Mathematical Problems of Computer Science 62, 9–16, 2024.

doi:10.51408/1963-0116

UDC 510.64

On Quantified Splitting Proof System for Propositional Calculi

Anahit A. Chubaryan

Yerevan State University, Yerevan, Armenia
e-mail: achubaryan@ysu.am

Abstract

In this paper, some new quantified propositional proof system is introduced and compared by proof complexities with other quantified and not quantified propositional proof systems. It is proved that the introduced system 1) is polynomially equivalent to its quantifier-free variant and 2) has exponential speed-up by sizes over some variants of the quantified resolution system. As the introduced system has a very simple proof construction strategy, it can be very useful not only in Logic, and therefore in Artificial Intelligence, but also in areas such as Computational Biology and Medical Diagnosis.

Keywords: Quantified propositional proof systems, Generalized Splitting system, Proof steps, Proof size, Exponential speed-up.

Article info: Received 29 September 2024; sent for review 15 October 2024; accepted 7 November 2024.

1. Introduction

It is well known that Mathematical Logic, in particular propositional calculi, is the base of Artificial Intelligence and therefore, has very *interesting applications* in fields such as Computational Biology and Medical Diagnosis.

Propositional proof complexity originates from the seminal paper by Cook and Reckhow. [1]. It provides a path for approaching the P vs. NP problem: proving super-polynomial lower bounds to all propositional proof systems is equivalent to showing that NP is different from coNP and therefore P is different from NP. It is well known that the exponential lower bounds for proof sizes of some sets of tautologies are obtained in many systems, but for some of the most natural calculi, in particular, for Frege systems, the question about polynomially bounded sizes is still open. While traditionally the complexity of proofs for propositional tautologies has been at the centre of research, the past two decades have witnessed a surge in proof complexity of quantified boolean formulas (QBFs), which give not only a new class of tautologies, but some quantifier-free tautologies can be proved simpler in any quantified systems. Some interesting survey of proof complexity for QBFs is given in [2], where the complexities for some QBF families are compared in different quantified propositional proof systems: variants of QBF resolution, QBF Frege systems, quantified versions of cutting planes, QBF sequent calculi and some others.

Based on the propositional system GS (Generalized Splitting), described in [3], a new quantified propositional proof system is introduced here. The place of the system GS in the hierarchy of propositional proof systems [1] remains unknown and moreover: by the comparison of the two main proof complexity characteristics (*step*s and *size*) for two classes of formulas in the system GS and Frege systems it is shown that for one class of considered formulas the bounds in the system GS are much better than those in the Frege systems, while for the second class the situation is quite the opposite [4].

From all the above mentioned it follows that the investigations of proof complexities in some quantified variants of the system GS can be important. Consequently, the possible practical applications of these systems in different non-mathematical areas may also be important.

2. Preliminaries

We will use the current concepts of a propositional formula, a proof system for propositional logic, proof complexity, and well-known notions of polynomial equivalence and exponential speed-up. The language of considered systems contains the propositional variables, logical connectives $\neg, \&, \vee, \supset, \leftrightarrow$ and parentheses (,). Following the usual terminology, we call the variables and negated variables *literals*. In [3], the following notions were introduced. Each of the following trivial identities for a propositional formula ψ we call a replacement rule.

$$
\begin{array}{llll}
0 \,\&\, \psi = 0, & \psi \,\&\, 0 = 0, & 1 \,\&\, \psi = \psi, & \psi \,\&\, 1 = \psi, \\
0 \vee \psi = \psi, & \psi \vee 0 = \psi, & 1 \vee \psi = 1, & \psi \vee 1 = 1, \\
0 \supset \psi = 1, & \psi \supset 0 = \bar{\psi}, & 1 \supset \psi = \psi, & \psi \supset 1 = 1, \\
\bar{0} = 1, & \bar{1} = 0, & \bar{\bar{\psi}} = \psi, & \\
0 \leftrightarrow \psi = \bar{\psi}, & \psi \leftrightarrow 0 = \bar{\psi}, & 1 \leftrightarrow \psi = \psi, & \psi \leftrightarrow 1 = \psi.
\end{array}
$$

The application of a replacement rule to some words consists in replacing some of its sub-words, having the form of the left-hand side of one of the above identities, by the corresponding right-hand side.

The proof system GS. Let φ be some formula and p be some of its variables. The results of the splitting method of formula φ by variable p (split variable) are the formulas $φ[p^{\boldsymbol{δ}}]$ for every **δ** from the set {0,1}, which are obtained from φ by assigning **δ** to each occurrence of p and successively using replacement rules. The generalization of the splitting method allows associating every formula φ with some tree with root, the nodes of which are labeled by formulas and edges, labeled by literals. The root is labeled by itself formula φ. If some node is labeled by formula v and α is its some variable, then both edges, which go out from this node, are labeled by one of literals $α^{\boldsymbol{δ}}$ for every **δ** from the set {0,1}, and each of the 2 "sons" of this node is labeled by the corresponding formula $v[α^{\boldsymbol{δ}}]$. Each leaf of the tree is labeled with some constant from the set {0,1}. The tree, which is constructed for formula φ by the described method, will be called a ***splitting tree*** (s.t.) of φ. It is obvious that by changing the order of split variables in the given formula φ, we can obtain different splitting trees of φ. We can note that the strategy of splitting tree construction is quite simple.

The ***GS*** proof system can be defined as follows: for every formula φ, some s.t. must be constructed and if all the leaves of the tree are labeled with the value 1, then the formula φ is a tautology and therefore we can consider the pointed constant 1 as an axiom, and for every formula v, which is the label of some s.t. node, and p is its split variable, then the following figure $v\,[p^0]$, $v\,[p^1]\ \vdash v$ can be considered as some inference rule, hence every above-described s.t. can be considered as some proof of φ in the system ***GS***. Note that if we consider the splitting method for formulas given in disjunctive normal form, then the ***GS*** system is the well-known Analytic Tableaux system.

By $|\varphi|$ we denote the size of a formula φ, defined as the number of all logical signs in it. It is obvious that the full size of a formula, which is understood to be the number of all symbols, is bounded by some linear function in $|\varphi|$.
The T-complexity (L-complexity) of s.t. is the *number (the sum of sizes) of different formulas*, with which its nodes are labeled. The **T-complexity (L-complexity) of GS-proof for tautology φ** is the value of minimal T-complexity (L-complexity) of its splitting trees.

3. Main Results

Quantified Splitting system (QS). A QBF is a propositional formula augmented with Boolean quantifiers ∀, ∃ that range over the Boolean values 0, 1. Every propositional formula is already a QBF. Let ϕ be a QBF. The semantics of the quantifiers are: $\forall x\phi(x) \equiv \phi[x^0]\ \&\ \phi[x^1]$ and $\exists x\phi(x) \equiv \phi[x^0] \vee \phi[x^1]$. In standardized QBF investigated in computer science , all quantifiers appear outermost in a (quantifier) prefix and are followed by a propositional formula, called a *matrix*. The variables following the quantifier ∀ are called *universal* variables, and the variables following the quantifier ∃ are called *existential variables*. The system **QS** works as follows: for any QBF formula φ, we use the system ***GS*** to the matrix of φ. S.t. for every GBF tautology φ must be the following: if for any step the splinted variable α is a universal variable of φ, then both subtrees, stuffed from the $α^0$ and $α^1$ labeled edges must have some branch, ended with value 1 labeled leaves; if for any step the splinted variable α is an existential variable of φ, then at least one of the subtrees, stuffed from the $α^0$ or $α^1$ labeled edges must have some branch, ended with value 1 labeled leaves.

QU-Resolution system. *Propositional resolution* [2] is a refutational system operating with clauses, i.e., it demonstrates the unsatisfiability of a given CNF. It has only a well-known *resolution rule*. *QBF resolution systems* work with fully quantified prenex QCNFs. As in propositional resolution, these QBF systems are refutational calculi, i.e., they refute false QBFs to be obtained by augmenting the propositional resolution system using the resolution rule by just one new rule, the *universal reduction rule*, $\frac{c}{\sigma(c)}$, where σ is partial substitution that allows a universal variable from a clause C to be replaced by either 0 or 1, provided that u appears right of all variables in C in the prefix Q. Intuitively, this means that a universal variable u can be deleted from a clause C if u is rightmost in C with respect to Q, i.e., no variable in C depends on u.

Theorem. 1) The systems GS and QS are polynomially equivalent by steps and sizes;

2) The system QS has an exponential speed-up by size over the QU-Resolution system.

Proof of 1) is obvious because every quantifier-free tautology $A(\mathbf{x_1}, \dots, \mathbf{x_n})$ can be presented as a QFB formula $\forall \mathbf{x_1} \dots \mathbf{x_n}\, A(\mathbf{x_1}, \dots, \mathbf{x_n})$, the matrix of which is $A(\mathbf{x_1}, \dots, \mathbf{x_n})$. Proof of 2) is based on the investigation of proof steps and sizes of equality families of QBFs

$$\mathbf{SC_n} = \exists \mathbf{x_1} \dots \mathbf{x_n} \forall \mathbf{u_1} \dots \mathbf{u_n} \exists \mathbf{t_1} \dots \mathbf{t_n} \left(\bigwedge_{1\le i\le n} (\mathbf{x_i} \leftrightarrow \mathbf{u_i}) \supset \bar{t}_i \right) \wedge \left(\bigvee_{1\le i\le n} \mathbf{t_i} \right).$$

For every s.t., we use its scheme, which is the same tree without node labels. It is not difficult to see that the s.t. of SC_1-matrix and the s.t. of SC_2 –matrix have the following schemes:

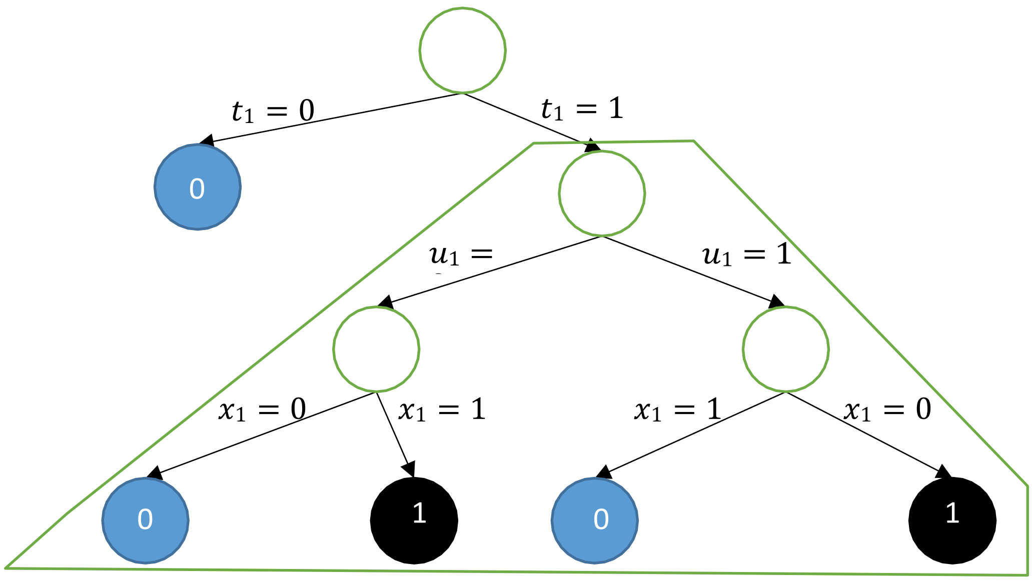

Fig. 1. SG_1

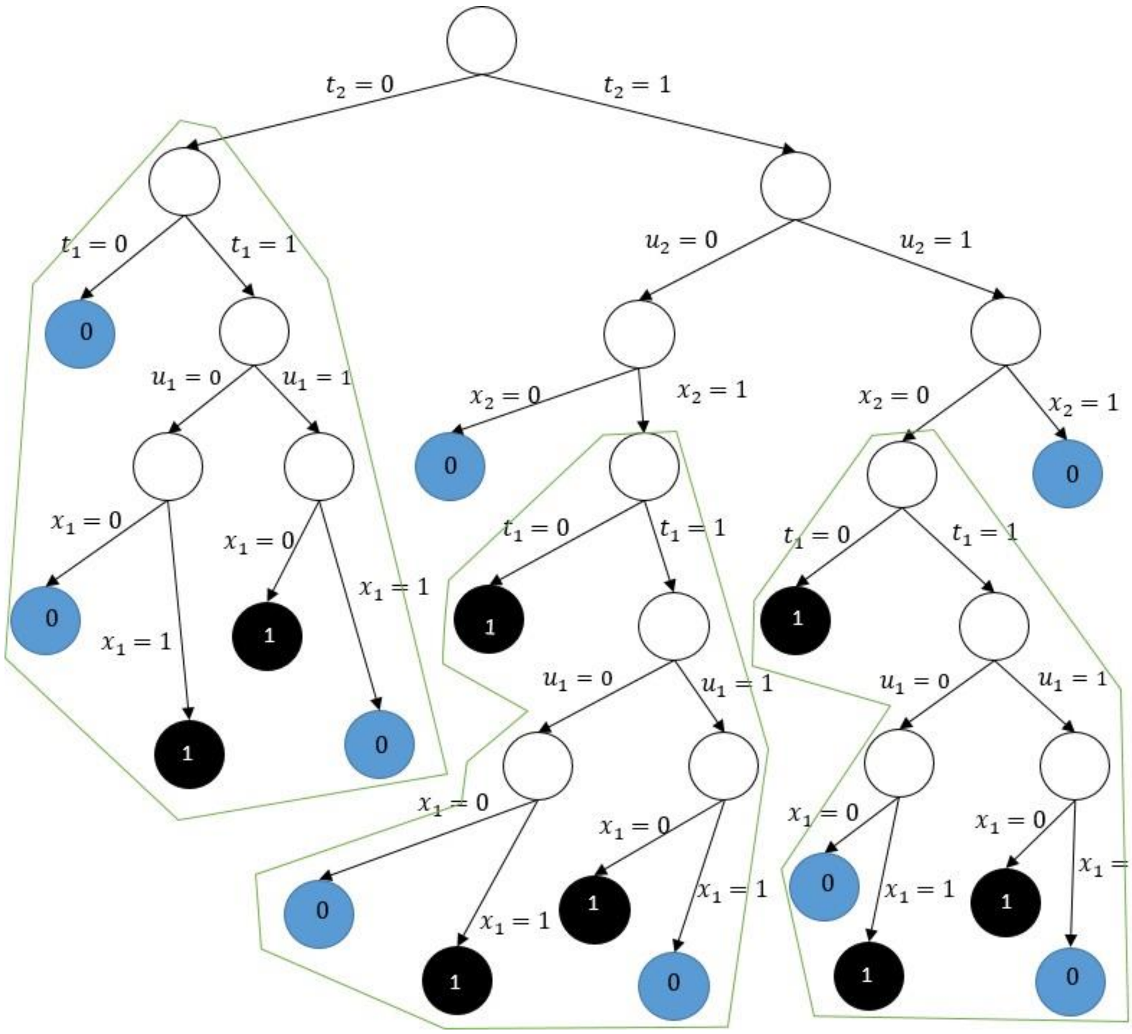

Fig 2. SG_2

It is not difficult to note that in the s.t. for SG_2-matrix the same subtree is repeated three times after splitting by $t_2 = 0$, by $t_2 = 1, u_2 = 0, x_2 = 1$ successively and by $t_2 = 1, u_2 = 1, x_2 = 0$ successively, and this subtree is the s.t. of SC_1-matrix.

It is not difficult to prove by analogy that in s.t. for SC_n-matrix the same subtree - s.t. for SC_{n-1} is repeated three times after splitting by $t_n = 0$, by $t_n = 1, u_n = 0, x_n = 1$ successively and by $t_n = 1, u_n = 1, x_n = 0$ successively.

If we denote the number of different formulas in the constructed s.t. of SG_n − matrix by (n), then we have $(1) = 6$ and $(n) = (n - 1) + 4$, therefore $(n) = 4n + 2$. Note, that the longest branch in any s.t. of SG_n −matrix must have $3n + 1$ nodes and size of SG_n −matrix $4n - 3$, therefore T-complexity of QS-proof for SG_n is $\theta(n)$ and L- complexity of QS-proof for SG_n is no more than $(4n + 2)(4n - 3) = \theta(n^2)$.

As it is mentioned in [2], the equality formulas SG_n are exponentially hard for QU-Resolution (i.e., they require proofs of exponential size), the point 2) of the theorem will be proved, if it is shown that the system QS p-simulates the system QU-Resolution. The last statement is obvious because a) the system GS p-simulates Resolution system and b) the *universal reduction rule*, $\frac{C}{\sigma(C)}$, where σ is a partial substitution that allows substituting a universal variable from a clause C by either 0 or 1, while in the system QS it is allowed to substitute every variable by either 0 or 1.

Note that analogies of the GS and QS systems can be constructed for Many-valued logic [5], which is more applicable in such fields as Formal Verification, Artificial Intelligence, Operations Research, Computational Biology and Medical Diagnosis.

4. Conclusion and Future Work

All the above results, besides their mathematical significance, have practical applications in many areas, therefore the following investigations can be useful:

a) as it is proved that the GS and QS systems are polynomially equivalent and as mentioned in the Introduction, the GS system and Frege systems are incomparable, then Frege systems and the QS system are also incomparable, therefore it is interesting to compare the QS system with the other quantifier-free and quantified systems, in particular, with quantified Frege;

b) as the introduced system has a very simple strategy for constructing proofs it is interesting to investigate how to use the many-valued analogies of the GS and QS systems in medical diagnosis.

References

[1] S.A. Cook and A. R. Reckhow, "The relative efficiency of propositional proof systems", *Journal of Symbolic Logic*, vol. 44, pp. 36-50, 1979.

[2] O. Beyersdorff, *Proof Complexity of Quantified Boolean Logic — A Survey*, World Scientific Publishing Company, Chapter 15, 2023.

[3] A. Chubaryan and Arm.Chubaryan, "Bounds of some proof complexity characteristics in the system of splitting generalization", (in Russian), *Otechestv. Nauka w epokhu izmenenij*, vol.10, no. 2(7), pp. 11-14, 2015.

[4] A. Chubaryan, S. Hovhanisyan and G. Gasparyan, "On some properties of the generalized splitting system", (in Russian), *Vestnik RAU*, vol. 2, pp. 34-42, 2019.

[5] A. Chubaryan, "Universal system for many-valued logic, based on splitting method, and some of its properties", *IJISSET*, vol. 5, no. 5, pp. 52-55, 2019. www.ijisset.org/articles/2019-2/volume-5-issue-5/

Ասույթային հաշվի ծավալիչներով տրոհման արտածման համակարգի վերաբերյալ

Անահիտ Ա. Չուբարյան

Երևանի պետական համալսարան, Երևան, Հայաստան
e-mail: achubaryan@ysu.am

Ամփոփում

Այս հոդվածում ներմուծվել է ասույթային հաշվի արտածման նոր ծավալիչներով համակարգ և ըստ արտածումների բարդությունների, այն համեմատվել է ասույթային հաշվի ծավալիչներով և առանց ծավալիչների արտածումների այլ համակարգերի հետ: Ապացուցվել է, որ ներմուծված համակարգը՝ 1) բազմանդամորեն համարժեք է իր իսկ առանց ծավալիչների տարբերակին, 2) ըստ արտածումների երկարությունների ունի ցուցչային արագացում ծավալիչներով ռեզոլյուցիոն համակարգի մի տարբերակի նկատմամբ: Քանի որ ներմուծված համակարգն ունի արտածումների որոնման շատ պարզ ընթացակարգ, այն կարող է օգտակար լինել ոչ միայն տրամաբանությունում, հետևաբար՝ արհեստական բանականության մշակումներում, այլ նաև այնպիսի ոլորտներում, ինչպիսիք են համակարգչային կենսաբանությունը և բժշկական ախտորոշումը:

Բանալի բառեր՝ Ասույթային հաշվի արտածումների ծավալիչներով համակարգ, ընդհանրացված տրոհման համակարգ, արտածման քայլեր, արտածման երկարություն, ցուցչային արագացում:

О квантифицированной системе расщеплений исчисления высказываний

Анаит А. Чубарян

Ереванский государственный университет, Ереван, Армения

e-mail: achubaryan@ysu.am

Аннотация

В данной статье введена новая квантифицированная система выводов исчисления высказываний и она сравнена по сложностям выводов с иными квантифицированными и неквантифицированными системами выводов исчисления высказываний. Доказано, что введённая система: 1) полиномиально эквивалентна своему неквантифицированному варианту, 2) имеет экспоненциальное ускорение по длинам выводов относительно некоторого варианта квантифицированной системы резолюций. Так как стратегия поиска выводов в введённой системе очень проста, то она может быть полезна не только в логике, а следовательно, при разработках искусственного интеллекта, но также в таких сферах, каковы компъютерная биология и медицинская диагностика.

Ключевые слова: квантифицированная система выводов исчисления высказываний, система обобщённых расщеплений, шаги вывода, длина вывода, экспоненциальное ускорение.