Information Processing Letters 116 (2016) 223–226

Contents lists available at ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

Tight lower bounds for the Workflow Satisfiability Problem based on the Strong Exponential Time Hypothesis

Gregory Gutin *, Magnus Wahlström

Royal Holloway, University of London, UK

ARTICLE INFO

Article history:
Received 27 August 2015
Received in revised form 26 October 2015
Accepted 7 November 2015
Available online 1 December 2015
Communicated by Ł. Kowalik

Keywords:
Workflow Satisfiability Problem
Strong Exponential Time Hypothesis
Fixed-parameter tractable
Constraint satisfaction
Computational complexity

ABSTRACT

The Workflow Satisfiability Problem (WSP) asks whether there exists an assignment of authorized users to the steps in a workflow specification, subject to certain constraints on the assignment. The problem is NP-hard even when restricted to just not equal constraints. Since the number of steps k is relatively small in practice, Wang and Li (2010) [21] introduced a parametrisation of WSP by k. Wang and Li (2010) [21] showed that, in general, the WSP is W[1]-hard, i.e., it is unlikely that there exists a fixed-parameter tractable (FPT) algorithm for solving the WSP. Crampton et al. (2013) [10] and Cohen et al. (2014) [6] designed FPT algorithms of running time $O^*(2^k)$ and $O^*(2^{k\log_2 k})$ for the WSP with so-called regular and user-independent constraints, respectively. In this note, we show that there are no algorithms of running time $O^*(2^{ck})$ and $O^*(2^{ck\log_2 k})$ for the two restrictions of WSP, respectively, with any $c < 1$, unless the Strong Exponential Time Hypothesis fails.

1. Introduction

The Workflow Satisfiability Problem (WSP) is a problem studied in the security research community, with important applications to information access control. In a WSP instance, one is given a set of k *steps* and a set of n *users*, and the goal is to find an assignment from the steps to the users, subject to some instance-specific *constraints* and *authorization lists*; see formal definition below. In practice, the number of steps tends to be much smaller than the number of users. Hence it is natural to study the problem from the perspective of *parameterized complexity*, taking k as a problem parameter. In general, the resulting parameterized problem is W[1]-hard [21], hence unlikely to be FPT, but for some natural types of constraints the problem has been shown to be FPT. In particular, Crampton et al. [10] gave an algorithm with a running time of $O^*(2^k)$ for so-called *regular* constraints, and Cohen et al. [6] gave an algorithm with a running time of $O^*(2^{k\log k})$[1] for *user-independent* constraints; see below. User-independent constraints in particular are common in the practice of access control. It was also shown that assuming the Exponential Time Hypothesis (ETH) [16], these algorithms cannot be improved to running times of $O(2^{o(k)})$ or $O(2^{o(k\log k)})$, respectively [10,6]. Still, because of the importance of the problem, the question of moderately improved running times, e.g., algorithms of running time $O(2^{ck})$, respectively $O(2^{ck\log k})$, for some $c < 1$, remained open and relevant. In this paper, we will show that no such algorithms are possible, unless the so-called Strong Exponential Time Hypothesis (SETH) [15] fails – that is, up to lower-order terms, the algorithms cited above are time optimal.

In the remainder of this section, we formally introduce the Workflow Satisfiability Problem (WSP) and some families of constraints of interest for the WSP. We briefly overview the WSP literature that considers the WSP as a parameterized problem, as suggested by Wang and Li [21],

* Corresponding author.
E-mail address: gutin@cs.rhul.ac.uk (G. Gutin).

[1] All logarithms in this paper are of base 2.

http://dx.doi.org/10.1016/j.ipl.2015.11.008
0020-0190/

and state our main results. We prove the results in the next section.

WSP In the WSP, the aim is to assign authorized users to the steps in a workflow specification, subject to some constraints arising from business rules and practices. The Workflow Satisfiability Problem has applications in information access control (e.g. see [2,3,5]), and it is extensively studied in the security research community (e.g. see [3,8,13,21]). In the WSP, we are given a set U of *users*, a set S of *steps*, a set $\mathcal{A} = \{A(s) : s \in S\}$ of *authorization lists*, where $A(s) \subseteq U$ denotes the set of users who are authorized to perform step s, and a set C of *constraints*. In general, a *constraint* $c \in C$ can be described as a pair $c = (T, \Theta)$, where $T \subseteq S$ is the *scope* of the constraint and Θ is a set of functions from T to U which specifies those assignments of steps in T to users in U that satisfy the constraint (authorizations disregarded). Authorizations and constraints described in the WSP literature are relatively simple such that we may assume that all authorisations and constraints can be checked in polynomial time (in $n = |U|$, $k = |S|$ and $m = |C|$). Given a *workflow* $W = (S, U, \mathcal{A}, C)$, W is *satisfiable* if there exists a function $\pi : S \to U$ called a *plan* such that

- π is *authorized*, i.e., for all $s \in S$, $\pi(s) \in A(s)$ (each step is allocated to an authorized user);
- π is *eligible*, i.e., for all $(T, \Theta) \in C$, $\pi|_T \in \Theta$ (every constraint is satisfied).

Wang and Li [21] were the first to observe that the number k of steps is often quite small and so can be considered as a parameter. As a result, the WSP can be studied as a parameterized problem. Wang and Li [21] proved that the WSP is *fixed-parameter tractable* (FPT) if it includes only some special types of practical constraints (authorizations can be *arbitrary* as in all other research on WSP mentioned below). This means that the WSP restricted to the types of constraints in [21] can be solved by an FPT *algorithm*, i.e., an algorithm of running time $O(f(k)(n + k + m)^{O(1)}) = O^*(f(k))$, where $f(k)$ is a computable function of k only and O^* hides polynomial factors. However, in general, the WSP is W[1]-hard [21], which means that it is highly unlikely that, in general, the WSP is FPT.[2] The paper of Wang and Li has triggered an extensive study of FPT algorithms for the WSP from both theoretical and algorithm engineering points of view. We will briefly overview literature on the topic after introduction of some important families of the WSP constraints. In what follows, for a positive integer p, $[p]$ denotes the set $\{1, 2, \ldots, p\}$.

WSP constraints We now introduce three families of WSP constraints which consecutively extend each other. Let T be a subset of S. A plan π satisfies a *steps-per-user counting constraint* (t_ℓ, t_r, T), if a user performs either no steps in T or between t_ℓ and t_r steps. Steps-per-user counting constraints generalize the cardinality constraints which have been widely adopted by the WSP community [2,4,17,20].

For $T \subseteq S$ and $u \in U$ let $\pi: T \to u$ denote the plan that assigns every step of T to u. A constraint $c = (L, \Theta)$ is *regular* if it satisfies the following condition: For any partition $L_1, \ldots, L_p$ of L such that for every $i \in [p]$ there exists an eligible[3] plan $\pi : L \to U$ and user u such that $\pi^{-1}(u) = L_i$, the plan $\bigcup_{i=1}^{p}(L_i \to u_i)$, where all u_i's are distinct, is eligible. Consider, as an example, a steps-per-user counting constraint (t_ℓ, t_r, L). Let $L_1, \ldots, L_p$ be a partition of L such that for every $i \in [p]$ there exists an eligible plan $\pi : L_i \to U$ and user u such that $\pi^{-1}(u) = L_i$. Observe that for each $i \in [p]$, we have $t_\ell \le |L_i| \le t_r$ and so the plan $\bigcup_{i=1}^{p}(L_i \to u_i)$, where all u_i's are distinct, is eligible. Thus, any steps-per-user counting constraint (t_ℓ, t_r, L) is regular.

A constraint (L, Θ) is *user-independent* if whenever $\theta \in \Theta$ and $\psi : U \to U$ is a permutation then $\psi \circ \theta \in \Theta$. In other words, user-independent constraints do not distinguish between users. Observe that all regular constraints are user-independent; however some user-independent constraints are not regular [10].

FPT algorithms for the WSP Crampton et al. [10] found a faster FPT algorithm, of running time $O^*(2^k)$, to solve the special cases of WSP studied by Wang and Li [21] and showed that the algorithm can be used for all regular constraints (all constraints studied in [21] are regular). Cohen et al. [6] showed that the WSP with only user-independent constraints is FPT and can be solved by an algorithm of running time $O^*(2^{k \log k})$. A simpler $O^*(2^{k \log k})$-time algorithm was designed by Karapetyan et al. [18] for WSP with user-independent constraints. Also an $O^*(2^{k \log k})$-time algorithm was obtained by Crampton et al. [9] for a natural optimization version of WSP, the Valued WSP, with (valued) user-independent constraints. The algorithms of these three papers were implemented in [7,18,9], respectively, and, in computational experiments, the implementations demonstrated a clear superiority of the FPT algorithms over well-known off-the-shelf solvers, the pseudo-boolean SAT solver SAT4J and the MIP solver CPLEX, for hard WSP and Valued WSP instances (in particular, the off-the-shelf solvers could not find solutions to many instances for which the FPT algorithm found solution within a few minutes).

Crampton et al. [10] and Cohen et al. [6], respectively, showed that under the Exponential Time Hypothesis (ETH) [16], there are no algorithms of running time $O^*(2^{o(k)})$ and $O^*(2^{o(k \log k)})$, respectively, for the WSP with regular and user-independent constraints, respectively. However, these results leave possibility of the existence of algorithms of running time $O^*(2^{ck})$ and $O^*(2^{ck \log k})$, respectively, with $c < 1$. Such algorithms could be of practical interest as well as theoretical, if the improvement were significant enough. The aim of this note is to show that, unfortunately, such algorithms do not exist unless the Strong Exponential-Time Hypothesis (SETH) fails. Recall that SETH [15] states that

[2] For recent excellent introductions to fixed-parameter algorithms and complexity, see, e.g., [12,14].

[3] We consider only constraints whose scope is a subset of L.