ON Q-RESOLUTION AND CDCL QBF SOLVING

Mikoláš Janota

Microsoft Research, Cambridge, UK

Q-RESOLUTION [BÜNING ET AL., 1995]

Q-resolution = Q-resolution rule $+ \forall$*-reduction*

Q-RESOLUTION [BÜNING ET AL., 1995]

Q-resolution = Q-resolution rule $+\,\forall$*-reduction*

Resolution

$$\frac{l \vee C_1 \qquad \neg l \vee C_2}{C_1 \vee C_2}\ (l \text{ existentially quantified})$$

Q-RESOLUTION [BÜNING ET AL., 1995]

Q-resolution = Q-resolution rule + ∀-reduction

Resolution

$$\frac{l \lor C_1 \qquad \neg l \lor C_2}{C_1 \lor C_2} \ (l \text{ existentially quantified})$$

∀-reduction

$$\frac{C \lor k}{C} \ (k \in C \text{ is universal with highest quant. level in } C)$$

Q-RESOLUTION [BÜNING ET AL., 1995]

Q-resolution = Q-resolution rule $+ \forall$*-reduction*

Resolution

$$\frac{l \vee C_1 \qquad \neg l \vee C_2}{C_1 \vee C_2} \text{ (l existentially quantified)}$$

$\forall$-reduction

$$\frac{C \vee k}{C} \text{ ($k \in C$ is universal with highest quant. level in C)}$$

Tautologous resolvents are generally unsound and not allowed!

Q-RESOLUTION [BÜNING ET AL., 1995]

Q-resolution = Q-resolution rule + $\forall$*-reduction*

Resolution

$$\frac{l \vee C_1 \qquad \neg l \vee C_2}{C_1 \vee C_2} \text{ (l existentially quantified)}$$

$\forall$-reduction

$$\frac{C \vee k}{C} \text{ ($k \in C$ is universal with highest quant. level in C)}$$

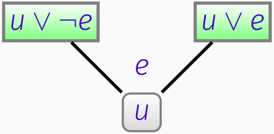
Tautologous resolvents are generally unsound and not allowed!

long-distance Q-resolution [Balabanov and Jiang, 2012] enables tautologous resolvents in *some cases*.

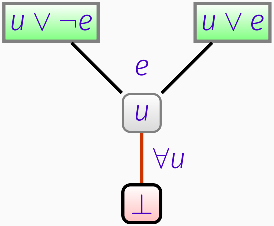
Q-RESOLUTION

$$\forall u \exists e.\ (u \vee \neg e) \wedge (u \vee e)$$

$$\forall u \exists e.\ (u \vee \neg e) \wedge (u \vee e)$$



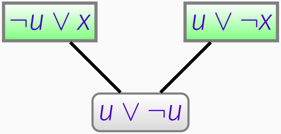
$$\forall u \exists e.\ (u \lor \neg e) \land (u \lor e)$$



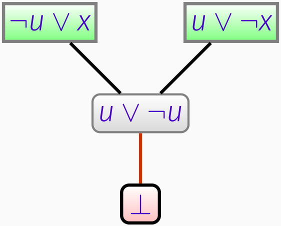
$$\forall u \exists x.\, (u \vee \neg x) \wedge (\neg u \vee x)$$

$\neg u \vee x$ $\quad$ $u \vee \neg x$

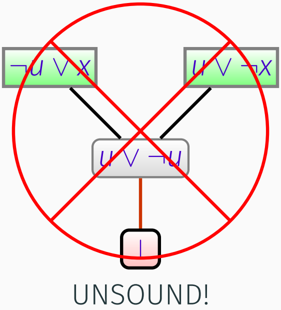
$$\forall u \exists x.\ (u \vee \neg x) \wedge (\neg u \vee x)$$



$$\forall u \exists x.\ (u \vee \neg x) \wedge (\neg u \vee x)$$



$$\forall u \exists x.\ (u \vee \neg x) \wedge (\neg u \vee x)$$



- Assign variables in the order of the prefix.

$$\exists x_1 \ldots x_n \forall u_1, \ldots, u_k \exists y_1 \ldots y_m \ldots . \phi$$

- Assign variables in the order of the prefix.

$$\exists x_1 \ldots x_n \forall u_1, \ldots, u_k \exists y_1 \ldots y_m \ldots . \phi$$

- Propagation + UIP Learning via Q-Resolution

- Easy to defeat [Lonsing, 2012]:

$$\exists x_1 \ldots x_n \forall z \exists y_1, \ldots, y_m .\, \phi_{\text{hard}}[X] \wedge \phi_{\text{easy}}[z, Y]$$

ASSIGNMENT ORDER IS A WEAKNESS

- Easy to defeat [Lonsing, 2012]:

$$\exists x_1 \ldots x_n \forall z \exists y_1, \ldots, y_m .\, \phi_{\mathrm{hard}}[X] \wedge \phi_{\mathrm{easy}}[z, Y]$$

- If $\phi_{\mathrm{hard}}[X]$ is hard to refute (e.g. Pigeon-hole) but $\phi_{\mathrm{easy}}[z, Y]$ is easy to refute, Q-Resolution defeats CDCL QBF Solving.

ASSIGNMENT ORDER IS A WEAKNESS

- Easy to defeat [Lonsing, 2012]:

$$\exists x_1 \ldots x_n \forall z \exists y_1, \ldots, y_m .\, \phi_{\text{hard}}[X] \wedge \phi_{\text{easy}}[z, Y]$$

- If $\phi_{\text{hard}}[X]$ is hard to refute (e.g. Pigeon-hole) but $\phi_{\text{easy}}[z, Y]$ is easy to refute, Q-Resolution defeats CDCL QBF Solving.
- Formulas are *independent*.

ASSIGNMENT ORDER IS A WEAKNESS

- Easy to defeat [Lonsing, 2012]:

$$\exists x_1 \ldots x_n \forall z \exists y_1, \ldots, y_m .\, \phi_{\text{hard}}[X] \wedge \phi_{\text{easy}}[z, Y]$$

- If $\phi_{\text{hard}}[X]$ is hard to refute (e.g. Pigeon-hole) but $\phi_{\text{easy}}[z, Y]$ is easy to refute, Q-Resolution defeats CDCL QBF Solving.
- Formulas are *independent*.
- The solver never assigns Y variables.

ASSIGNMENT ORDER IS A WEAKNESS

- Easy to defeat [Lonsing, 2012]:

$$\exists x_1 \ldots x_n \forall z \exists y_1, \ldots, y_m.\, \phi_{\text{hard}}[X] \wedge \phi_{\text{easy}}[z, Y]$$

- If $\phi_{\text{hard}}[X]$ is hard to refute (e.g. Pigeon-hole) but $\phi_{\text{easy}}[z, Y]$ is easy to refute, Q-Resolution defeats CDCL QBF Solving.
- Formulas are *independent*.
- The solver never assigns Y variables.
- Several ways how to deal with this issue.

- Consider a formula $\exists \mathcal{X}\, \forall z \exists \mathcal{L} \,.\, \phi[\mathcal{X}, z, \mathcal{L}]$ such that proofs are exponential if they start on $\mathcal{L}$ first.

- Consider a formula $\exists \mathcal{X}\, \forall z \exists \mathcal{L}\,.\, \phi[\mathcal{X}, z, \mathcal{L}]$ such that proofs are exponential if they start on $\mathcal{L}$ first.
- Prove that a DPLL-based solver resolves on $\mathcal{L}$ first.

THE FORMULA: COMPLETION PRINCIPLE

a_1	$\cdots$	a_1	$\cdots$	a_N	$\cdots$	a_N
b_1	$\cdots$	b_N	$\cdots$	b_1	$\cdots$	b_N

THE FORMULA: COMPLETION PRINCIPLE

a_1	$\cdots$	a_1	$\cdots$	a_N	$\cdots$	a_N
b_1	$\cdots$	b_N	$\cdots$	b_1	$\cdots$	b_N

- $\mathcal{X} = x_{11} \ldots x_{nn}$

a_1	$\cdots$	a_1	$\cdots$	a_N	$\cdots$	a_N
b_1	$\cdots$	b_N	$\cdots$	b_1	$\cdots$	b_N

- $\mathcal{X} = x_{11} \dots x_{nn}$
- $\mathcal{L} = a_1 \dots a_n, b_1 \dots b_n$

a_1	$\cdots$	a_1	$\cdots$	a_N	$\cdots$	a_N
b_1	$\cdots$	b_N	$\cdots$	b_1	$\cdots$	b_N

- $\mathcal{X} = x_{11} \dots x_{nn}$
- $\mathcal{L} = a_1 \dots a_n, b_1 \dots b_n$
- $\exists \mathcal{X} \forall z \exists \mathcal{L}$

$$x_{ij} \vee z \vee a_i,\ i, j \in 1..n$$

$$\neg x_{ij} \vee \neg z \vee b_j,\ i, j \in 1..n$$

$$\bigvee_{i \in 1..n} \neg a_i$$

$$\bigvee_{i \in 1..n} \neg b_i$$

- Completion Principle hard when resolving inside out [Janota and Marques-Silva, 2015].

COMPLETION PRINCIPLE: HARDNESS

- Completion Principle hard when resolving inside out [Janota and Marques-Silva, 2015].
- Even stronger: deriving a unit clause under such condition is already hard.

- Completion Principle hard when resolving inside out [Janota and Marques-Silva, 2015].
- Even stronger: deriving a unit clause under such condition is already hard.
- Before a unit clause is learned, there is no *propagation across levels*, i.e. $\mathcal{L}$ variables are given a value only after all $\mathcal{X}$ variables are given a value.

- Completion Principle hard when resolving inside out [Janota and Marques-Silva, 2015].
- Even stronger: deriving a unit clause under such condition is already hard.
- Before a unit clause is learned, there is no *propagation across levels*, i.e. $\mathcal{L}$ variables are given a value only after all $\mathcal{X}$ variables are given a value.
- Derivations of the first unit clause is always by resolving $\mathcal{L}$ first.

- In the proof we did not consider solution learning (only clause learning).

INTRICACIES

- In the proof we did not consider solution learning (only clause learning).
- When the universal plays correctly each time, there's no solution learning and then the same results apply.

- In the proof we did not consider solution learning (only clause learning).
- When the universal plays correctly each time, there's no solution learning and then the same results apply.

n	CDCL	CDCL + SDCL	CDCL + SDCL − pure lits.
4	101	101	101
5	1081	1081	751
6	19611	19611	3531
7	370811	370811	36411
8	$>$ 9995451	$>$ 10000981	5464551
9	$>$ 10612011	$>$ 10619361	$>$ 931211
10	$>$ 10303551	$>$ 10313901	$>$ 8608251

SUMMARY AND CONCLUSIONS

- Studied a formula where CDCL is over-performed by Q-resolution.

SUMMARY AND CONCLUSIONS

- Studied a formula where CDCL is over-performed by Q-resolution.
- Comes down to order of resolution: we force the solver to resolve on the innermost variables first.

SUMMARY AND CONCLUSIONS

- Studied a formula where CDCL is over-performed by Q-resolution.
- Comes down to order of resolution: we force the solver to resolve on the innermost variables first.
- Breaking the order: unit clauses, solution learning, propagation across levels.

SUMMARY AND CONCLUSIONS

- Studied a formula where CDCL is over-performed by Q-resolution.
- Comes down to order of resolution: we force the solver to resolve on the innermost variables first.
- Breaking the order: unit clauses, solution learning, propagation across levels.
- Learning unit clauses does not help because deriving a unit clauses already hard.

SUMMARY AND CONCLUSIONS

- Studied a formula where CDCL is over-performed by Q-resolution.
- Comes down to order of resolution: we force the solver to resolve on the innermost variables first.
- Breaking the order: unit clauses, solution learning, propagation across levels.
- Learning unit clauses does not help because deriving a unit clauses already hard.
- Solution learning may only help if universal player makes a wrong decision.

- Studied a formula where CDCL is over-performed by Q-resolution.
- Comes down to order of resolution: we force the solver to resolve on the innermost variables first.
- Breaking the order: unit clauses, solution learning, propagation across levels.
- Learning unit clauses does not help because deriving a unit clauses already hard.
- Solution learning may only help if universal player makes a wrong decision.
- Pure literals worsen performance.

SUMMARY AND CONCLUSIONS

- Studied a formula where CDCL is over-performed by Q-resolution.
- Comes down to order of resolution: we force the solver to resolve on the innermost variables first.
- Breaking the order: unit clauses, solution learning, propagation across levels.
- Learning unit clauses does not help because deriving a unit clauses already hard.
- Solution learning may only help if universal player makes a wrong decision.
- Pure literals worsen performance.
- Is the formula really hard for solution + conflict learning?

- Studied a formula where CDCL is over-performed by Q-resolution.
- Comes down to order of resolution: we force the solver to resolve on the innermost variables first.
- Breaking the order: unit clauses, solution learning, propagation across levels.
- Learning unit clauses does not help because deriving a unit clauses already hard.
- Solution learning may only help if universal player makes a wrong decision.
- Pure literals worsen performance.
- Is the formula really hard for solution + conflict learning?
- Other methods in solving to break the order?

Thank You for Your Attention!

Questions?

Balabanov, V. and Jiang, J.-H. R. (2012).
Unified QBF certification and its applications.
Formal Methods in System Design, 41(1):45–65.

Büning, H. K., Karpinski, M., and Flögel, A. (1995).
Resolution for quantified Boolean formulas.
Inf. Comput., 117(1).

Janota, M. and Marques-Silva, J. (2015).
Expansion-based QBF solving versus Q-resolution.
Theoretical Computer Science, 577(0):25–42.

Lonsing, F. (2012).
Dependency Schemes and Search-Based QBF Solving: Theory and Practice.
PhD thesis, Johannes Kepler Universität.