after player 1 plays, some a_i is missing in the top row. Then the entire set B below the a_i chunk is present in the bottom row and so player 2 chooses the bottom row to win. Otherwise, no a_i is missing in the top row, so player 2 can win by choosing the top row. This fact (that player 2 can always win) is called the completion principle.

Based on the completion principle, the false sentence CR_n is formulated to express the notion that player 1 has a winning strategy. For each column $\begin{bmatrix} a_i \\ b_j \end{bmatrix}$ of the table (denote this the $(i,j)^{th}$ column), there is a boolean variable $x_{i,j}$. Let $x_{i,j} = 0$ denote that player 1 'deletes b_j' (i.e, keeps a_i) from the $(i,j)^{th}$ column, and $x_{i,j} = 1$ denotes that player 1 keeps b_j in the $(i,j)^{th}$ column. There is a variable z to denote the choice of player 2: $z = 0$ means 'choose top row'. The Boolean variables a_i, b_j, for $i, j \in [n]$ encode that for the chosen values of all the $x_{k,\ell}$, and the row chosen via z, at least one copy of the element a_i and b_j respectively is kept. (eg. $(x_{i,j} \wedge z) \Rightarrow b_j$). Let $\tilde{x}$, $\tilde{a}$ and $\tilde{b}$ stands for the vector of variables $\{x_{1,1}, x_{1,2}, \ldots, x_{n,n}\}$, $\{a_1, \ldots, a_n\}$, and $\{b_1, \ldots, b_n\}$ respectively. Now CR_n can be framed as follows:

$$\exists(\tilde{x}_{i,j})\ \forall z\ \exists\tilde{a}\exists\tilde{b}\ \left((\tilde{a},\tilde{b} \text{ consistent with } \tilde{x}, z)\ \wedge \bigvee_i \bar{a}_i \wedge \bigvee_j \bar{b}_j \right)$$

The inner formula can be expressed as the conjunction of the following clauses:

$$\text{For } i,j \in [n], C_{i,j} : \quad (x_{i,j} \vee z \vee a_i) \tag{1}$$

$$\text{For } i,j \in [n], D_{i,j} : \quad (\bar{x}_{i,j} \vee \bar{z} \vee b_j) \tag{2}$$

$$\bigvee_{i\in[n]} \bar{a}_i \tag{3}$$

$$\bigvee_{i\in[n]} \bar{b}_i \tag{4}$$

3. Tree-like Q-resolution proof for CR_n

Observe that to begin we cannot apply the $\forall$-Reduction rule because the only universal variable z has been blocked, in all clauses where it appears, by existential variables from $\tilde{a}$ and $\tilde{b}$. We also cannot resolve any $C_{i,j}$ and $D_{i,j}$ on variable $x_{i,j}$ because the resolvent is a tautology, which is not allowed in Q-resolution. We are thus forced to resolve on $\tilde{a}$ and $\tilde{b}$ variables initially.

We proceed as follows: We derive $\bar{z}$, and then apply a $\forall$-Reduction to derive $\Box$. To derive $\bar{z}$, we first derive each of the clauses $W_j = \bar{z} \vee b_j$ in a distinct tree T_j. Then we can put together these trees with the clause from

(4), and in n resolution steps, obtain $\bar{z}$, as follows: let C_1 denote the clause (4). For $\ell \in [n]$, resolve C_ℓ and W_ℓ (on variable b_ℓ) to get $C_{\ell+1}$. Note that for $\ell > 1$, C_ℓ has the form $\bar{z} \vee \bigvee_{k \geq \ell} \bar{b}_k$. So C_{n+1} is $\bar{z}$ as desired.

Now we describe the trees T_j that derive $W_j = \bar{z} \vee b_j$. We first derive the clause $x_{1,j} \vee x_{2,j} \vee \ldots \vee x_{n,j} \vee z$ in a tree T'_j described later. Now the $\forall$-Reduction rule is applicable, since all the $\tilde{x}$ variables are quantified before z. Thus we can obtain the clause $Y_{1,j} = x_{1,j} \vee x_{2,j} \vee \ldots \vee x_{n,j}$. Now for $\ell \in [n]$, resolve $Y_{\ell,j}$ with the clause $D_{\ell,j}$ from (2) (on variable $x_{\ell,j}$) to get $Y_{\ell+1,j}$. Note that for $\ell > 1$, $Y_{\ell,j}$ has the form $\bar{z} \vee b_j \vee \bigvee_{k \geq \ell} x_{k,j}$. So $Y_{n+1,j}$ is $\bar{z} \vee b_j$ as desired.

It remains to describe tree T'_j deriving $x_{1,j} \vee x_{2,j} \vee \ldots \vee x_{n,j} \vee z$. This is similar to the above step, using clause (3) which we shall denote $Z_{1,j}$ along with the clauses $C_{\ell,j}$ from (1). For $\ell \in [n]$, resolve $Z_{\ell,j}$ and $C_{\ell,j}$ on variable (a_ℓ) to get $Z_{\ell+1,j}$. For $\ell > 1$, $Z_{\ell,j}$ has the form $z \vee \bigvee_{k<\ell} x_{k,j} \vee \bigvee_{k \geq \ell} \bar{a}_k$. So $Z_{n+1,j}$ is $z \vee \bigvee_{k \in [n]} x_{k,j}$ as desired.

Size of the Refutation: Each T'_j has n resolution steps. Each T_j has T'_j, one $\forall$-reduction, and then n more resolution steps. Once all T_j's are constructed, we use another n resolutions steps followed by one last $\forall$-reduction. Overall, there are $n(2n+1)$ resolution steps and $n+1$ $\forall$-reductions. Thus the total refutation size is $O(n^2)$.

Acknowledgements

We thank the anonymous reviewers for their insightful comments.

References

[1] H. Kleine Büning, M. Karpinski, A. Flögel, Resolution for Quantified Boolean Formulas, Information and Computation 117 (1995) 12–18.

[2] M. Janota, J. Marques-Silva, On propositional QBF expansions and Q-resolution, in: Theory and Applications of Satisfiability Testing - SAT - 16th International Conference, Helsinki, Finland, July 8-12. Proceedings, Vol. 7962, 2013, pp. 67–82.

[3] M. Janota, W. Klieber, J. Marques-Silva, E. M. Clarke, Solving QBF with counterexample guided refinement, in: Theory and Applications of Satisfiability Testing - SAT - 15th International Conference, Trento, Italy, June 17-20. Proceedings, 2012, pp. 114–128.

[4] M. Cadoli, A. Giovanardi, M. Schaerf, An algorithm to evaluate Quantified Boolean Formulae, in: Proceedings of the Fifteenth National Conference on Artificial Intelligence and Tenth Innovative Applications of Artificial Intelligence Conference, AAAI, July 26-30, Madison, Wisconsin, USA., 1998, pp. 262–267.

[5] M. Cadoli, M. Schaerf, A. Giovanardi, M. Giovanardi, An algorithm to evaluate Quantified Boolean Formulae and its experimental evaluation, Journal of Automated Reasoning 28 (2) (2002) 101–142.

[6] M. Janota, J. Marques-Silva, $\forall\mathsf{Exp}{+}\mathsf{Res}$ does not p-simulate Q-resolution, in: International Workshop on Quantified Boolean Formulas, 2013, pp. 17–21, `http://fmv.jku.at/qbf2013/reportQBFWS13.pdf`.

[7] M. Janota, J. Marques-Silva, Expansion-based QBF solving versus Q-resolution, Theoretical Computer Science 577 (2015) 25–42.

[8] O. Beyersdorff, L. Chew, M. Janota, Proof complexity of resolution-based QBF calculi, in: 32nd International Symposium on Theoretical Aspects of Computer Science, STACS, 2015, pp. 76–89, full version in ECCC TR-2014/120.

[9] M. L. Bonet, J. L. Esteban, N. Galesi, J. Johannsen, On the relative complexity of resolution refinements and cutting planes proof systems, SIAM Journal on Computing 30 (5) (2000) 1462–1484.

[10] J. Johannsen, Exponential incomparability of tree-like and ordered resolution, manuscript (2001).

[11] J. Buresh-Oppenheim, T. Pitassi, The complexity of resolution refinements, Journal of Symbolic Logic 72 (4) (2007) 1336–1352.