Contributions to the Theory of Practical QBF Solving

Allen Van Gelder
Computer Science Dept.
Univ. of California
Santa Cruz, CA, USA

http://www.cse.ucsc.edu/∼avg/

http://www.cse.ucsc.edu/∼avg/Papers/
These slides are qpup-trans.pdf

http://www.cse.ucsc.edu/∼avg/ProofChecker/
Software directory, contains QdpllexpSimple.tar.

Overview of Topics in QBF Solving

Exponential Case for Usual Clause Learning Procedure

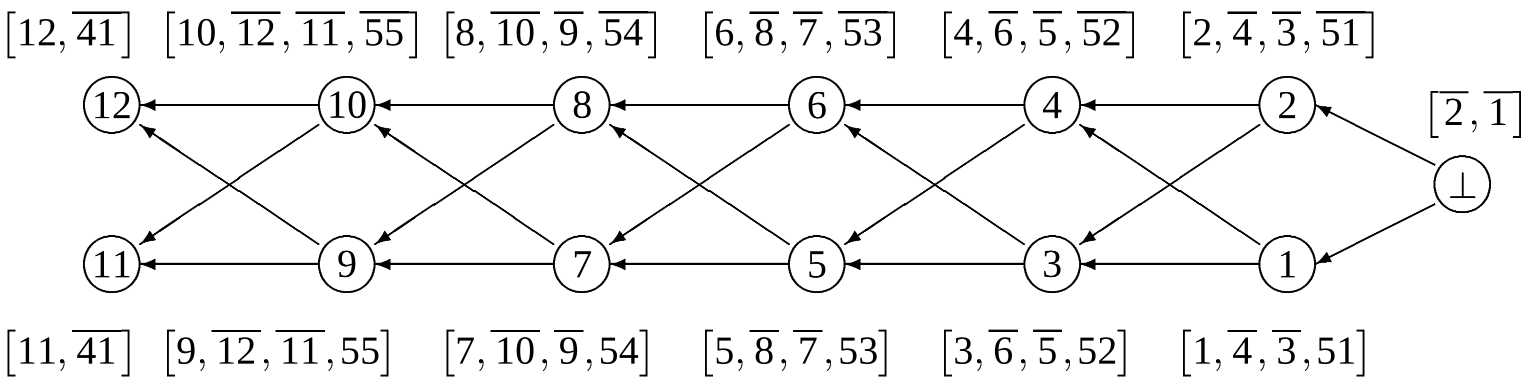
Shaky Proposal for QBF Pseudo-Unit Propagation (QPUP)

Observations on Pure Literals

- Treat Existential Pure Literals as Assumptions
- Treat Universal Pure Literals as Universal Reductions

Depth-Monotonic Literals

Exponential Case for Usual Clause Learning Procedure



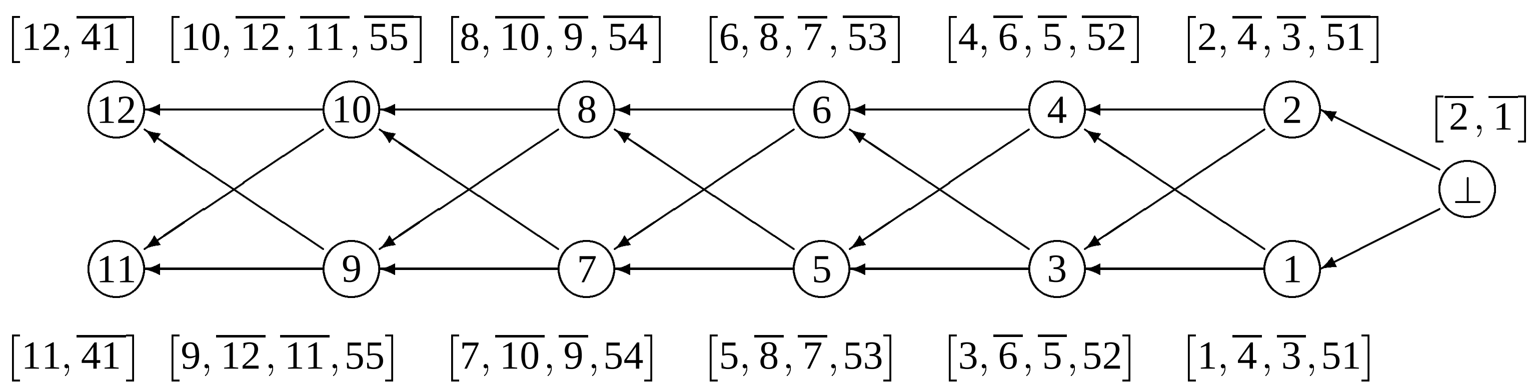
Assume outermost existential 41 is true, implying 11 and 12 at innermost scope.

Now 55 is tailing, allowing 9 and 10 to be implied.

- In each four-literal clause the two negative existential literals "block" the universal literal.
- After they are falsified by unit-clause propagation, the universal literal can be reduced, yielding a new implied existential literal.

This pattern continues until $[\overline{2}, \overline{1}]$ is falsified.

Exponential Case for Usual Clause Learning Procedure, Part 2



Learning Scheme

- Try to resolve out most recently assigned (i.e., trail latest) existential.
- If tautology, resolve out innermost quantifier scope (max qdepth).

Walk through shows 11 and 12 get resolved out 2^k times.

Exponential Case for Usual Clause Learning Procedure, Part 3

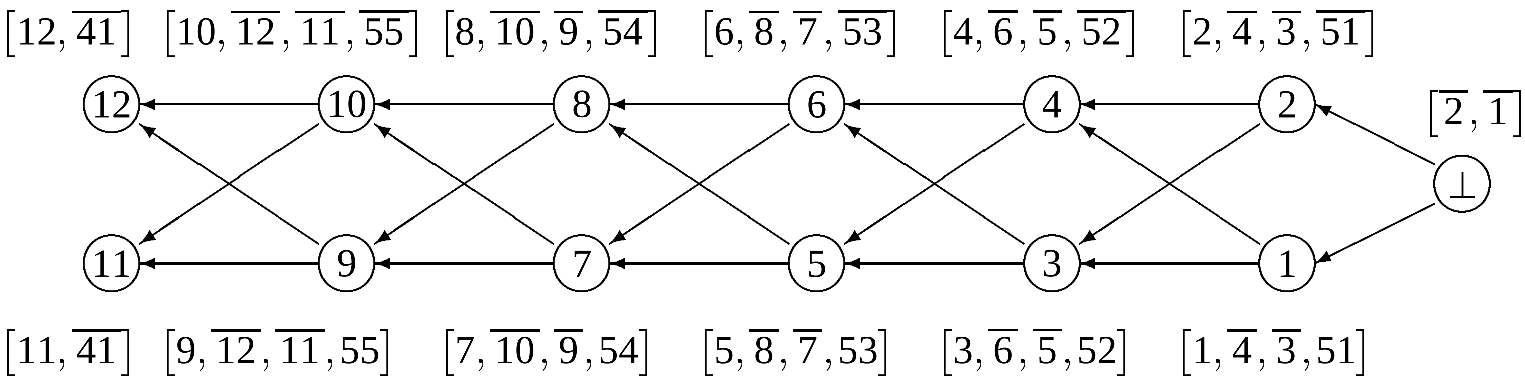
Running times in seconds on *qdpllexp* family

family index	18	19	20	21	22	23
`QuBE 1.3`	10	22	47	105	segv	segv
`depQBF 0.1`	8	16	32	69	140	298
`CirQit3.15`	1	1	3	5	11	21

Running times in seconds on a tougher version of *qdpllexp* family

family index	18	19	20	21	22	23
`QuBE 1.3`	>5hr	>5hr	>5hr	>5hr	memout	memout
`depQBF 0.1`	175	365	777	1606	3364	6934
`CirQit3.15`	9	17	33	67	135	267

An Alternative: QBF Pseudo-Unit Propagation



$\text{qpup}(12) = [12, \overline{41}]$
$\text{qpup}(11) = [11, \overline{41}]$
$\text{qpup}(10) = [10, \overline{41}]$
$\text{qpup}(9) = [9, \overline{41}]$
$\ldots$
$\text{qpup}(\perp) = [\overline{41}]$

Last is the learned clause.

In general, the learned clause has negations of some of the assumptions.

Making QPUP Practical: a Fuzzy idea

- Find a *safe* UIP literal.
- Treat assignments at lower decision levels as assumptions.
- Make latest assumption the safe UIP literal.
- Do QPUP from there through the falsified clause.

Safe means: Since the UIP will be in all derived clauses it should not block any universal reductions.

The *most recent* existential assumed literal is a safe UIP.

Complications:

- Unit clauses with large qdepth (very inner scopes)
- Decision levels with Universal assumed literals.
- Existential pure literals
- Universal pure literals
- Universal “implied” literals — from unit cubes
- Oh No! *Dependency Schemes*.

Existential Pure Literals

These are *not* logically implied from the assumptions.

So, treat as a new assumption.

However, *never* let it be the UIP literal for learning

- Pretend it was assigned at a lower decision level; choose something else.

Theorem

If e is existential pure based on original clauses and ...

If $\overline{e}$ is in a learned clause, say C, then ...

then $C - \overline{e}$ is also logically implied by the original formula (as restricted at the time that e became pure).

An existential pure literal cannot have a quadrangle dependency on any universal literal, so it can move scopes without changing the truth value of the formula.

Universal Pure Literals

These are *not* logically implied from the assumptions.

So, treat as a universal reductions (i.e., clause by clause).

Justification:
No existential literal can have a quadrangle dependency on any universal pure literal, so the universal pure literal can "sink" to innermost scope without changing the truth value of the formula.

Depth-Monotonic Literals

See the proceedings.

Conclusion

Theory is a lot easier than implementation.

Useful theory should make implementation easier.