Electronic Colloquium on Computational Complexity, Report No. 159 (2020)

Relating existing powerful proof systems for QBF

Leroy Chew and Marijn J.H. Heule

Carnegie Mellon University, Pittsburgh, USA

Abstract. We advance the theory of QBF proof systems by showing the first simulation of the universal checking format QRAT by a theory-friendly system. We show that the sequent system G fully p-simulates QRAT, including the Extended Universal Reduction (EUR) rule which was recently used to show QRAT does not have strategy extraction. Because EUR heavily uses resolution paths our technique also brings resolution path dependency and sequent systems closer together. While we do not recommend G for practical applications this work can potentially show what features are needed for a new QBF checking format stronger than QRAT.

1 Introduction

Various applications can be naturally expressed as quantified Boolean formulas (QBF) and QBF solvers have become powerful tools in recent years. However different solvers act in radically different ways, thus universally verifying the results of these solvers is difficult but highly desired. The proof system QRAT has been proposed as a universal checking format for QBF solvers and preprocessors. However, while QRAT appears to be strong enough for many modern techniques [3, 11, 13], it was shown that unless $\mathsf{P} = \mathsf{PSPACE}$ it is asymmetrical on true and false QBFs [4]. While the asymmetry is not as serious as an unconditional lower bound, it does make us question the longevity of the format.

In order to fix this we must look for alternatives, but we do not want to sacrifice any of QRAT's strengths, especially QRAT's short proofs for formulas with PSPACE-hard strategies. This unfortunately makes it hard (PSPACE-hard in fact) for most other QBF proof systems to simulate QRAT. Nonetheless, in this paper we find a proof system able to capture QRAT's full power, that does not share QRAT's asymmetry problems. We show a p-simulation of QRAT by a theoretical sequent calculus, known as G, created by Krajíček and Pudlák [17].

It was surprising that QRAT was this powerful to begin with. Most QBF systems have a property known as *strategy extraction*: There exist polynomial time algorithms to extract from a proof a circuit that tells us how a quantified variable should be played it order to witness the QBF as true or false. The QBF proof system extended Frege+$\forall$-Red was shown to have a conditional optimality *among* QBF proof systems with strategy extraction [2]. The propositional versions of QRAT and extended Frege+$\forall$-Red, known as DRAT and extended Frege [12], are p-equivalent. Yet QRAT unexpectedly ended up not having strategy extraction

ISSN 1433-8092

(unless $\mathsf{P} = \mathsf{PSPACE}$) [4], showing that extended Frege+$\forall$-Red is not powerful enough to simulate QRAT.

G is powerful enough, because it sacrifices automatability for a high degree of non-determinism. For example, let A, B, and C be three QBFs. If $A \vdash B$ and $B \vdash C$ are known implications, G allows you to "cut" the QBF B and derive $A \vdash C$. Knowing which B should be cut is very difficult to automate, because the scope ranges over all QBFs. While other QBF systems use a cut, G is the only known QBF system that allows cut objects to be QBFs themselves. For example, the resolution rules cuts over a variable. As a consequence, it is a weaker proof system that is easier to automate.

Our simulation of QRAT also depends on G's quantifier introduction rule, which, just like the cut rule, uses a QBF that is removed from the formula. This QBF witnesses a new quantifier, which is added to the derivation. In other systems this is function is performed by reduction rules [5] such as the one found in Extended Frege+$\forall$-Red. However, that reduction rule only has witnesses on the level of propositional circuits [1], not on QBFs. G's QBF witnesses are needed when dealing with QRAT.

Unlike G, QRAT has no way of using a QBF witness. In terms of cut rules, QRAT seems to be of intermediate strength as it would appear that the non-deterministic objects are propositional circuits. In this regard, QRAT is similar to Extended Frege+$\forall$-Red. However, QRAT uses a stronger universal reduction rule compared to Extended Frege+$\forall$-Red: extended universal reduction (EUR).

This rule is the culprit as to why QRAT breaks strategy extraction. We give it the most attention in our simulation argument. Without EUR, QRAT has efficient strategy extraction for false QBFs. Strategy extraction for true QBF is always possible in QRAT since EUR ends up being useless for true formulas [7]. This is precisely the reason why QRAT is asymmetric on true and false. It is also the reason that every other rule except EUR can be simulated by using a strategy extraction technique, writing the circuit construction for QRAT rules explicitly in G and then formally proving them. But for the hardest part, simulating the EUR rule, it is strictly necessary to use all the QBF level non-determinism that G can manage.

EUR works by utilising the theory of *dependency schemes*, which helps alleviate some of the rigidity when dealing with how quantifiers are ordered. The relationship between dependency schemes and other QBF techniques is somewhat mysterious, and we hope that our result also sheds some light on these. Any simulation proof using a sound calculus is automatically a soundness argument and therefore we show another soundness proof for QRAT. This means that our simulation on the EUR rule ends up formalising how the dependencies

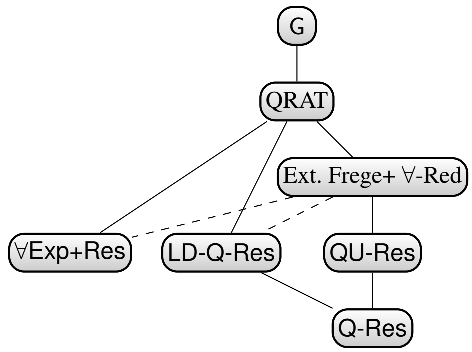

Fig. 1. Simulation structure for QBF calculi, a dashed line indicates a conjectured simulation has not been found, the lack of one implies a conditional or unconditional lower bound

work. In particular we provide a new soundness idea for dependency schemes using *resolution paths*, which is what EUR uses.

2 Overview of contributions

In this paper, we show that G simulates QRAT, which by transitivity can simulate other systems (see Figure 1). We believe this sets the most important condition for future universal QBF checking formats. If we want f to be our next universal checking format, then ideally f should p-simulate G.

While G has many rules, many of them are straightforward and do little more than represent the definition of Boolean operations. Using these rules to capture the complex reasoning in QRAT, requires some work. Essentially, another soundness argument has to be made for QRAT but formalised entirely in G. Our p-simulation proof therefore takes up the entirety of this paper. There are, however, some fundamental ideas that allow the proof of p-simulation to happen, which we will mention here.

2.1 Simulation by strategy extraction.

Many QBF proof systems have the strategy extraction property for Skolem or Herbrand functions. If you have a proof system f and a proof system g that has strategy extraction, then one method for proving that f p-simulates g is to take a g-proof, extract the circuits via strategy extraction and then construct an f-proof validating the circuits as witnesses. This technique first saw use putting extended Frege+ $\forall$-Red into a normal form [1].

This is the idea behind our G p-simulation of QRAT's rules. In Section 4, we use the strategy extraction procedures from [3, 7] to observe how the use of each

QRAT rule builds a strategy circuit. We use this strategy building technique to give us two main theorems:

Theorem 1. *Given a CNF ϕ closed under prefix Π. Suppose that the QRAT rules QRATA or ATA can add C to ϕ, or the QRAT rules QRATE or ATE can remove C from $\phi \wedge C$. Then the sequent $\Pi\phi \vdash \Pi'\phi \wedge C$ has a polynomial size* G *proof. Where Π' contains all variables from Π and any additional variables from C.*

Theorem 2. *Given a CNF $\psi = \phi \wedge (C \vee l)$ closed under prefix Π. Suppose that the QRAT rule QRATU can reduce $C \vee l$ to C, where C is a clause and l is a literal. Then the sequent $\Pi\phi \wedge (C \vee l) \vdash \Pi\phi \wedge C$ has a polynomial size* G *proof.*

2.2 Formalising independence.

Formalising strategy extraction does not work for the only remaining rule EUR. Instead, in order to simulate EUR it is necessary to formalise what makes EUR sound, namely independence. In Section 5 we do exactly this, we come up with a sequent that represents the independence that EUR uses and then show that it has a short G proof. These sequents of Theorem 3 use QBFs and most of the work here is using the dependency scheme to come up with the correct witnesses to introduce the quantifiers.

Theorem 3. *For any CNF ϕ with a subset χ_1 and let $\mathfrak{C}$ denote the set of all clauses of the resolution paths from χ_1, in the existential literals of a prefix Π.* G *can prove $\Pi(\phi \backslash \chi_1), \Pi \bigwedge_{D \in \mathfrak{C}} D \vdash \Pi\phi$ in a polynomial size proof.*

Using QBF witnesses for EUR. While strategy extraction for circuits is not possible for EUR, EUR still preserves whether a QBF is true or false and therefore whether Skolem or Herbrand functions exist. Instead of expressing the strategies via propositional circuits and using them to create witnesses, we create witnesses out of QBFs. Using QBFs instead of circuits is adequate for our G proofs because we can make cuts and instantiations with QBFs. In Section 5.2 we find the correct QBF witnesses and we can cut with the sequent from Theorem 3 to show that this gives us p-simulation of EUR.

Theorem 4. *Let ϕ be a CNF with Π a prefix. Suppose that the QRAT rule QRATU can reduce clause $C \vee l$ to C, where C is a clause and l is a literal. Then the sequent $\Pi\phi \wedge (C \vee l) \vdash \Pi\phi \wedge C$ has a polynomial size* G *proof.*

And finally, our main theorem is that G p-simulates QRAT. In other words, if QRAT has a proof that QBF Ψ is true, one can construct a G proof of sequent $\vdash \Psi$ and if QRAT has a proof that Ψ is false, one can construct a G proof of sequent $\Psi \vdash$.

Theorem 5. *G p-simulates QRAT.*

The proof of this follows directly from the short proofs from the various theorems as one can use the cut rule to chain all the sequents together.

3 Preliminaries

Quantified Boolean formulas. Quantified Boolean formulas (QBF) extend propositional logic with quantifiers $\forall, \exists$ that work on propositional atoms [14]. We use notation $A[x/y]$ to replace all instances of term y with term x in A. The standard QBF semantics is that $\forall x \Psi$ is satisfied by the same Boolean assignments as $\Psi[0/x] \wedge \Psi[1/x]$ and $\exists x \Psi$ is satisfied by the same Boolean assignments as $\Psi[0/x] \vee \Psi[1/x]$.

For QRAT, we consider QBFs in prenex normal form $\Pi\phi$ with ϕ being a conjunction of clauses. The prefix Π is arranged in a linear order (we use $x <_\Pi y$ to denote x is left of y). For prefixes Π and Π' let $\Pi \subseteq \Pi'$ mean for every variable $\exists x$ in Π, $\exists x$ is in Π', and for every variable $\forall y$ in Π, $\forall y$ is in Π'. And if a and b are variables in Π with $a \leq_\Pi b$ then $a \leq_{\Pi'} b$.

If the prefix Π quantifies all variables in ϕ, then we say $\Pi\phi$ is *closed*. A closed prenex QBF may be thought of as a game between two players. One player is responsible for assigning values to the existentially quantified variables, and the other responsible for the universally quantified variables. The existential player wins the game if the formula evaluates to true once all assignments have been made, the universal player wins if the formula evaluates to false. The players take turns to make assignments according to the quantifier prefix, so the order of the prefix dictates the turns of the game.

A strategy for the universal player on QBF $\Pi\phi$ is a method for choosing assignments for each universal u that depends only on variables earlier than u in Π. For each individual u we call a function that gives a winning strategy for the universal player a *Herbrand function*. The dual concept for the existential player is the *Skolem function*.

Clausal proofs. A *proof system* is a polynomial time function that maps proofs to theorems. Proof system f is said to p-simulate proof system g if there is a polynomial time mapping τ from g proofs to f proofs such that for each g-proof π, $f(\tau(\pi)) = g(\pi)$.

In propositional logic, a literal is a variable (x) or its negation ($\neg x$), a clause is a disjunction of literals and a formula in conjunctive normal form (CNF) is a conjunction of clauses. For a literal l, we denote its basic variable as $\mathrm{var}(l)$, if $l = \mathrm{var}(l)$ then $\bar{l} = \neg\mathrm{var}(l)$, and if $l = \neg\mathrm{var}(l)$ then $\bar{l} = \mathrm{var}(l)$. For a clause C, $\bar{C}$ represents the conjunction $\bigwedge_{c \in C} \bar{c}$, each $\bar{c}$ can be thought of as a singleton clause. It is natural to understand a CNF as a set of clauses, and a clause as a set of literals. As such we will use notation $C \in \phi$ to indicate that CNF ϕ has the clause C in its matrix. Similarly $l \in C$ indicates that clause C contains literal l. Set notation is also used to define sub-clauses and sub-formulas.

Unit propagation. Unit propagation simplifies a CNF ϕ by building a partial assignment and applying it to ϕ. It builds the assignment by satisfying any literal that appears in a singleton (unit) clause. Doing so may negate opposite literals in other clauses and result in them effectively being removed from that clause. In this way, unit propagation can create more unit clauses and can keep on propagating until no more unit clauses are left. We denote by $\phi \vdash_1 \bot$ that unit propagation derives the empty clause from ϕ. Unit propagation is used heavily to check the rules of DRAT and QRAT.

3.1 The rules of QRAT

Definition 6. *Fix a prefix Π, assume that Π is strictly ordered. Now consider a clause D and a literal l (not necessarily in D) we define, $O^l_D = \{k \in D \mid k <_\Pi l, k \in D\}, I^l_D = \{k \in D \mid k >_\Pi l, k \in D\}$. O^l_D is called the* outer clause *and I^l_D is called the* inner clause.

The first rule ATA/ATE is a simple propositional implication using unit propagation.

Definition 7 (Asymmetric Tautology Addition/Elimination (ATA)/(ATE)).
Let ϕ be a CNF with Π a prefix. Let C be a clause not in ϕ. Let Π' be a prefix including the variables of C and ϕ, $\Pi \subset \Pi'$.
Suppose $\phi \wedge \bar{C} \vdash_1 \bot$. Then we can make the following inferences.

$$\frac{\Pi\phi}{\Pi'\phi \wedge C}\,(ATA) \qquad\qquad \frac{\Pi\phi \wedge C}{\Pi\phi}\,(ATE)$$

The next rules, QRATA and QRATE, deal with adding or removing a clause, but this time the Skolem function for a particular existential literal l changes as a result of this rule. This means that QRATA and QRATE preserve truth but do not necessarily preserve the strategies.

Definition 8 (Quantified Resolution Asymmetric Tautology Addition/Elimination (QRATA)).
Let $\Pi\phi$ be a PCNF with closed prefix Π and CNF matrix ϕ. Let C be a clause not in ϕ. Let Π_1 and Π_2 be disjoint prefixes and x a variable such that $\Pi \subseteq \Pi_1 \exists x \Pi_2$. The difference in prefix is simply to allow new variables.
If there is existential literal l, with $\mathrm{var}(l) = x$ such that for every $D \in \phi$ with $\bar{l} \in D$, $\phi \wedge \bar{C} \wedge \bar{l} \wedge \bar{O}^l_D \vdash_1 \bot$, then we can derive:

$$\frac{\Pi\phi}{\Pi_1\exists x\Pi_2\phi\wedge(C\vee l)}\textit{(QRATA)}\qquad\frac{\Pi_1\exists x\Pi_2\phi\wedge(C\vee l)}{\Pi_1\exists x\Pi_2\phi}\textit{(QRATE)}$$

Example 9. $\forall x\exists y(x\vee y)\wedge(\bar{x}\vee\bar{y})$ is true, so it has a QRAT proof. To prove $\forall x\exists y(x\vee y)\wedge(\bar{x}\vee\bar{y})$ in QRAT we need to remove all the clauses. QRATE can remove $(\bar{x}\vee\bar{y})$ wrt to literal $\bar{y}$ as the only clause with y in it is $(x\vee y)$ and the condition $(x\vee y)\wedge\bar{x}\wedge x\wedge y\vdash_1\bot$ holds. When $(x\vee y)$ is the only clause left we can use QRATE wrt to literal y since the condition is vacuously true (there are no clauses left with $\bar{y}$ in them). We are left with the empty CNF which confirms our starting QBF true.

The next rule QRATU removes a literal from a clause, the condition is similar to that of QRATA/QRATE but uses a universal literal instead of an existential one.

Definition 10 (Quantified Resolution Asymmetric Tautology Universal (QRATU)).

Let $\Pi_1\exists x\Pi_2\phi$ be a PCNF with closed prefix $\Pi_1\forall u\Pi_2$ and CNF matrix ϕ. Let $C\vee l$ be a clause with universal literal l, with $\mathrm{var}(l)=u$.

If for every $D\in\phi$ with $\bar{l}\in D$, $\phi\wedge\bar{C}\wedge\bar{O}^l_D\vdash_1\bot$, then we can derive

$$\frac{\Pi_1\forall x\Pi_2\phi\wedge(C\vee l)}{\Pi_1\forall x\Pi_2\phi\wedge C}\textit{(QRATU w.r.t. l)}$$

The definition of the final rule: Extended Universal Reduction (EUR) is based on the resolution paths (see Definition 26, in Section 5). Informally, a resolution path is a path in the graph with vertices that are clauses, where an edge indicates that a variable in a set $\mathcal{S}$ can be resolved between the two clauses. Originally, it was defined using the version of resolution path that allowed the pivot variable to be immediately re-used. This is in fact weaker than in Definition 11, as we get more dependencies. But Definition 11 is in line with the intention of EUR which is to exploit independence to make reductions.

Definition 11. *Let $\Pi_1\exists x\Pi_2\phi$ be a PCNF with closed prefix $\Pi_1\forall u\Pi_2$ and CNF matrix ϕ. Let $C\vee l$ be a clause with universal literal l, with $\mathrm{var}(l)=u$.*

If the resolution path $\mathfrak{C}(\phi\wedge C,C,\mathcal{S})$ (see Definition 26 later in Section 5) contains no clause D such that $\bar{l}\in D$, when $\mathcal{S}$ is the set of existential variables right of l in the prefix (i.e. in Π_2), then we can derive

$$\frac{\Pi_1\forall l\Pi_2\phi\wedge(C\vee l)}{\Pi_1\forall l\Pi_2(\phi\wedge C)}\textit{(EUR)}$$

A *QRAT search* starts with a closed prenex CNF Ψ and uses the six QRAT rules to modify the QBF. A search is a proof of the truth of Ψ if it removes all clauses and we are left with an empty CNF and proves the falsity of Ψ if it adds an empty clause. The six rules are only required in search mode, once we have determined whether a QBF is true or false the rules can be relaxed. QRAT proofs of truth are allowed to add any clause arbitrarily, and QRAT proofs of falsify are allowed to arbitrarily delete a clause [8].

Example 12. Take the false QBF $\exists x \forall u \exists y (x \vee u \vee y) \wedge (\bar{x} \vee \bar{u} \vee y) \wedge (\bar{y})$. There are no resolution paths in the variables right of u that connect clauses $(x \vee u \vee y)$ and $(\bar{x} \vee \bar{u} \vee y)$. The only paths from each join to $(\bar{y})$ but are unable to reuse the same literal to connect the opposing clause. This means that $(x \vee u \vee y)$ can be reduced to $(x \vee y)$ via EUR and then $(\bar{x} \vee \bar{u} \vee y)$ can be reduced to $(\bar{x} \vee y)$ by the same argument. The empty clause can be added with the ATA rule as $(x \vee y) \wedge (\bar{x} \vee y) \wedge (\bar{y}) \vdash_1 \bot$.

3.2 The sequent system G

Let Γ and Δ each be sets of logical formulas[1]. A *sequent* $\Gamma \vdash \Delta$ expresses that any Boolean assignment that satisfies every formula in Γ also satisfies at least one formula in Δ. Sequents can be used for propositional logic, first order logic and QBF. In QBF we have to be careful about how we talk about assignments, because there are many examples in the QBF literature where assignments are over bound variables. When we are talking about how sequents work, the assignments ignore bound variables, $\forall u \Psi(u)$ has the same satisfying assignments as $\Psi(0) \wedge \Psi(1)$, the variable u is ignored.

In a G sequent $\Gamma \vdash \Delta$. Γ and Δ are sets of QBFs, note here that these QBFs are not necessarily in prenex form and they are also not necessarily closed, so they may contain a mix of bound and free variables. The rules of G are given in Figure 2.

The main difference between G and propositional sequent calculi are the quantifier rules $(\exists \vdash), (\vdash \exists), (\forall \vdash)$ and $(\vdash \forall)$. The rules $(\vdash \exists)$ and $(\forall \vdash)$ are the most flexible, allowing you to replace a term (which can be expressed by a QBF) with a quantified variable. We include the proviso that if the term contains any free variables these variables cannot be bound elsewhere [16]. Finding the best term to replace is crucial to the simulation argument used in this paper. The rules $(\exists \vdash)$ and $(\vdash \forall)$ are stricter, not only is one only allowed to substitute for a variable p, but once bound that variable must disappear entirely. This often

[1] Classically, sequents work on ordered multisets, but have exchange and contraction rules that make it the same as sets. The multiset version adds only polynomially many lines to derivations and we are interested in polynomial simulation, so we present a p-equivalent system here.

$$\frac{}{A \vdash A}\ (\vdash) \qquad \frac{}{\bot \vdash}\ (\bot \vdash) \qquad \frac{}{\vdash \top}\ (\vdash \top)$$

$$\frac{\Gamma \vdash \Sigma}{\Delta, \Gamma \vdash \Sigma}\ (\bullet \vdash) \qquad \frac{\Gamma \vdash \Sigma}{\Gamma \vdash \Sigma, \Delta}\ (\vdash \bullet)$$

$$\frac{\Gamma \vdash \Sigma, A}{\neg A, \Gamma \vdash \Sigma}\ (\neg \vdash) \qquad \frac{A, \Gamma \vdash \Sigma}{\Gamma \vdash \Sigma, \neg A}\ (\vdash \neg)$$

$$\frac{A, \Gamma \vdash \Sigma}{B \wedge A, \Gamma \vdash \Sigma}\ (\bullet\wedge \vdash) \qquad \frac{A, \Gamma \vdash \Sigma}{A \wedge B, \Gamma \vdash \Sigma}\ (\wedge\bullet \vdash)$$

$$\frac{\Gamma \vdash \Sigma, A \qquad \Lambda \vdash \Delta, B}{\Gamma, \Lambda \vdash \Sigma, \Delta, A \wedge B}\ (\vdash \wedge)$$

$$\frac{\Gamma \vdash \Sigma, A}{\Gamma \vdash \Sigma, B \vee A}\ (\vdash \bullet\vee) \qquad \frac{\Gamma \vdash \Sigma, A}{\Gamma \vdash \Sigma, A \vee B}\ (\vdash \vee\bullet)$$

$$\frac{A, \Gamma \vdash \Sigma \qquad B, \Lambda \vdash \Delta}{A \vee B, \Gamma, \Lambda \vdash \Sigma, \Delta}\ (\vee \vdash)$$

$$\frac{\Gamma \vdash \Sigma, A \qquad A, \Lambda \vdash \Delta}{\Gamma, \Lambda \vdash \Sigma, \Delta}\ (\text{cut})$$

$$\frac{A(B), \Gamma \vdash \Sigma}{\forall x A(x), \Gamma \vdash \Sigma}\ (\forall \vdash) \qquad \frac{\Gamma, \vdash \Sigma, A(p)}{\Gamma \vdash \Sigma, \forall x A(x)}\ (\vdash \forall)$$

$$\frac{A(p), \Gamma \vdash \Sigma}{\exists x A(x), \Gamma \vdash \Sigma}\ (\exists \vdash) \qquad \frac{\Gamma, \vdash \Sigma, A(B)}{\Gamma, \vdash \Sigma, \exists x A(x)}\ (\vdash \exists)$$

A, B are QBFs and $\Gamma, \Lambda, \Sigma, \Delta$ are sets of QBFs,
Variable p does not appear free on the lower sequents in ($\exists \vdash$), ($\vdash \forall$).
The free variables of B are not bound in A in ($\forall \vdash$), ($\vdash \exists$).

Fig. 2. Rules of the sequent calculus G [17]

means that the other G rules (such as the other quantifier rules) need to be applied first to remove instances of p from the other side of the sequent.

Example 13. The QBF $\forall x \exists y (x \vee y) \wedge (\neg x \vee \neg y)$ is true as seen in Example 12. There are no free variables, but we nevertheless understand it to be true under all assignments. The sequent $\vdash \forall x \exists y (x \vee y) \wedge (\neg x \vee \neg y)$ represents this and can be proved from the rules of G. We can start with axiom $z \vdash z$ and use the negation and disjunction rules in LK to get sequent $\vdash z \vee \neg z$ and similarly we can get $\vdash \neg z \vee \neg\neg z$, LK has a conjunction rule to put these together and we can continue in G.

$$\dfrac{\dfrac{\vdash (z \vee \neg z) \wedge (\neg z \vee \neg\neg z)}{\vdash \exists y (z \vee y) \wedge (\neg z \vee \neg y)}\,(\vdash \exists)}{\vdash \forall x \exists y (x \vee y) \wedge (\neg x \vee \neg y)}\,(\vdash \forall)$$

As we can observe, this proof is cut-free. This means it has to build up the sequents starting from the innermost connectives, working its way outward. Because this formula is small without many variables, the proof is also small, but cut becomes more practical in larger more complicated formulas.

The most important thing to notice about this proof is that $(\neg z)$ is used as the witness in $(\vdash \exists)$ to quantify y. Since there is only one QBF and it is on the right hand side, the witness also tells us the Skolem function. For $(\vdash \exists)$ we do not have to quantify all instances of $\neg z$ into y.

After we apply $(\vdash \exists)$, we have sequent $\vdash \exists y (z \vee y) \wedge (\neg z \vee \neg y)$. This sequent should be read as: in all assignments to the free variables (z is only free variable left), the sequent $\vdash \exists y (z \vee y) \wedge (\neg z \vee \neg y)$ is true. It is intuitive to see how $(\vdash \forall)$ soundly applies here, replacing z with x and giving us the final QBF. In this example we used variable z to eventually become the variable x. In later examples and proofs, to avoid renaming every variable we will sometimes use the same symbols for variables before and after they are quantified.

Example 14. Given a set of free variables X suppose we have a CNF $\phi(X)$ and we quantify the X variables with a prefix Π. Consider a set of variables X' with $|X'| = |X|$ and then let Π' be the X' version of Π. Similarly define X'' and Π''.

$\Pi\phi(X) \vdash \Pi'\phi(X')$ can be proven in G by starting with $\phi(X) \vdash \phi(X)$ and adding the X variables when quantifying the left hand side variables and X' variables when quantifying the right hand side variables. $\Pi''\phi(X'') \vdash \Pi'\phi(X')$ and $\Pi\phi(X) \vdash \Pi''\phi(X'')$ can be proved in a similar way and through G's connective rules we can get sequent $\Pi\phi(X) \vee \Pi''\phi(X'') \vdash \Pi'\phi(X') \wedge \Pi''\phi(X'')$. This allows the sequent to be expressed entirely on the left hand side as $(\Pi\phi(X) \vee \Pi''\phi(X'')) \wedge (\neg\Pi'\phi(X') \vee \neg\Pi''\phi(X'')) \vdash$. This sequent expresses little more than the law of non-contradiction for QBF but we can add more quantifiers to make it interesting.

We can use ($\forall \vdash$) with witness $\Pi''\phi(X'')$ to turn it into the universal variable z to get $\forall z(\Pi\phi(X) \vee z) \wedge (\neg\Pi'\phi(X') \vee \neg z) \vdash$. Expressed in PCNF (and G is able to change to this PCNF) this becomes a instance of the Select family which have PSPACE-hard strategies [4]. The formulas looks like $\forall z \Pi \bar{\Pi}' \exists T(\phi(X) \vee z) \wedge (\bar{\phi}(X', T) \vee \neg z)$. $\bar{\Pi}'$ switches the quantifiers used but retains the same order and $\bar{\phi}(X', T)$ expresses $\neg\phi(X')$ using Tseitin variables T. The $\vee z$ and $\vee\neg z$ are distributed throughout all the clauses in order to make this a PCNF.

Every Select formula has a short refutation in QRAT. The way do that is to reduce z and $\neg z$ literals. Since the $\phi(X)$ clauses and $\bar{\phi}(X', T)$ clauses do not share any literals there is no resolution path between them and every z and $\neg z$ can be reduced with EUR. What we are left with is a formula in the Duality family which has short refutations in Extended Frege+$\forall$-Red which QRAT is able to simulate [4].

Even for very basic tautologies G proofs require many lines, and we will see in our simulation that although the simulation is polynomial, it uses considerably more lines. To simplify our explanation, and avoid reinventing the wheel, we omit certain steps, particularly in propositional logics as we are focused mainly on QBF.

Lemma 15. *The following substitutions can be made in short G proofs, based on logical equivalence laws:*

- *Double negation*
- *De Morgan's laws*
- *Distributive laws*
- *We can treat ',' on the left part of a sequent as interchangeable with '$\wedge$'*
- *We can treat',' on the right part of a sequent as interchangeable with '$\vee$'*

Proof. We can get these rules from the known power of LK, the propositional fragment of G. LK is known to p-simulate Frege systems [16]. And the laws of equivalence can be used as axioms in a Frege system. $\square$

The next lemmas show us common applications of the quantifier rules.

Lemma 16. *Given QBFs A and B and a prefix $\Pi = \mathcal{Q}_1 x_1, \ldots, \mathcal{Q}_n x_n$ containing variables that may or may not be in A or B. If we can derive the sequent $A \vdash B$ in an m length proof, we can derive the sequent $\Pi A \vdash \Pi B$ in a $O(m + |\Pi|)$ length proof.*

Proof. We define $\Pi_i = \mathcal{Q}_{n-i+1}x_{n-i+1}, \dots, \mathcal{Q}_n x_n$ and we define A_i and B_i in the reverse order starting with $A_n = A$ and $B_n = B$. Let $y_1 \dots y_n$ be propositional variables. We define $A_{i-1} = A_i[y_{n-i+1}/x_{n-i+1}]$ and $B_{i-1} = B_i[y_{n-i+1}/x_{n-i+1}]$.
Induction hypothesis: $\Pi_i A_i \vdash \Pi_i B_i$ has G proof of length $2i + m$.
Base case: When $i = 0$, Π_i is empty so we can use the proof of $A \vdash B$, however we replace the variables in the steps of the proof so that we get $A_0 \vdash B_0$.
Inductive step: If $Q_{n-i} = \exists$, then we apply ($\vdash \exists$) using y_{i+1} as the term that we replace with bound variable x_{i+1} in B_{n-i+1}, now y_{i+1} no longer appears on the right part of the sequent, only appearing on the left part where we can use ($\exists \vdash$) to quantify A_{n-i+1} replacing y_{i+1} with x_{i+1}.

Symmetrically, if $Q_{n-i} = \forall$, then we apply ($\forall \vdash$) using y_{i+1} as the term that we replace with bound variable x_{i+1} in A_{n-i+1}, now y_{i+1} no longer appears on the left part of the sequent, only the right part, where we can use ($\vdash \forall$) to quantify B_{n-i+1} replacing y_{i+1} with x_{i+1}.

Once we reach $i = n$ we get $\Pi A \vdash \Pi B$ and we have only used $2|\Pi| + m$ steps. $\square$

Corollary 17. *For any propositional formulas A and B, and quantifier prefix Π there are short G proofs of $\Pi(A \wedge B) \vdash \Pi A$.*

Lemma 18. *For any QBF ϕ with free variables x and y, the sequent $\Pi \exists x \exists y \phi \vdash \Pi \exists y \exists x \phi$ has a short G proof.*

Proof. We can follow this particular G derivation. The key is that while quantifying the right hand side using ($\vdash \exists$) for x and y, we do not need to interrupt by quantifying the left hand side until a $\forall$ quantifier appears. It is only ($\vdash \forall$) and ($\exists \vdash$) which require the other side be quantified first. $\square$

Lemma 19. *For any QBF ϕ where the variable x does not occur. If $A, B \in \{\Pi \exists x \phi, \Pi \phi, \Pi \forall x \phi\}$ then $A \vdash B$ has a short G proof.*

Proof. We start with sequent $\phi \vdash \phi$ we can use any of ($\vdash \forall$), ($\forall \vdash$), ($\vdash \exists$), ($\exists \vdash$) as x and y do not appear anywhere in ϕ. Then Lemma 16 allows us to add Π. $\square$

4 Using strategies to simulate QRAT rules

In this section we use strategy extraction to show a G simulation of rules ATA, ATE, QRATA, QRATE and QRATU. The final rule EUR does not have strategy extraction. Since, G does not allow empty disjunctions or conjunctions, we treat $\perp$ as the empty disjunction and $\top$ as the empty conjunction.

We break the proofs of simulation up into new lemmas. Because we do not care too much about the order of clauses in a CNF we can afford to be ambiguous to whether $(\bullet\wedge\vdash)$ or $(\wedge\bullet\vdash)$ is used in a proof so we just use $(\wedge\vdash)$ to signify this. Similarly we can use $(\vdash\vee)$ in this way. Firstly we show that how we can turn unit propagation into a proof of a useful sequent in G.

Lemma 20. *If conjunctive normal form formula ϕ can be shown to be contradictory via unit propagation, then the sequent $\phi\vdash$ has a polynomially bounded proof in G. (Recall that an empty right hand side of a sequent is equivalent to the empty disjunction).*

Proof. We can prove this by induction on the number of unit clauses needed to derive a contradiction.
Inductive Hypothesis: If CNF ϕ can be shown to be a contradiction in m many unit propagation steps. There is G proof of sequent $\phi\vdash$ in $O(m)$ many lines.
Base Case: Suppose we reach a contradiction using unit literals x and $\bar{x}$ we can represent this with G sequent $x,\bar{x}\vdash$.

We can now strengthen the left side of the sequent to whatever we want, adding the remaining clauses.
Inductive Step: Suppose we have a CNF ϕ and a unit clause l, divide ϕ into three parts, ϕ_l contains clauses of the form $C\vee l$, $\phi_{\bar{l}}$ contains clauses of the form $C\vee\bar{l}$ and ϕ_0 contains clauses C where $l\notin C$ and $\bar{l}\notin C$. Suppose via the induction hypothesis that $\phi_l,\phi_0,\bigwedge_{C\vee\bar{l}\in\phi_{\bar{l}}}C\vdash$ is proven in G. Then we do the following:

$$\cfrac{\cfrac{\cfrac{\cfrac{\phi_l,\phi_0,\bigwedge_{C\vee\bar{l}\in\phi_{\bar{l}}}C\vdash}{\phi_l,\phi_0,\bigwedge_{C\vee\bar{l}\in\phi_{\bar{l}}}C\vdash\bar{l}}\,(\vdash\bullet)\qquad\cfrac{\bar{l}\vdash\bar{l}}{\phi_l,\phi_0,\bar{l}\vdash\bar{l}}\,(\bullet\vdash)}{\phi_l,\phi_0,\bar{l}\vee\bigwedge_{C\vee\bar{l}\in\phi_{\bar{l}}}C\vdash\bar{l}}\,(\vee\vdash)}{\neg\bar{l},\phi_l,\phi_0,\bar{l}\vee\bigwedge_{C\vee\bar{l}\in\phi_{\bar{l}}}C\vdash}\,(\neg\vdash)}{\phi\vdash}\,(\text{Lemma 15})$$

The QRAT rules modify existing Skolem functions in order to preserve the truth of QBFs when changing the formula. Imagine we already have a Skolem function for existential literal l in a CNF ϕ, let us modify that Skolem function so now it returns true whenever all outer clauses O^l_D for clauses D with $\bar{l}\in D\in\phi$ are true and just play the same in all other cases. What we will show is that G can confirm formally that this will still be a Skolem function.

Lemma 21. *Let ϕ be a CNF and for literal l define $l'=l\vee\bigwedge^{\bar{l}\in D}_{D\in\phi}O^l_D$ where $O^l_D\subseteq D$, $\bar{l}\notin O^l_D$, then $\phi\vdash\phi[l'/l]$.*

Proof. Let us consider each clause in ϕ. There are three cases for the sequents we want to prove. 1. $K \vdash K$ for $l, \bar{l} \notin K$, $K \in \phi$. 2. $K \vee l \vdash K \vee l \vee \bigwedge_{D\in\phi}^{\bar{l}\in D} O_D^l$, for $K \vee l \in \phi$. 3. $K \vee \bar{l} \vdash K \vee \neg(l \vee \bigwedge_{D\in\phi}^{\bar{l}\in D} O_D^l)$, for $K \vee \bar{l} \in \phi$. We now prove each case:

1. Achieved by the axiom in G.
2. We can take $K \vee l \vdash K \vee l$ and weaken the right side with $\bigwedge_{D\in\phi}^{\bar{l}\in D} O_D^l$.
3. We can prove this by a derivation in G.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{O_K^l \vdash O_K^l}{\vdash O_K^l, \neg O_K^l}\,(\vdash \neg)}{\vdash O_K^l \vee \neg O_K^l}\,(\text{Lemma 15})}{K \vee \bar{l} \vdash O_K^l \vee \neg O_K^l}\,(\text{Lemma 15})}{K \vee \bar{l} \vdash K \vee \bigvee_{D\in\phi}^{\bar{l}\in D} \neg O_D^l}\,(\vdash \vee) \qquad K \vee \bar{l} \vdash K \vee \bar{l}}{K \vee \bar{l} \vdash (K \vee \bar{l}) \wedge (K \vee \bigvee_{D\in\phi}^{\bar{l}\in D} \neg O_D^l)}\,(\vdash \wedge)}{K \vee \bar{l} \vdash K \vee (\bar{l} \wedge \bigvee_{D\in\phi}^{\bar{l}\in D} \neg O_D^l)}\,(\text{Lemma 15})}{K \vee \bar{l} \vdash K \vee \neg(l \vee \bigwedge_{D\in\phi}^{\bar{l}\in D} O_D^l)}\,(\text{Lemma 15})$$

Therefore if ϕ is not the empty CNF we can gain the conjunction $\phi \vdash \phi[l'/l]$. If ϕ is the empty CNF then $\bot \vdash \bot$ suffices. $\square$

$l' = l \vee \bigwedge_{D\in\phi}^{\bar{l}\in D} O_D^l$ is actually the modification of the Skolem function that allows QRATA to happen [7]. We show using a G sequent that under the QRATA condition it is sound to add the new clause.

Lemma 22. *Let $l' = l \vee \bigwedge_{D\in\phi}^{\bar{l}\in D} O_D^l$, and for all $D \in \phi$, $\bar{l} \in D$ we have $\phi, \neg(C \vee l \vee O_D^l)$ is a contradiction via unit propagation.* G *can derive a short proof of $\phi \vdash C \vee l'$.*

Proof. For each $D \in \phi$, $\bar{l} \in D$, the sequent $\phi, \bar{C}, \bar{l}, \bar{O}_D^l \vdash$ can be proved in G using Lemma 20. We can use rule $(\vdash \neg)$ and Lemma 15 to get $\phi \vdash C \vee l \vee O_D^l$. If there are some $D \in \phi$, $\bar{l} \in D$ we can take a conjunction, using $(\vdash \wedge)$ to get $\phi \vdash C \vee l \vee \bigwedge_{D\in\phi}^{\bar{l}\in D} O_D^l$, as required.

If ϕ is the empty CNF, there are no $D \in \phi$ such that $\bar{l} \in D$. If there are no $D \in \phi$ such that $\bar{l} \in D$, then $l' = l \vee \top$, so instead we start with $\vdash \top$ weakening the right hand side and strengthening the left hand side to get $\phi \vdash C \vee l \vee \top$. $\square$

Theorem 1 *Given a CNF ϕ closed under prefix Π. Suppose that the QRAT rules QRATA/ATA can add C to ϕ, or the QRAT rules QRATE/ATE can remove C from*

$\phi \wedge C$. Then the sequent $\Pi\phi \vdash \Pi'\phi \wedge C$ has a polynomial size G *proof. Where Π' contains all variables from Π and any additional variables from C.*

Proof. Suppose that C is added via ATA or removed via ATE, this means that the unit propagation $\phi, \bar{C} \vdash_1 \bot$ holds. Using Lemma 20 gives a short G proof of $\phi, \bar{C} \vdash$.

We then continue using propositional rules to get $\phi \vdash \phi \wedge C$ and Lemma 16 to get $\Pi\phi \vdash \Pi\phi \wedge C$.

$$\dfrac{\dfrac{\dfrac{\dfrac{\phi, \bar{C} \vdash}{\phi \vdash \neg\bar{C}}\ (\vdash \neg)}{\phi \vdash C}\ (\text{Lemma 15}) \qquad \phi \vdash \phi}{\phi \vdash \phi \wedge C}\ (\vdash \wedge)}{\Pi\phi \vdash \Pi\phi \wedge C}\ (\text{Lemma 16})$$

Suppose that $C = C' \vee l$ is added via QRATA or removed via QRATE and also suppose there is existential literal l, with $\mathrm{var}(l) = x$ such that for every $D \in \phi$ with $\bar{l} \in D$, $\phi \wedge \bar{C} \wedge \bar{O}^l_D \vdash_1 \bot$ Let $\Pi' = \Pi_1 \exists x \Pi_2$. Then the sequent we need to prove is $\Pi_1 \exists x \Pi_2 \phi \vdash \Pi_1 \exists x \Pi_2 \phi \wedge (C' \vee l)$.

Let $l' = l \vee \bigwedge_{D \ni \bar{l}} O^l_D$ using the definition of outer clauses. We will eventually use l' as a witness for $(\vdash \exists)$ in G. But firstly, we can use Lemmas 21 and 22 to get $\phi \vdash \phi[l'/l]$ and $\phi \vdash (C \vee l')$ in a short proof. We can then proceed in a G proof utilising Lemma 16.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\phi \vdash \phi[l'/l] \qquad \phi \vdash C' \vee l'}{\phi \vdash \phi[l'/l] \wedge (C' \vee l')}\ (\vdash \wedge)}{\Pi_2\phi \vdash \Pi_2\phi[l'/l] \wedge (C' \vee l')}\ (\text{Lemma 16})}{\Pi_2\phi \vdash \exists x \Pi_2 \phi \wedge (C' \vee l)}\ (\vdash \exists)}{\exists x \Pi_2 \phi \vdash \exists x \Pi_2 \phi \wedge (C' \vee l)}\ (\exists \vdash)}{\Pi_1 \exists x \Pi_2 \phi \vdash \Pi_1 \exists x \Pi_2 \phi \wedge (C' \vee l)}\ (\text{Lemma 16})$$

When using QRATE and ATE $\Pi' = \Pi$ but for QRATA and ATA C could contain variables not in Π. However we can derive $\Pi\phi \vdash \Pi'\phi$ using Lemma 19 and then use the cut rule. ☐

Example 23. Suppose we have QBF $\forall x \exists y(\neg x \vee \neg y)$ and we want to add clause $(x \vee y)$. In QRAT this is a single line. In G the simulation given by Theorem 1 is as follows (note we will not detail derivations using Lemma 15).

We use $y' = y \vee x$. So first we show Lemma 21 that the existing clause $\neg x \vee \neg y$ works under this change.

$$
\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\neg x \vdash \neg x}{\vdash \neg x, \neg\neg x}\,(\vdash \neg)}{\vdash \neg x \vee \neg\neg x}\,(\text{L. } 15)}{\neg x \vee \neg y \vdash \neg x \vee \neg\neg x}\,(\bullet \vdash) \qquad \neg x \vee \neg y \vdash \neg x \vee \neg y}{\neg x \vee \neg y \vdash (\neg x \vee \neg y) \wedge (\neg x \vee \neg\neg x)}\,(\vdash \wedge)}{\cfrac{\neg x \vee \neg y \vdash \neg x \vee (\neg y \wedge \neg\neg x)}{\neg x \vee \neg y \vdash \neg x \vee \neg(y \vee \neg x)}\,(\text{L. } 15)}\,(\text{L. } 15)
$$

Next we show Lemma 22 that the new clause $x \vee y$ is implied by this substitution.

$$
\cfrac{\cfrac{\cfrac{\cfrac{\neg x \vdash \neg x}{\neg x, \neg\neg x \vdash}\,(\neg \vdash)}{\neg x \vee \neg y, \neg x, \neg y, \neg\neg x \vdash}\,(\bullet \vdash)}{\neg x \vee \neg y \vdash \neg\neg x, \neg\neg y, \neg\neg\neg x}\,(\vdash \neg)}{\neg x \vee \neg y, \vdash x \vee y \vee \neg x}\,(\text{L. } 15)
$$

And finally we takes these two clauses together and add the quantifiers, quantifying over $y' = y \vee x$ on the right hand side.

$$
\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\neg x \vee \neg y \vdash \neg x \vee \neg(y \vee \neg x) \qquad \neg x \vee \neg y, \vdash x \vee y \vee \neg x}{\neg x \vee \neg y \vdash (\neg x \vee \neg(y \vee \neg x)) \wedge (x \vee y \vee \neg x)}\,(\vdash \wedge)}{\neg x \vee \neg y \vdash \exists y' (\neg x \vee \neg y') \wedge (x \vee y')}\,(\vdash \exists)}{\exists y (\neg x \vee \neg y) \vdash \exists y' (\neg x \vee \neg y') \wedge (x \vee y')}\,(\exists \vdash)}{\forall x \exists y (\neg x \vee \neg y) \vdash \exists y' (\neg x \vee \neg y') \wedge (x \vee y')}\,(\forall \vdash)}{\forall x \exists y (\neg x \vee \neg y) \vdash \forall x \exists y' (\neg x \vee \neg y') \wedge (x \vee y')}\,(\vdash \forall)
$$

We now do the same for QRATU, but with the Herbrand function.

Lemma 24. *Let ϕ be a CNF and $l' = l \wedge \bigvee_{D \in \phi}^{\bar{l} \in D} \bar{O}_D^l$, where $O_D^l \subset D$, $\bar{l} \notin O_D^l$, then $\phi[l'/l] \vdash \phi$.*

Proof. We need to show three different implications on clauses in ϕ. 1. $K \vdash K$ for $l, \bar{l} \notin K$. 2. $K \vee l \wedge \bigvee_{D \in \phi}^{\bar{l} \in D} \bar{O}_D^l \vdash K \vee l$. 3. $K \vee \neg(l \wedge \bigvee_{D \in \phi}^{\bar{l} \in D} \bar{O}_D^l) \vdash K \vee \bar{l}$. These can be proven in the following ways:

1. Achieved by the axiom rule $(\vdash)$ in G.
2. We can take $K \vee l \vdash K \vee l$ and strengthen the left side with $\bigvee_{D \in \phi}^{\bar{l} \in D} \bar{O}_D^l$.
3. If O_K^l is empty then we prove $\perp \vdash K \vee \bar{l}$ using $(\vdash \bullet)$ on $\perp \vdash$, otherwise we take $O_K^l \vdash O_K^l$ and weaken the right hand side with $(\vdash \vee)$ to get $O_K^l \vdash K \vee \bar{l}$.

$$
\dfrac{\dfrac{\dfrac{K\vee\bar{l}\vdash K\vee\bar{l}}{K\vee\bar{l}\vdash K\vee\bar{l},O^l_K}\,(\vdash\bullet)\qquad \dfrac{\dfrac{O^l_K\vdash O^l_K}{O^l_K\vdash K\vee\bar{l},O^l_K}\,(\vdash\bullet)}{\bigwedge^{\bar{l}\in D}_{D\in\phi}O^l_D\vdash K\vee\bar{l},O^l_K}\,(\bullet\vdash)}{K\vee\bar{l}\vee\bigwedge^{\bar{l}\in D}_{D\in\phi}O^l_D\vdash K\vee\bar{l},O^l_K}\,(\vee\vdash)\qquad O^l_K\vdash K\vee\bar{l}}{\dfrac{K\vee\bar{l}\vee\bigwedge^{\bar{l}\in D}_{D\in\phi}O^l_D\vdash K\vee\bar{l}}{K\vee\neg(l\wedge\bigvee^{\bar{l}\in D}_{D\in\phi}\bar{O}^l_D)\vdash K\vee\bar{l}}\,(\text{L. }15)}\,(\text{cut})
$$

We can repeatedly use the $(\vdash\wedge)$ rule to get $\phi[l'/l]\vdash\phi$. In the case that ϕ is the empty CNF, $\top\vdash\top$ suffices. □

In QRATA we showed in Lemma 22 we could add the new clause when written in terms of the Skolem function, here we show that we can make a QRATU reduction when written in terms of the new Herbrand function.

Lemma 25. *Let ϕ be a CNF and $l' = l\wedge\bigvee^{\bar{l}\in D}_{D\in\phi}\bar{O}^l_D$ where $O^l_D\subseteq D$, $\bar{l}\notin O^l_D$, and for every $D\in\phi$ with $\bar{l}\in D$, $\phi\wedge\neg C\wedge\bar{O}^l_D$ is a contradiction via unit propagation. Then $\phi, C\vee l'\vdash C$ has a short proof in G.*

Proof. For any $D\in\phi$ with $\bar{l}\in D$, $\phi,\bar{C},\bar{O}^l_D$ is a contradiction via unit propagation and so we can use Lemma 20 to get a short proof of sequent $\phi,\bar{C},\bar{O}^l_D\vdash$ and thus with Lemma 15, $(\vdash\neg)$ and double negation rule $\phi,\bar{O}^l_D\vdash C$. If there is at least one $D\in\phi$ with $\bar{l}\in D$, we can use $(\vee\vdash)$ repeatedly to get $\phi,\bigvee^{\bar{l}\in D}_{D\in\phi}\bar{O}^l_D\vdash C$. We then continue in G

$$
\dfrac{\dfrac{\phi,\bigvee^{\bar{l}\in D}_{D\in\phi}\bar{O}^l_D\vdash C}{\phi,l\wedge\bigvee^{\bar{l}\in D}_{D\in\phi}\bar{O}^l_D\vdash C}\,(\bullet\wedge\vdash)\qquad\dfrac{C\vdash C}{\phi,C\vdash C}\,(\bullet\vdash)}{\phi,C\vee l\wedge\bigvee^{\bar{l}\in D}_{D\in\phi}\bar{O}^l_D\vdash C}\,(\vee\vdash)
$$

If there are no clauses $D\in\phi$ with $\bar{l}\in D$ then $l' = l\wedge\bot$

$$
\dfrac{\dfrac{C\vdash C\qquad\dfrac{\dfrac{\bot\vdash}{\bot\vdash C}\,(\vdash\bullet)}{l\wedge\bot\vdash C}\,(\wedge\vdash)}{C\vee l\wedge\bot\vdash C}\,(\vee\vdash)}{\phi,C\vee l\wedge\bot\vdash C}\,(\bullet\vdash)
$$

□

Theorem 2 *Given a CNF $\psi=\phi\wedge(C\vee l)$ closed under prefix Π. Suppose that the QRAT rule QRATU can reduce $C\vee l$ to C. Then the sequent $\Pi\phi\wedge(C\vee l)\vdash\Pi\phi\wedge C$ has a polynomial size G proof.*

Proof. Let $\Pi = \Pi_1 \forall x \Pi_2$ with $x = \mathrm{var}(l)$. Let $l' = l \wedge \bigvee_{D \in \phi}^{\bar{l} \in D} \bar{O}_D^l$ using the definition of outer clauses. $\phi[l'/l] \vdash \phi$ and $\phi, C \vee l' \vdash C$ by Lemmas 25 and 24.

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\phi[l'/l] \vdash \phi \qquad \phi, C \vee l' \vdash C}{\phi[l'/l], C \vee l' \vdash C}\,\text{(cut)} \qquad \dfrac{\phi[l'/l] \vdash \phi}{\phi[l'/l], C \vee l' \vdash \phi}\,(\bullet \vdash)}{\phi[l'/l], C \vee l' \vdash \phi \wedge C}\,(\vdash \wedge)}{\phi[l'/l] \wedge (C \vee l') \vdash \phi \wedge C}\,\text{(Lemma 15)}}{\Pi_2 \phi[l'/l] \wedge (C \vee l') \vdash \Pi_2 \phi \wedge C}\,\text{(Lemma 16)}}{\forall x \Pi_2 \phi \wedge (C \vee l) \vdash \Pi_2 \phi \wedge C}\,(\forall \vdash)}{\forall x \Pi_2 \phi \wedge (C \vee l) \vdash \forall x \Pi_2 \phi \wedge C}\,(\vdash \forall)}{\Pi_1 \forall x \Pi_2 \phi \wedge (C \vee l) \vdash \Pi_1 \forall x \Pi_2 \phi \wedge C}\,\text{(Lemma 16)}
$$

□

The problem of strategies for EUR. In [4] it was shown that strategy extraction for EUR is not possible for circuits (under complexity assumptions), so using propositional witnesses as in Theorem 1 and 2 will not work. But we have not yet used a key property of G- witnesses with quantifiers. We will give a QBF witness for EUR in Section 5.2, but in order to have any hope of using it we must do some G formalisation of the dependency condition that allows EUR to work.

5 Resolution path independence

5.1 Resolution paths

We recap the reflexive resolution dependency scheme used in QRAT. This is the most difficult part of QRAT for G to simulate, therefore we give it the most attention. This is also the rule that allows QRAT to be stronger than Herbrand strategy extraction [4].

With resolution paths, the idea is to ask the question: *can these two clauses both appear in the same connected proof?* The reason we talk about paths is we consider clauses as vertices on a graph where vertices are connected by an edge if they share a variable and the literals are in opposite polarities, in other words an edge represents that a resolution can happen between the clauses.

The resolution path between two clauses is a path in this graph. From a clause C we can define the set of vertices reachable via resolution paths as $\mathfrak{C}$. However only particular literals can be used to move in the path. We should not reuse the same variable twice in succession. E.g. If we start with clause $x \vee y$ we can add $\bar{y} \vee z$ to $\mathfrak{C}$, but we should not use $\bar{y}$ as the next pivot, as the introduction of this clause removes it via resolution. In Definition 26 we treat this formally by keeping a set of usable literals $\mathfrak{L}$.

We also take into consideration the situation in QBF, in dependency schemes we only consider resolution paths on existential variables and only at certain levels. Instead of talking about existential variables and quantification orders we give a set of variables $\mathcal{S}$ for which we only consider resolution paths on, and build the theory from that.

Definition 26. *Consider a CNF ϕ and subset χ of clauses in ϕ and a subset $\mathcal{S}$ of variables. $\mathfrak{L}(\phi, \chi, \mathcal{S})$ lists the* $\mathcal{S}$-literals on the resolutions paths from χ *and* $\mathfrak{C}(\phi, \chi, \mathcal{S})$ *lists* the clauses on the the resolution paths from χ. *These are found using an iterative procedure until reaching a fix-point.*
Initialisation. *We start with the clauses in χ and the $\mathcal{S}$ literals in those clauses. $\mathfrak{L}(\phi, \chi, \mathcal{S}) \leftarrow \{l \mid$ there is $C \in \chi$ s.t. $l \in C, \mathrm{var}(l) \in \mathcal{S}\}$ and $\mathfrak{C}(\phi, \chi, \mathcal{S}) \leftarrow \chi$.*
Adding a clause. *If there if some D such that $\bar{p} \in D$ and $p \in \mathfrak{L}(\phi, \chi, \mathcal{S})$, then we can update $\mathfrak{L}(\phi, \chi, \mathcal{S})$ and $\mathfrak{C}(\phi, \chi, \mathcal{S})$. $\mathfrak{L}(\phi, \chi, \mathcal{S}) \leftarrow \mathfrak{L}(\phi, \chi, \mathcal{S}) \cup \{q \in D \mid q \neq \bar{p}, \mathrm{var}(q) \in \mathcal{S}\}$ and $\mathfrak{C}(\phi, \chi, \mathcal{S}) \leftarrow \mathfrak{C}(\phi, \chi, \mathcal{S}) \cup \{D\}$ We continue this until we reach fix-point, in other words for all $p \in \mathfrak{L}(\phi, \chi, \mathcal{S})$ if $D \in \phi$ and $\bar{p} \in D$, then $\{q \in D \mid q \neq \bar{p}, \mathrm{var}(q) \in \mathcal{S}\} \subset \mathfrak{L}(\phi, \chi, \mathcal{S})$ and $D \in \mathfrak{C}(\phi, \chi, \mathcal{S})$. Fix-point is reached in polynomial time.*

In QBF we use the resolution path to talk about connected Q-Resolution [15] proofs, and since Q-Resolution only resolves on existential pivots we need only to consider paths through existential variables. The lack of resolution path is used to show independence of clauses with opposing universal literals. So if all clauses with u in it cannot be connected via a resolution path to clauses with $\bar{u}$, then the universal player is free to choose whatever value of u, as whether there is a refutation is independent of the choice of clauses. We also only need to consider resolution paths using existential variables *to the right* of u in the prefix, as the question is whether there will be a refutation once the universal player has made their move.

The theories of resolution paths are used in QRAT, specifically in the EUR rule which allows a clause $C \vee u$ to be strengthened to C when u is a universal variable and there is no $D \in \mathfrak{C}(\phi \wedge C, C, \mathcal{S})$ with $\neg u$ in it, $\mathcal{S}$ being the set of inner existential variables with respect to u. The way to show a simulation of EUR is to formalise the property of resolution path independence into a sequent.

Theorem 3 *For any CNF ϕ with a subset χ_1 and let $\mathfrak{C} = \mathfrak{C}(\phi, \chi_1, \mathcal{S})$, where $\mathcal{S}$ is the set of existential variables of a prefix Π. G can prove the sequent $\Pi \bigwedge_{D \in \mathfrak{C}} D, \Pi(\phi \backslash \chi_1) \vdash \Pi\phi$ in a polynomial size proof.*

Proof. Define the following:

– ϕ_1 contains all clauses in all resolution paths of χ_1. ($\phi_1 = \bigwedge_{D \in \mathfrak{C}(\phi, \chi_1, \mathcal{S})} D$.)

- χ_2 contains the remaining clauses not reachable via resolution paths from χ_1. ($\chi_2 = \phi \setminus \phi_1$.)
- ϕ_2 closes χ_2 under resolution paths. ($\phi_2 = \bigwedge_{D\in\mathfrak{C}(\phi,\chi_2,\mathcal{S})} D$.)
- L_1 is all outgoing literals on res. paths from χ_1. ($L_1 = \mathfrak{L}(\phi, \chi_1, \mathcal{S})$.)
- L_2 is all outgoing literals on res. paths from χ_2. ($L_2 = \mathfrak{L}(\phi, \chi_2, \mathcal{S})$.)

Overlapping Clauses. Note that the existence of a resolution path between clauses D_1 and D_2 is symmetric. By definition, clauses of χ_2 are not in ϕ_1, but also clauses of χ_1 are not in ϕ_2. However resolution paths are not necessarily transitive, C could have a path to D and D could have a path to E, but if the variable used to enter D from C is the same literal to exit D to get to E that paths cannot be conjoined. This means ϕ_1 and ϕ_2 (which we can also think of as sets of clauses) are not necessarily disjoint.

First we observe that if there is some $D \in \phi_1 \cap \phi_2$ then there is a unique "entry" literal $z \in D$, $\mathrm{var}(z) \in \mathcal{S}$ such that $\bar{z} \in \mathfrak{L}(\phi, \chi_1, \mathcal{S})$ and $\bar{z} \in \mathfrak{L}(\phi, \chi_2, \mathcal{S})$, in other words $\bar{z}$ is an outgoing literal in both sets of paths.

We can prove this because there must be at least one $\mathcal{S}$-literal $\bar{z} \in \mathfrak{L}(\phi, \chi_1, \mathcal{S})$ that puts $D \in \mathfrak{C}(\phi, \chi_1, \mathcal{S})$ via $z \in D$ and there must be at least one $\mathcal{S}$-literal $\bar{z}' \in \mathfrak{L}(\phi, \chi_2, \mathcal{S})$ that puts $D \in \mathfrak{C}(\phi, \chi_2, \mathcal{S})$ via $z' \in D$. If $z \neq z'$, then z' we can make a path between χ_i and χ_j by reaching D from a path χ_1 using literal z to enter D and reverse the path from χ_2 to D, now using z' to exit D.

Finding Existential Witnesses. We want to show a sequent with two QBFs on the left hand side that use the same quantified variables, but in order to do this we have to treat the variables as different before they are quantified. For each $x \in L$ we use x^1 and x^2. For the right hand side we need terms that act as existential witnesses, we can assign each $\mathcal{S}$-literal a propositional term l' in the literals l^1, l^2 but the expression depends on l and $\bar{l}$'s inclusion in the sets L_1 and L_2

If an $\mathcal{S}$-literal l is in L_1 its negation cannot be in L_2 and vice versa, otherwise there would be a resolution path between χ_1 and χ_2.

- If either l or $\bar{l}$ are in L_1 and neither l nor $\bar{l}$ are in L_2 then let $l' = l^1$
- If either l or $\bar{l}$ are in L_2 and neither l nor $\bar{l}$ are in L_1 then let $l' = l^2$
- If l is in $L_1 \cap L_2$ then $\bar{l} \notin L_1 \cup L_2$, define $l' = l^1 \vee l^2$
- If $\bar{l}$ is in $L_1 \cap L_2$ then $l \notin L_1 \cup L_2$, define $l' = \neg(\bar{l}^1 \vee \bar{l}^2)$

l' preserves negation. We use each term l' on the right hand side to replace for l, these we will use as witnesses for $(\vdash \exists)$, but to do this we will first need $\phi_1^1, \phi_2^2 \vdash \phi'$, where f^1 is formula f with all $\mathcal{S}$-literals l replaced by l^1, f^2 is f with all $\mathcal{S}$-literals l replaced by l^2 and f' is f with all $\mathcal{S}$-literals l replaced with term l'.

Proving $\phi_1^1, \phi_2^2 \vdash \phi'$ requires our observation on entry literals. Without loss of generality if a clause $D \in \phi$ is only in ϕ_1 and not ϕ_2, then $D' = D^1$ and it is straightforward to prove $\phi_1^1, \phi_2^2 \vdash D^1$ since $D^1 \in \phi_1^1$. However if $D \in \phi_1 \cap \phi_2$ then $D = K \vee z$ where z is the unique entry literal and we let K be the sub-clause of remaining literals. $\bar{z}$ must be in $L_1 \cap L_2$ so $z' = \neg(\bar{z}^1 \vee \bar{z}^2)$. Every $\mathcal{S}$-literal k in K is also in $L_1 \cap L_2$ so $k' = k^1 \vee k^2$. $D' = K' \vee \neg(\bar{z}^1 \vee \bar{z}^2)$.

$$\dfrac{\dfrac{\dfrac{D^1 \vdash K^1 \vee z^1}{\phi_1^1 \vdash K^1 \vee z^1}\,(\wedge \vdash)}{\phi_1^1, \phi_2^2 \vdash K^1 \vee z^1}\,(\bullet \vdash) \qquad \dfrac{\dfrac{D^2 \vdash K^2 \vee z^2}{\phi_2^2 \vdash K^2 \vee z^2}\,(\wedge \vdash)}{\phi_1^1, \phi_2^2 \vdash K^2 \vee z^2}\,(\bullet \vdash)}{\dfrac{\phi_1^1, \phi_2^2 \vdash K^1 \vee z^1 \wedge K^2 \vee z^2}{\dfrac{\phi_1^1, \phi_2^2 \vdash K' \vee z^1 \wedge K' \vee z^2}{\phi_1^1, \phi_2^2 \vdash K' \vee \neg(\neg z^1 \vee \neg z^2)}\,(L.\ 15)}\,(\vdash \vee)}\,(\vdash \wedge)$$

We take all these individual sequents together into a conjunction and get $\phi_1^1, \phi_2^2 \vdash \phi'$. We can strengthen ϕ_2^2 to $\phi^2 \setminus \chi_1^2$ on the left hand side since clauses from χ_1 cannot appear in ϕ_2. We end up with $\phi_1^1, \phi^2 \setminus \chi_1^2 \vdash \phi'$.

Adding the Quantifiers. We now add the quantifiers from innermost to outermost. When we need to quantify a universal variable y we require universal quantifiers $\forall y$ for both of the formulas on the left hand side. ($\forall \vdash$) require a witness and each time we can just use y itself, then we simply use variable y for ($\vdash \forall$) on the right hand side. For existential variables x we first quantify the right hand side using the term x'. Now for the left hand side variable x^i only appears in one of the two formulas, so we can use that to quantify $\exists x$ for each. Adding in all the quantifiers grants us $\Pi\phi_1, \Pi(\phi \backslash \chi_1) \vdash \Pi\phi$ as required.

5.2 Extended Universal Reduction

Consider using universal reduction to reduce $\Pi\phi \wedge (C \vee l)$ into $\Pi\phi \wedge C$. The condition in standard universal reduction is that all literals $y \in C$ are quantified to the left of l in prefix Π, i.e $y <_\Pi l$. For the soundness, we can observe how Herbrand functions are preserved moving backwards in the proof.

We have to show that if there is a Herbrand function σ_l for the succedent then there is a Herbrand function σ'_l for the antecedent. Because all variables in C are left of l these variables we name this $\boldsymbol{x}_l$ and this is the domain of the Herbrand function and can be used to construct it. We let $\sigma'_l(\boldsymbol{x}_l) = 0$ whenever C is falsified, and $\sigma'_l(\boldsymbol{x}_l) = \sigma_l(\boldsymbol{x}_l)$ otherwise.

We note that σ'_l for standard UR the universal player never downgrades her outcome, when $\neg C$ she always guarantees her victory, either winning where she would have won otherwise or winning when she would have lost otherwise, when C is true she plays according σ_l and, since all the outcomes are now the same, she also does not downgrade her game.

In EUR we cannot use the condition $\neg C$ as it may contain variables to the right of l, but there is a similar situation where the universal player can safely set l to 0. If she knows she can play her remaining moves such that the existential player cannot satisfy every clause without $\bar{l}$ in them, then it does not matter if she satisfies all the clauses with $\bar{l}$ in them by setting l to 0. She only has to guarantee her victory on a subset of clauses that do not contain $\bar{l}$. According to our EUR condition, that subset can precisely be $\mathfrak{C}(\phi \wedge C, C, \mathcal{S})$, the set of clauses in the resolutions paths from C, where $\mathcal{S}$ is all existential variables right of l.

In order to play this she requires foresight of the outcome for the remaining moves. For this reason it cannot be used to build a circuit strategy. However, Herbrand functions can still be made by using quantifiers on the variables right of l. Let $\Pi_2 \subset \Pi$ be the part of the prefix strictly right of l. If we have Herbrand function σ_l for $\Pi\phi \wedge C$ we can find another Herbrand function:

$$\sigma'_l(\boldsymbol{x}_l) = \begin{cases} 0 & \text{if } \neg\Pi_2(\bigwedge_{D \in \mathfrak{C}(\phi \wedge C, C, \mathcal{S})} D[\bot/l]) \\ \sigma_l(\boldsymbol{x}_l) & \text{otherwise.} \end{cases}$$

$\sigma'_l(\boldsymbol{x}_l)$ is a valid Herbrand function for $\Pi\phi \wedge C$, but how is it also valid for $\Pi\phi \wedge C \vee l$? This is due to the essential independence condition that is required for EUR. If under some assignment to the free variables $\Pi_2(\bigwedge_{D \in \mathfrak{C}(\psi, C, \mathcal{S})} D[\bot/l])$ is true but $\Pi_2(\phi \wedge C)$ is false, then Theorem 3 tells us $\Pi_2\phi$ must be false, so C becomes irrelevant to the refutation.

We have to show this all formally in G. We will prove as much as we can before using Theorem 3.

Lemma 27. *Let $\forall u \Pi_2 \psi$ be a QBF with Π_2 a prefix, u a variable and $\psi = \phi \wedge C \vee l$, where C is a clause and ϕ a CNF and literal l has variable u. Let $\mathcal{S}$ denote the set of all existential literals in Π_2. Let $\mathfrak{C}$ be a shorthand for $\mathfrak{C}(\psi, C \vee l, \mathcal{S})$ and assume that there is no $D \in \mathfrak{C}$ with $\bar{l}$ in D. Let Δ be a shorthand for $\Pi_2(\bigwedge_{D \in \mathfrak{C}} D[\bot/l])$. Let l' be the formula $l \wedge \Delta$. Then the following are provable in polynomial size G proofs. (A) $\forall u \Pi_2 \psi \vdash \Delta$. (B) $\forall u \Pi_2 \psi \vdash \Pi_2 \phi \wedge (C \vee l')$. (C) $\forall u \Pi_2 \psi \vdash \Pi_2(\phi \wedge C), \Pi_2(\bigwedge_{D \in \mathfrak{C}} D)$.*

Proof (Proof of A).

We start with $\bigwedge_{D\in\mathfrak{C}} D[\bot/l] \vdash \bigwedge_{D\in\mathfrak{C}} D[\bot/l]$.

$$\dfrac{\bigwedge_{D\in\mathfrak{C}} D[\bot/l] \vdash \bigwedge_{D\in\mathfrak{C}} D[\bot/l]}{\phi[\bot/l] \wedge (C \vee \bot) \vdash \bigwedge_{D\in\mathfrak{C}} D[\bot/l]}\ (\wedge\vdash)$$

By Lemma 16 we can add the Π_2 on both sides. And finally by using $(\forall\vdash)$ rule over a constant symbol $\bot$ (or $\top$ if $l = \bar{u}$), we get sequent $\forall u \Pi_2(\phi \wedge C \vee l) \vdash \Pi_2(\bigwedge_{D\in\mathfrak{C}} D[\bot/l])$.

Proof (Proof of B). Let $l' = l \wedge \Pi_2(\bigwedge_{D\in\mathfrak{C}} D[\bot/l])$ we have to show what the substitution $[l'/l]$ can prove for every clause $D \in \phi$.

1. For any $D \in \phi$ with $l, \bar{l} \notin D$ we have $D \vdash D$.
2. For any $D \in \phi$ with $l \in D$, we have $D[l'/l] \vdash D$.
3. For any $D \in \phi$ with $\bar{l} \in D$, we have $D[l'/l] \vdash D, \neg\Delta$.

To show these we do the following:

1. $D \vdash D$ is an axiom of G.
2. Let $D = K \vee l$, note that l' is just a strengthening of l so $K \vee l \vdash K \vee l$ comes out of strengthening the left hand side.

$$\dfrac{\dfrac{\dfrac{l \vdash l}{l \vdash K \vee l}\ (\vdash\vee)}{l' \vdash K \vee l}\ (\wedge\vdash) \qquad \dfrac{K \vdash K}{K \vdash K \vee l}\ (\vdash\vee)}{K \vee l' \vdash K \vee l}\ (\vee\vdash)$$

3. Let $D = K \vee \bar{l}$, this makes the sequent we wish to prove $K \vee \neg(l \wedge \Delta) \vdash K \vee \bar{l}, \neg\Delta$ which is an application of Lemma 15.

Now we take a conjunction of all cases of 1,2 and 3 along with $C \vee l'$ and get $\phi[l'/l] \wedge C \vee l' \vdash \phi \wedge C \vee l', \neg\Delta$ We can use the negation rule to bring Δ to the LHS. Which allows us to cut with Lemma 27A to get $\forall u \Pi_2 \psi, \phi[l'/l] \wedge (C \vee l') \vdash \phi \wedge (C \vee l')$. We can quantify both sides by Π_2 using the technique from Lemma 16 to get $\forall u \Pi_2(\phi \wedge C \vee l), \Pi_2(\phi[l'/l] \wedge C \vee l') \vdash \Pi_2(\phi \wedge C \vee l')$. Using u' as the term (where $u' = l'$ if $u = l$ and $\bar{u}' = l'$ if $\bar{u} = l$) $\Pi_2(\phi[l'/l] \wedge C \vee l')$ can be quantified universally to get $\forall u \Pi_2(\phi \wedge C \vee l) \vdash \Pi_2(\phi \wedge C \vee l')$.

Proof (Proof of C). Using $l' = l \wedge \Pi_2(\bigwedge_{D\in\mathfrak{C}} D[\bot/l])$, we make the following derivation.

$$
\dfrac{\dfrac{\dfrac{\dfrac{\Delta \vdash \Delta}{l' \vdash \Delta}\,(\wedge\vdash)}{l' \vdash \Delta, C}\,(\vdash \bullet) \qquad \dfrac{C \vdash C}{C \vdash \Delta, C}\,(\vdash \bullet)}{C \vee l' \vdash \Delta, C}\,(\vee\vdash) \qquad \dfrac{\dfrac{C \vdash C}{C, \phi \vdash C}\,(\bullet\vdash) \qquad \dfrac{\phi \vdash \phi}{C, \phi \vdash \phi}\,(\bullet\vdash)}{C, \phi \vdash \phi \wedge C}\,(\vdash\wedge)}{\dfrac{\phi, C \vee l' \vdash \phi \wedge C, \Delta}{\phi \wedge C \vee l' \vdash \phi \wedge C, \Delta}\,(\text{Lemma 15})}\,(\text{cut})
$$

Now we quantify the Π_2 variables using the same technique as Lemma 16 to get $\Pi_2(\phi \wedge C \vee l') \vdash \Pi_2(\phi \wedge C), \Delta$. In order to simplify the right hand side even further, for every $D \in \mathfrak{C}$ we take axioms $D[\bot/l] \vdash D[\bot/l]$ and since $\bar{l}$ does not appear in D, we can always obtain $D[\bot/l] \vdash D$ by weakening the right hand side. We can strengthen the left hand side to $\bigwedge_{D\in\mathfrak{C}} D[\bot/l]$ and get the conjunction $\bigwedge_{D\in\mathfrak{C}} D[\bot/l] \vdash \bigwedge_{D\in\mathfrak{C}} D$. By Lemma 16 we get $\Delta \vdash \Pi_2(\bigwedge_{D\in\mathfrak{C}} D)$, this we can use to cut with our sequent.

$$
\dfrac{\Pi_2(\phi \wedge C \vee l') \vdash \Pi_2(\phi \wedge C), \Delta \qquad \Delta \vdash \Pi_2(\bigwedge_{D\in\mathfrak{C}} D)}{\Pi_2(\phi \wedge C \vee l') \vdash \Pi_2(\phi \wedge C), \Pi_2(\bigwedge_{D\in\mathfrak{C}} D)}
$$

We can simply cut with Lemma 27B to get $\forall u \Pi_2 \psi \vdash \Pi_2(\phi \wedge C), \Pi_2(\bigwedge_{D\in\mathfrak{C}} D)$.

Were we to show $\forall u \Pi_2 \psi \vdash \Pi_2(\phi \wedge C)$, proving EUR's sequent in G would be a matter of adding the remaining quantifiers with Lemma 16. $\forall u \Pi_2 \psi \vdash \Pi_2(\phi \wedge C), \Pi_2(\bigwedge_{D \in \mathfrak{C}} D)$ is almost what we need, the only disagreement is when $\forall u \Pi_2 \psi$ is true, $\Pi_2(\phi \wedge C)$ is false and $\Pi_2(\bigwedge_{D \in \mathfrak{C}} D)$ is true. If that occurs, then we can apply Theorem 3 and use it to tell us $\Pi_2 \phi$ must be false. However, $\forall u \Pi_2 \psi$ cannot possibly be true if $\Pi_2 \phi$ is false meaning this situation does not occur and we effectively have $\forall u \Pi_2 \psi \vdash \Pi_2(\phi \wedge C)$. We can formalise this in G.

Theorem 4 *Let ϕ be a CNF with Π a prefix. Suppose that the QRAT rule EUR can reduce clause $C \vee l$ to C, where C is a clause and l is a literal. Then the sequent $\Pi \phi \wedge (C \vee l) \vdash \Pi \phi \wedge C$ has a polynomial size G proof.*

Proof. Let $\Pi = \Pi_1 \forall u \Pi_2$, where $u = var(l)$. Let $\mathfrak{C}$ be shorthand for $\mathfrak{C}(\phi \wedge C, C, \mathcal{S})$ with $\mathcal{S}$ denoting all $\exists$ literals in Π_2. Lemma 27 gets us most of the way through this proof, but we need to use Theorem 3 with PCNF $\Pi_2(\phi \wedge C)$ with $\chi_1 = C$ to obtain $\Pi_2 \phi, \Pi_2 \bigwedge_{D \in \mathfrak{C}} D \vdash \Pi_2(\phi \wedge C)$.

We use ($\forall \vdash$) to gain $\forall u \Pi_2(\phi \wedge C \vee l) \vdash \Pi_2(\phi \wedge C \vee l)$, and Corollary 17 to gain $\Pi_2(\phi \wedge C \vee l) \vdash \Pi_2 \phi$. We cut these two with our sequent from Theorem 3 to get $\forall u \Pi_2(\phi \wedge C \vee l), \Pi_2(\bigwedge_{D \in \mathfrak{C}} D) \vdash \Pi_2(\phi \wedge C)$. Now we use Lemma 27 to gain $\forall u \Pi_2(\phi \wedge C \vee l) \vdash \Pi_2(\phi \wedge C), \Pi_2(\bigwedge_{D \in \mathfrak{C}} D)$. Cutting these two sequents together removes $\Pi_2(\bigwedge_{D \in \mathfrak{C}} D)$ and gets us $\forall u \Pi_2(\phi \wedge C \vee l), \Pi_2(\phi \wedge C \vee l) \vdash \Pi_2(\phi \wedge C)$. Quantifying the free l on the left hand side contracts the two QBFs into one. We can add the remaining quantifiers with Lemma 16.

6 Conclusion

Theorem 5 *G p-simulates QRAT.*

Proof. True QBF. If Ψ is a true QBF and we have a QRAT proof π_{QRAT}. We show that we can obtain a G proof π_{G} of $\vdash \Psi$ in polynomial time from π_{QRAT}.

π_{QRAT} is a sequence of lines $L_0 \ldots L_m$ using steps ATE, QRATE and clause addition.

Induction Hypothesis *(on increasing i)*: We can obtain a polynomial size G proof of $L_i \vdash \Psi$

Base Case: ($i = 0$) The first QBF L_0 in a QRAT proof is the initial QBF which here is Ψ. $\Psi \vdash \Psi$ is an axiom in G.

Inductive Step: We derive the sequent $L_{i+1} \vdash L_i$ depending on the QRAT rule.

- Clause addition: If we add clause C to CNF ϕ (under prefix Π) we use Corollary 17 to gain sequent: $\Pi \phi \wedge C \vdash \Pi \phi$
- ATE: If we remove clause C from CNF ϕ (under prefix Π) we use Theorem 1 to gain sequent: $\Pi \phi \backslash \{C\} \vdash \Pi \phi$

- QRATE: If we remove clause C from CNF ϕ (under prefix Π)we use Theorem 1 to gain sequent: $\Pi\phi \backslash \{C\} \vdash \Pi\phi$

Since we have $L_i \vdash \Psi$ by the induction hypothesis we use the cut rule to get $L_{i+1} \vdash \Psi$

Final Case: For the final line L_m we have the empty CNF. By the induction hypothesis we have $\Pi\phi \vdash \Psi$, where $\Pi\phi$ is L_{m-1}. The only difference for this final step is that we have to deal with the empty CNF, but this is not difficult to deal with. We represent L_m as $\Pi\top$. Using Theorem 1 we can get $\Pi\top \vdash \Pi\top \wedge \phi$.

To complete the proof we do the following G steps:

$$\dfrac{\dfrac{\dfrac{\dfrac{\vdash \top}{\vdash \Pi\top}\,(\text{L. } 16) \qquad \Pi\top \vdash \Pi\top \wedge \phi}{\vdash \Pi\top \wedge \phi}\,(\text{cut}) \qquad \dfrac{\dfrac{\phi \vdash \phi}{\top \wedge \phi \vdash \phi}\,(\bullet\wedge\vdash)}{\Pi\top \wedge \phi \vdash \Pi\phi}\,(\text{L. } 16)}{\vdash \Pi\phi}\,(\text{cut}) \qquad \Pi\phi \vdash \Psi}{\vdash \Psi}\,(\text{cut})$$

False QBF. If Ψ is a false QBF and we have a QRAT proof π_{QRAT}. We show that we can obtain a G proof π_{G} of $\Psi \vdash$ in polynomial time from π_{QRAT}.

π_{QRAT} is a sequence of lines $L_0 \dots L_m$ using steps ATE, QRATE and clause addition.

Induction Hypothesis: We can obtain a polynomial size G proof of $\Psi \vdash L_i$.

Base Case: $(i = 0)$ The first QBF L_0 in a QRAT proof is the initial QBF which here is Ψ. $\Psi \vdash \Psi$ is an axiom in G.

Inductive Step: We derive the sequent $L_i \vdash L_{i+1}$ depending on the QRAT rule.

- Clause deletion: If we delete clause C from CNF ϕ (under prefix Π) we use Corollary 17 to gain sequent: $\Pi\phi \vdash \Pi\phi \backslash \{C\}$
- ATA: If we add clause C to CNF ϕ (under prefix Π) we use Theorem 1 to gain sequent: $\Pi\phi \vdash \Pi\phi \wedge C$.
- QRATA: If we add clause C to CNF ϕ (under prefix Π) we use Theorem 1 to gain sequent: $\Pi\phi \vdash \Pi\phi \wedge C$.
- QRATU: If we remove literal l from clause C in CNF ϕ (under prefix Π) we use Theorem 4 to gain sequent: $\Pi\phi \vdash \Pi\phi \backslash \{C \vee l\} \wedge C$.

We use the cut rule to cut $\Pi\phi$ to get $\Psi \vdash L_{i+1}$

Final Case: For the final line L_m we have the empty clause. By the induction hypothesis we also have $\Psi \vdash \Pi\phi$, The final line either adds an empty clause via ATA, or reduces a singleton universal literal l using EUR or QRATA.

If the empty clause is added via ATA we can use Lemma 20 to gain $\Pi\phi \vdash$ and then use cut to get $\Psi \vdash$. If we use EUR or QRATU there is universal variable u with literal l such that $\Pi = \Pi_1 \forall u \Pi_2$

$$
\dfrac{\Psi \vdash \Pi\phi \qquad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\bot \vdash \bot}{\Pi_2\phi\backslash\{l\} \wedge \bot \vdash \bot}}{\Pi_2\bot \vdash \bot}}{\forall u \Pi_2\phi \vdash \Pi_2\bot}}{\Pi\phi \vdash \bot} \qquad \bot \vdash}{\Pi\phi \vdash}}{\Psi \vdash}
$$

We have finally proven that G p-simulates QRAT, but this is only the beginning of the search for a new checking format for QBF. In our opinion, G is not suitable in a practical setting. The next step in this search should be to find out how or if extension variables can be used to represent full QBFs, in order to simulate G. The hard part of this will be simulating the non-prenex QBFs. Non prenex QBF solvers have recently seen some interest [10, 18], so getting a practical proof system that has a way of handling them would be very beneficial.

While a genuine QBF-cut extended variable systems may exists and could be used in practice, improvements in the direction of Propagation Redundancy [9] would likely exist and we would want to develop QBF systems further along these lines.

References

1. Beyersdorff, O., Bonacina, I., Chew, L., Pich, J.: Frege systems for quantified boolean logic. J. ACM **67**(2) (Apr 2020). https://doi.org/10.1145/3381881, `https://doi.org/10.1145/3381881`
2. Chew, L.: Hardness and optimality in QBF proof systems modulo NP. Electron. Colloquium Comput. Complex. **25**, 178 (2018), `https://eccc.weizmann.ac.il/report/2018/178`
3. Chew, L., Clymo, J.: The equivalences of refutational QRAT. In: Janota, M., Lynce, I. (eds.) Theory and Applications of Satisfiability Testing - SAT 2019 - 22nd International Conference, SAT 2019, Lisbon, Portugal, July 9-12, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11628, pp. 100–116. Springer (2019). https://doi.org/10.1007/978-3-030-24258-9_7, `https://doi.org/10.1007/978-3-030-24258-9\_7`
4. Chew, L., Clymo, J.: How QBF expansion makes strategy extraction hard. In: Peltier, N., Sofronie-Stokkermans, V. (eds.) Automated Reasoning - 10th International Joint Conference, IJCAR 2020, Paris, France, July 1-4, 2020, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12166, pp. 66–82. Springer (2020). https://doi.org/10.1007/978-3-030-51074-9_5, `https://doi.org/10.1007/978-3-030-51074-9\_5`
5. Egly, U.: On stronger calculi for QBFs. In: Proc. 19th International Conference on Theory and Applications of Satisfiability Testing (2016)
6. Gentzen, G.: Untersuchungen über das logische Schließen. Mathematische Zeitschrift **39**, 68–131 (1935)
7. Heule, M., Seidl, M., Biere, A.: Efficient extraction of skolem functions from qrat proofs. 2014 Formal Methods in Computer-Aided Design, FMCAD 2014 pp. 107–114 (12 2014). https://doi.org/10.1109/FMCAD.2014.6987602

8. Heule, M., Seidl, M., Biere, A.: A unified proof system for QBF preprocessing. In: Automated Reasoning – 7th International Joint Conference, IJCAR. vol. 8562, pp. 91–106. Springer (2014). https://doi.org/10.1007/978-3-319-08587-6_7, http://dx.doi.org/10.1007/978-3-319-08587-6_7
9. Heule, M.J., Kiesl, B., Biere, A.: Strong extension-free proof systems. Journal of Automated Reasoning **64**(3), 533–554 (2020)
10. Janota, M.: QFUN: towards machine learning in QBF. CoRR **abs/1710.02198** (2017), http://arxiv.org/abs/1710.02198
11. Kiesl, B., Heule, M.J.H., Seidl, M.: A little blocked literal goes a long way. In: Gaspers, S., Walsh, T. (eds.) Theory and Applications of Satisfiability Testing – SAT 2017. pp. 281–297. Springer International Publishing, Cham (2017)
12. Kiesl, B., Rebola-Pardo, A., Heule, J.H.: Extended resolution simulates DRAT. In: Galmiche, D., Schulz, S., Sebastiani, R. (eds.) Automated Reasoning - 9th International Joint Conference, IJCAR 2018, Held as Part of the Federated Logic Conference, FloC 2018, Oxford, UK, July 14-17, 2018, Proceedings. Lecture Notes in Computer Science, vol. 10900, pp. 516–531. Springer (2018)
13. Kiesl, B., Seidl, M.: QRAT polynomially simulates $\forall$Exp+Res. In: Janota, M., Lynce, I. (eds.) Theory and Applications of Satisfiability Testing – SAT 2019. pp. 193–202. Springer International Publishing, Cham (2019)
14. Kleine Büning, H., Bubeck, U.: Theory of quantified Boolean formulas. In: Biere, A., Heule, M., van Maaren, H., Walsh, T. (eds.) Handbook of Satisfiability, Frontiers in Artificial Intelligence and Applications, vol. 185, pp. 735–760. IOS Press (2009)
15. Kleine Büning, H., Karpinski, M., Flögel, A.: Resolution for quantified Boolean formulas. Inf. Comput. **117**(1), 12–18 (1995)
16. Krajíček, J.: Proof complexity. In: European congress of mathematics (ECM). pp. 221–231. Stockholm, Sweden, Zurich: European Mathematical Society (2005)
17. Krajíček, J., Pudlák, P.: Quantified propositional calculi and fragments of bounded arithmetic. Zeitschrift für mathematische Logik und Grundlagen der Mathematik **36**, 29–46 (1990)
18. Tentrup, L.: CAQE and quabs: Abstraction based QBF solvers. J. Satisf. Boolean Model. Comput. **11**(1), 155–210 (2019). https://doi.org/10.3233/SAT190121, https://doi.org/10.3233/SAT190121

ECCC ISSN 1433-8092
https://eccc.weizmann.ac.il