On the Parameterized Complexity of Synthesizing Boolean Petri Nets With Restricted Dependency

Ronny Tredup
Universität Rostock, Institut für Informatik, Theoretische Informatik
Albert-Einstein-Straße 22, 18059, Rostock, GErmany

ronny.tredup@uni-rostock.de

Evgeny Erofeev
Department of Computing Science, Carl von Ossietzky Universität Oldenburg,
D-26111 Oldenburg, Germany

evgeny.erofeev@informatik.uni-oldenburg.de

Modeling of real-world systems with Petri nets allows to benefit from their generic concepts of parallelism, synchronisation and conflict, and obtain a concise yet expressive system representation. Algorithms for synthesis of a net from a sequential specification enable the well-developed theory of Petri nets to be applied for the system analysis through a net model. The problem of τ-synthesis consists in deciding whether a given directed labeled graph A is isomorphic to the reachability graph of a Boolean Petri net N of type τ. In case of a positive decision, N should be constructed. For many Boolean types of nets, the problem is NP-complete. This paper deals with a special variant of τ-synthesis that imposes restrictions for the target net N: we investigate *dependency d-restricted τ-synthesis (DRτS)* where each place of N can influence and be influenced by at most d transitions. For a type τ, if τ-synthesis is NP-complete then DRτS is also NP-complete. In this paper, we show that DRτS parameterized by d is in XP. Furthermore, we prove that it is $W[2]$-hard, for many Boolean types that allow unconditional interactions set and reset.

1 Introduction

Petri nets are widely used for modeling of parallel processes and distributed systems due to their ability to express the relations of causal dependency, conflict and concurrency between system actions. In system analysis, one aims to check behavioral properties of such models, and many of these properties are decidable [12] for Petri nets and their reachability graphs which represent systems' behaviors. The task of system synthesis is opposite: A system model has to be constructed from a given specification of the desired behavior. Labeled transition systems serve as a convenient formalism for the behavioral specification, and the goal is then to construct a Petri net whose reachability graph is isomorphic to the input transition system. The relevance of the interest to the synthesis is justified in several ways. In comparison to the sequential description of the system given by a transition system, the presence of concurrency/parallelism in a Petri net on a fine-grained level allows to encompass the full interleaving in the behavior in a concise yet clear way. As a result, this yields a usually much more compact system model without loss of the expressiveness, as long as the synthesis terminates successfully. Besides, the alorithms of automatic synthesis ensure that the constructed model is correct-by-design, and hence it does not require any further verification. Moreover, the well-developed theory of Petri nets [12, 13] suggests a wide range of methods and techniques for behavioral and structural analysis of the synthesised model, supporting possible improvements and optimization purposes in the initial system. Altogether,

J. Lange, A. Mavridou, L. Safina, A. Scalas (Eds.):
13th Interaction and Concurrency Experience (ICE 2020).
EPTCS 324, 2020, pp. 78–95, doi:10.4204/EPTCS.324.7

these allow many areas to benefit from Petri net synthesis, e.g., extracting concurrency from sequential specifications like TS and languages [4], process discovery [1], supervisory control [13] or the synthesis of speed independent circuits [8].

The complexity of Petri net synthesis significantly depends on the restrictions which are implied by the specification, or imposed on the target system model, or both, and ranges from undecidable [16] via NP-complete [24, 25] down to polynomial [10, 17].

In this work, we study the complexity of synthesis for Boolean nets [3, pp. 139-152], where each place contains at most one token, for any reachable marking. A place of such a net is often considered as a Boolean condition which is *true* if the place is marked and false otherwise. In a Boolean Petri net, a place p and a transition t are related by one of the Boolean *interactions*: *no operation* (nop), *input* (inp), *output* (out), *unconditionally set to true* (set), *unconditionally reset to false* (res), *inverting* (swap), *test if true* (used), and *test if false* (free). These interactions define in which way p and t influence each other: The interaction inp (out) defines that p must be *true* (*false*) before and *false* (*true*) after t's firing; free (used) implies that t's firing proves that p is *false* (*true*); nop means that p and t do not affect each other at all; res (set) implies that p may initially be both *false* or *true* but after t's firing it is *false* (*true*); swap means that t inverts p's current Boolean value.

Boolean Petri nets are classified by the sets of interactions between places and transitions that can be applied. A set τ of Boolean interactions is called a *type of net*. A net N is of type τ (a τ-*net*) if it applies at most the interactions of τ. For a type τ, the τ-*synthesis* problem consists in deciding whether a given *transition system* A is isomorphic to the reachability graph of some τ-net N, and in constructing N if it exists. The complexity of synthesis strongly depends on the target Boolean type of nets. Thus, while τ-synthesis for elementary net systems (the case of $\tau = \{\mathsf{nop}, \mathsf{inp}, \mathsf{out}\}$) is shown to be NP-complete [2], the same problem for flip-flop nets ($\tau = \{\mathsf{nop}, \mathsf{inp}, \mathsf{out}, \mathsf{swap}\}$) is polynomial [17].

This paper addresses the computational complexity of a special instance of τ-synthesis called *Dependency d-Restricted τ-Synthesis* (DRτS), which sets a limitation for the number of connections of a place. This synthesis setting targets to those τ-nets in which every place must be in relation nop with all but at most d transitions of the net, while the synthesis input is not confined. In system modeling [15], places of Petri nets are usually meant as conditions or resources, while transitions are meant as actions or agents. Hence, the formulation of d-restricted synthesis takes into consideration not only the concurrency perspective but also possible *a priori* limitations on the number of agents which compete for the access to some resource in the modeled system. From the theoretical perspective, the problem of synthesis has been extensively studied in the literature for the conventional Petri nets and their subclasses, which are often defined via various structural restrictions: Recently, improvements of the existing synthesis techniques have been suggested for choice-free (transitions cannot share incoming places) [7], weighted marked graphs (each place has at most one input and one output transition) [10, 11], fork-attribution (choice-free and at most one input for each transition) [27] and other net classes [6, 26]. In these works, the limitations were mainly subject to the quantity of connections between places and transitions. On the other hand, the results on synthesis of k-bounded (never more than k tokens on a place) [20], safe (1-bounded) and elementary nets [3] investigate classes which are defined through behavioral restrictions. Further, generalized settings of the synthesis problem for these and some other classes were studied [21], and NP-completeness results for many of them were presented. In contrast to this multitude of P/T net classes, for Boolean nets, only the constrains for the set of interactions have appeared in the literature, deriving for instance flip-flop nets [17], trace nets [5], inhibitor nets [14]. This kind of constrain can be considered as behavioral limitation, leaving out the question of synthesis of possible structurally defined subclasses of Boolean nets. The present paper aims to piece out the shortage by investigating the notion of d-restriction which limits the amount of connections between a place and transitions. The notion was initially introduced in [22], where the complexity of d-restricted synthesis has been studied for a number of Boolean types, and the W[1]-hardness of this problem has been proven. The current paper extends the previous work and tackles the problem for many types that necessarily include interactions res and set. We demonstrate the W[2]-hardness of d-restricted synthesis for these types, which makes a clear distinction to the earlier results.

The paper is organized as follows. After introducing of the necessary definitions in Section 2, the main contributions on W[2]-hardness of DRτS are presented in Section 3. Section 4 suggests an outlook of the further research directions. Due to space restrictions, we omit some proofs, which can all be found in the technical report [23].

2 Preliminaries

In this section, we introduce the notions used throughout the paper and support them by examples.

Parameterized Complexity. Due to space restrictions, we only give the basic notions of Parameterized complexity (used in this paper) and refer to [9] for further related definitions. A *parameterized* problem is a language $L \subseteq \Sigma^* \times \mathbb{N}$, where Σ is a fixed alphabet and $\mathbb{N}$ is the set of natural numbers. For an input $(x,k) \in \Sigma^* \times \mathbb{N}$, k is called the *parameter*. We define the size of an instance (x,k), denoted by $|(x,k)|$, as $|x|+k$, that is, k is encoded in unary. Let $f,g : \mathbb{N} \to \mathbb{N}$ be two computable functions. The parameterized language L is *slice-wise polynomial* (XP), if there exists an algorithm $\mathscr{A}$ such that, for all $(x,k) \in \Sigma^* \times \mathbb{N}$, algorithm $\mathscr{A}$ decides whether $(x,k) \in L$ in time bounded by $f(k) \cdot |(x,k)|^{g(k)}$; if the runtime of $\mathscr{A}$ is even bounded by $f(k) \cdot |(x,k)|^{\mathscr{O}(1)}$, then L is called *fixed-parameter tractable* (FPT). In order to classify parameterized problems as being FPT or not, the W-hierarchy FPT $\subseteq W[1] \subseteq W[2] \subseteq \dots \subseteq$ XP is defined [9, p. 435]. It is believed that all the sub-relations in this sequence are strict and that a problem is not FPT if it is $W[i]$-hard for some $i \geq 1$. Let $L_1, L_2 \subseteq \Sigma^* \times \mathbb{N}$ be two parameterized problems. A *parameterized* reduction from L_1 to L_2 is an algorithm that given an instance (x,k) of L_1, outputs an instance (x',k') of L_2 in time $f(k) \cdot |x|^{\mathscr{O}(1)}$ for some computable function f such that (x,k) is a yes-instance of L_1 if and only if (x',k') is a yes-instance of L_2 and $k' \leq g(k)$ for some computable function g. If L_1 is $W[i]$-hard and there is a parameterized reduction from L_1 to L_2, then L_2 is $W[i]$-hard, too.

Transition Systems. A (deterministic) *transition system* (TS, for short) $A = (S,E,\delta)$ is a directed labeled graph with the set of nodes S (called *states*), the set of labels E (called *events*) and partial *transition function* $\delta : S \times E \longrightarrow S$. If $\delta(s,e)$ is defined, we say that event e *occurs* at state s, denoted by $s \xrightarrow{e}$. An *initialized* TS $A = (S,E,\delta,\iota)$ is a TS with a distinct *initial* state $\iota \in S$ where every state $s \in S$ is *reachable* from ι by a directed labeled path.

Boolean Types of Nets [3]. The following notion of Boolean types of nets allows to capture *all* Boolean Petri nets in a *uniform* way. A *Boolean type of net* $\tau = (\{0,1\}, E_\tau, \delta_\tau)$ is a TS such that E_τ is a subset of the *Boolean interactions*: $E_\tau \subseteq I = \{\mathsf{nop}, \mathsf{inp}, \mathsf{out}, \mathsf{set}, \mathsf{res}, \mathsf{swap}, \mathsf{used}, \mathsf{free}\}$. Each interaction $i \in I$ is a binary partial function $i : \{0,1\} \to \{0,1\}$ as defined in Figure 1. For all $x \in \{0,1\}$ and all $i \in E_\tau$, the transition function of τ is defined by $\delta_\tau(x,i) = i(x)$. Since a type τ is completely determined by E_τ, we often identify τ with E_τ.

τ-Nets. Let $\tau \subseteq I$. A Boolean Petri net $N = (P,T,f,M_0)$ of type τ (a τ-*net*) is given by finite disjoint sets P of *places* and T of *transitions*, a (total) *flow function* $f : P \times T \to \tau$, and an *initial marking* $M_0 : P \longrightarrow \{0,1\}$. A transition $t \in T$ can *fire* in a marking $M : P \longrightarrow \{0,1\}$ if $\delta_\tau(M(p), f(p,t))$ is defined for all $p \in P$. By firing, t produces the marking $M' : P \longrightarrow \{0,1\}$ where $M'(p) = \delta_\tau(M(p), f(p,t))$ for all $p \in P$, denoted by $M \xrightarrow{t} M'$. The behavior of τ-net N is captured by a transition system A_N, called the *reachability graph* of N. The states set $RS(N)$ of A_N consists of all markings that can be reached from initial state M_0 by sequences of transition firings. The dependency number $d_p = |\{t \in T \mid f(p,t) \neq \mathsf{nop}\}|$ of a place p of N is the number of transitions whose firing can possibly influence p or be influenced by the marking of p. The *dependency number* d_N of a τ-net N is defined as $d_N = \max\{d_p \mid p \in P\}$. For $d \in \mathbb{N}$, a τ-net is called *(dependency) d-restricted* if $d_N \leq d$.

Example 1. *Figure 2 shows the type $\tau = \{\mathsf{nop}, \mathsf{inp}, \mathsf{swap}\}$ and the 2-restricted τ-net $N = (\{R_1, R_2\}, \{a, b\}, f, M_0)$ with places R_1, R_2, flow-function $f(R_1, a) = f(R_2, b) = \mathsf{inp}$, $f(R_1, b) = \mathsf{nop}$, $f(R_2, a) = \mathsf{swap}$ and initial marking $M_0 = (M_0(R_1), M_0(R_2)) = (1, 0)$. Since $1 \xrightarrow{\mathsf{inp}} 0 \in \tau$ and $0 \xrightarrow{\mathsf{swap}} 1 \in \tau$, the transition a can fire in M_0, which leads to the marking $M = (M(R_1), M(R_2)) = (0, 1)$. After that, b can fire, which results in the marking $M' = (M'(R_1), M'(R_2)) = (0, 0)$. The reachability graph A_N of N is depicted on the right hand side of Figure 2.*

τ-Regions. Let $\tau \subseteq I$. If an input A of τ-synthesis allows a positive decision, we want to construct a corresponding τ-net N. TS represents the behavior of a modeled system by means of *global states* (states of TS) and transitions between them (events). Dealing with a Petri net, we operate with *local states* (places) and their changing (transitions), while the global states of a net are markings, i.e., combinations of local states. Since A and A_N must be isomorphic, N's transitions correspond to A's events. The connection between global states in TS and local states in the sought net is given by *regions of TS* that mimic places: A τ-region $R = (sup, sig)$ of $A = (S, E, \delta, \iota)$ consists of the *support* $sup : S \to \{0, 1\}$ and the *signature* $sig : E \to E_\tau$ where every edge $s \xrightarrow{e} s'$ of A leads to an edge $sup(s) \xrightarrow{sig(e)} sup(s')$ of type τ. If $P = q_0 \xrightarrow{e_1} \dots \xrightarrow{e_n} q_n$ is a path in A, then $P^R = sup(q_0) \xrightarrow{sig(e_1)} \dots \xrightarrow{sig(e_n)} sup(q_n)$ is a path in τ. We say P^R is the *image* of P (under R). Notice that R is *implicitly* defined by $sup(\iota)$ and sig: Since A is reachable, for every state $s \in S(A)$, there is a path $\iota \xrightarrow{e_1} \dots \xrightarrow{e_n} s_n$ such that $s = s_n$. Thus, since τ is deterministic, we inductively obtain $sup(s_{i+1})$ by $sup(s_i) \xrightarrow{e_i} sup(s_{i+1})$ for all $i \in \{0, \dots, n-1\}$ and $s_0 = \iota$. Consequently, we can compute sup and, thus, R purely from $sup(\iota)$ and sig, cf. Figure 5 and Example 3. A region (sup, sig) models a place p and the associated part of the flow function f. In particular, $f(p, e) = sig(e)$ and $M(p) = sup(s)$, for marking $M \in RS(N)$ that corresponds to $s \in S(A)$. Every set $\mathscr{R}$ of τ-regions of A defines the *synthesized τ-net* $N_A^{\mathscr{R}} = (\mathscr{R}, E, f, M_0)$ with $f((sup, sig), e) = sig(e)$ and $M_0((sup, sig)) = sup(\iota)$ for all $(sup, sig) \in \mathscr{R}, e \in E$.

State and Event Separation. To ensure that the input behavior is captured by the synthesized net, we have to distinguish global states, and prevent the firings of transitions when their corresponding

x	$\mathsf{nop}(x)$	$\mathsf{inp}(x)$	$\mathsf{out}(x)$	$\mathsf{set}(x)$	$\mathsf{res}(x)$	$\mathsf{swap}(x)$	$\mathsf{used}(x)$	$\mathsf{free}(x)$
0	0		1	1	0	1		0
1	1	0		1	0	0	1	

Figure 1: All interactions i of I. If a cell is empty, then i is undefined on the respective x.

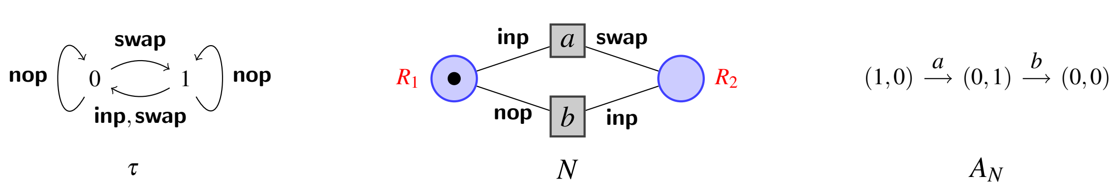

Figure 2: The type $\tau = \{\mathsf{nop}, \mathsf{inp}, \mathsf{swap}\}$ and a τ-net N and its reachability graph A_N.

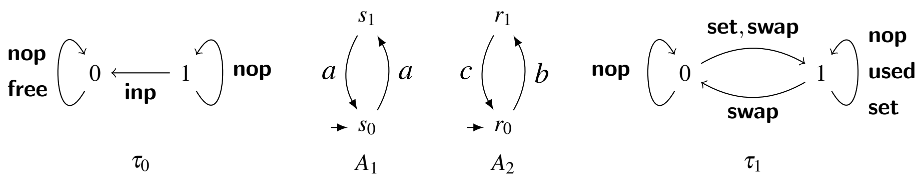

Figure 3: The type $\tau_0 = \{\mathsf{nop}, \mathsf{inp}, \mathsf{free}\}$, the TSs A_1 and A_2 and the type $\tau_1 = \{\mathsf{nop}, \mathsf{swap}, \mathsf{used}, \mathsf{set}\}$.

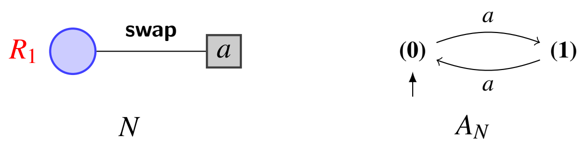

Figure 4: The 1-restricted τ_1-net N, where τ_1 is defined according to Figure 3 and $N = N^{\mathscr{R}}_{A_1}$ according to Example 2, and its reachability graph A_N.

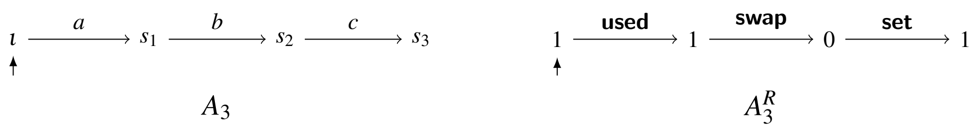

Figure 5: The TS A_3, a simple directed path and its image A_3^R under R, with R from Example 3.

events are not present in TS. This is stated as so called *separation atoms* and *problems*. A pair (s,s') of distinct states of A defines a *states separation atom* (SSP atom). A τ-region $R = (sup, sig)$ *solves* (s,s') if $sup(s) \neq sup(s')$. If every SSP atom of A is τ-solvable then A has the τ-*states separation property* (τ-SSP, for short). A pair (e,s) of event $e \in E$ and state $s \in S$ where e does not occur, that is $\neg s \xrightarrow{e}$, defines an *event/state separation atom* (ESSP atom). A τ-region $R = (sup, sig)$ *solves* (e,s) if $sig(e)$ is not defined on $sup(s)$ in τ, that is, $\neg sup(s) \xrightarrow{sig(e)}$. If every ESSP atom of A is τ-solvable then A has the τ-*event/state separation property* (τ-ESSP, for short). A set $\mathscr{R}$ of τ-regions of A is called τ-*admissible* if for each SSP and ESSP atom there is a τ-region R in $\mathscr{R}$ that solves it. We say that A is τ-solvable if it has a τ-admissible set. The next lemma establishes the connection between the existence of τ-admissible sets of A and the existence of a τ-net N that solves A:

Lemma 1 ([3])**.** *A TS A is isomorphic to the reachability graph of a τ-net N if and only if there is a τ-admissible set $\mathscr{R}$ of A such that $N = N^{\mathscr{R}}_A$.*

Example 2. *Let τ_0, τ_1, A_1 and A_2 be defined in accordance to Figure 3. The TS A_1 has no ESSP atoms. Hence, it has the τ_0-ESSP and τ_1-ESSP. The only SSP atom of A_1 is (s_0,s_1). It is τ_1-solvable by $R_1 = (sup_1, sig_1)$ with $sup_1(s_0) = 0$, $sup_1(s_1) = 1$, $sig_1(a) = \mathsf{swap}$. Thus, A_1 has the τ_1-admissible set $\mathscr{R} = \{R_1\}$, and the τ_1-net $N = N^{\mathscr{R}}_A = (\{R_1\}, \{a\}, f, M_0)$ with $M_0(R_1) = sup_1(s_0)$ and $f(R_1,a) = sig_1(a)$ solves A_1. Figure 4 depicts N (left) and its reachability graph A_N (right). The SSP atom (s_0,s_1) is not τ_0-solvable, thus, neither is A_1. TS A_2 has ESSP atoms (b,r_1) and (c,r_0), which are both τ_1-unsolvable. The only SSP atom (r_0,r_1) in A_2 can be solved by the τ_1-region $R_2 = (sup_2, sig_2)$ with $sup_2(r_0) = 0$, $sup_2(r_1) = 1$, $sig_2(b) = \mathsf{set}$, $sig_2(c) = \mathsf{swap}$. Thus, A_2 has the τ_1-SSP, but not the τ_1-ESSP. None of the (E)SSP atoms of A_2 can be solved by any τ_0-region. Notice that the τ_1-region R_2 maps two events to a signature different from nop. Thus, in case of d-restricted τ_1-synthesis, R_2 would not be valid for $d = 1$.*

Example 3. *Let A_3 be defined in accordance to Figure 5 and τ_1 according to Figure 3. It defines $sup(\iota) = 1$, $sig(a) = \mathsf{used}$, $sig(b) = \mathsf{swap}$ and $sig(c) = \mathsf{set}$ implicitly a τ_1-region $R = (sup, sig)$ of A_3 as follows: $sup(s_1) = \delta_{\tau_1}(1, \mathsf{used}) = 1$, $sup(s_2) = \delta_{\tau_1}(1, \mathsf{swap}) = 0$ and $sup(s_3) = \delta_{\tau_1}(0, \mathsf{set}) = 1$. The image A_3^R of A_3 (under R) is depicted on the right hand side of Figure 5. One easily verifies that $\delta_{A_3}(s,e) = s'$ implies $\delta_{\tau_1}(sup(s), sig(e)) = sup(s')$, cf. Figure 3.*

By Lemma 1, every τ-admissible set $\mathscr{R}$ implies that $N_A^{\mathscr{R}}$ τ-solves A. In this paper, we investigate the complexity of synthesising a solving τ-net N whose dependency number d_N does not exceed a natural number d. Recall that if $\mathscr{R}$ is a set of A's regions, then $\mathscr{R}$'s regions model places of the synthesized net $N_A^{\mathscr{R}}$. Thus, $d_{N_A^{\mathscr{R}}} \leq d$ if and only if $\mathscr{R}$ is *d-restricted*, that is, every region $R = (sup, sig)$ of $\mathscr{R}$ is *d-restricted*: $|\{e \in E \mid sig(e) \neq \mathsf{nop}\}| \leq d$. By Lemma 1, this implies that there is a d-restricted τ-net N if and only if there is a d-restricted τ-admissible set $\mathscr{R}$. This finally leads to the following parameterized problem that is the main subject of this paper:

Dependency Restricted τ-Synthesis (DRτS)

Input: a finite, reachable TS A, a natural number d.

Parameter: d

Decide: whether there exists a d-restricted τ-admissible set $\mathscr{R}$ of A.

3 Dependency d-Restricted τ-Synthesis

For a start, we observe that, similar to (unrestricted) τ-synthesis [19], DRτS is in NP. Moreover, there is a trivial reduction from τ-synthesis to DRτS: Since a τ-region can map at most all events of a TS $A = (S, E, \delta, \iota)$ not to nop, A is τ-solvable if and only if A is τ-solvable by $|E|$-restricted τ-regions. Thus, if τ-synthesis is NP-complete, then DRτS is also NP-complete.

Let's argue that DRτS belongs to the complexity class XP. Let $A = (S, E, \delta, \iota)$ be a TS, $d \in \mathbb{N}$ and let $|A|$ be the maximum number of edges that A possibly has, that is, $|A| = |S|^2|E|$. A τ-region $R = (sup, sig)$ is implicitly defined by $sup(\iota)$ and sig. We are interested in regions of A for which there is an $i \in \{0, \dots, d\}$ such that $|\{e \in E \mid sig(e) \neq \mathsf{nop}\}| = i$. For every event $e \in E$, we have at most $|\tau| - 1 \leq 7$ interactions that are different from nop. Since $sup(\iota) \in \{0, 1\}$, we have to consider at most $2 \cdot 7^d \cdot \sum_{i=0}^{d} \binom{|E|}{i}$ regions at all, which can be estimated by $\mathscr{O}(7^d |A|^d)$

To check if the chosen signature actually implies regions of A and to solve the (E)SSP atoms of A, we need to construct the regions explicitly, that is, we have to compute sup. To do so, we firstly compute a spanning tree A' of A, which is doable in time $\mathscr{O}(|A|^2)$ by the algorithm of Tarjan [18] and needs to be done only once. In A', there is exactly one path from ι to s for all $s \in S$, and A' has $|S| - 1$ edges. Thus, having a spanning tree, $sup(\iota)$ and sig, it costs time at most $\mathscr{O}(|A|)$ to compute sup. The effort to compute all *potentially* interesting regions explicitly is thus at most $\mathscr{O}(7^d |A|^{d+1})$. After that, we check for any fixed potential region if it is actually a well-defined region, that is, whether $s \xrightarrow{e} s'$ implies $sup(s) \xrightarrow{sig(e)} sup(s')$. For a fixed region, this is doable in time $\mathscr{O}(|A|)$. Thus the effort to compute all interesting regions of A is $\mathscr{O}(7^d |A|^{d+2})$.

For a fixed separation atom (s, s') or (e, s) we simply have to check if $sup(s) \neq sup(s')$ or if $\delta_\tau(sup(s), sig(e))$ is not defined, respectively, which is doable in time $\mathscr{O}(|A|)$. Since we have at most $\mathscr{O}(|A|^2)$ separation atoms and at most $\mathscr{O}(7^d |A|^d)$ regions, the check for the (E)SSP is doable in time $\mathscr{O}(7^d |A|^{d+2})$. Finally, if we add up the effort to get all interesting regions and the effort to check whether these regions witness the (E)SSP of A, then we obtain that the effort of the problem is bounded by $\mathscr{O}(7^d |A|^{d+2})$.

On the other hand, in this section, we argue that DRτS is $W[2]$-hard for a range of Boolean types. The following theorem presents the result through the enumeration of these types.

Theorem 1. *Dependency d-Restricted τ-Synthesis is $W[2]$-hard if*

1. $\tau \supseteq \{\mathsf{nop}, \mathsf{inp}, \mathsf{set}\}$ *or* $\tau \supseteq \{\mathsf{nop}, \mathsf{out}, \mathsf{res}\}$,
2. $\tau = \{\mathsf{nop}, \mathsf{set}, \mathsf{res}\} \cup \omega$ *or* $\tau = \{\mathsf{nop}, \mathsf{set}, \mathsf{res}, \mathsf{swap}\} \cup \omega$ *and* $\emptyset \neq \omega \subseteq \{\mathsf{free}, \mathsf{used}\}$,
3. $\tau = \{\mathsf{nop}, \mathsf{set}, \mathsf{swap}\} \cup \omega$, $\tau = \{\mathsf{nop}, \mathsf{out}, \mathsf{set}, \mathsf{swap}\} \cup \omega$, $\tau = \{\mathsf{nop}, \mathsf{res}, \mathsf{swap}\} \cup \omega$ *or* $\tau = \{\mathsf{nop}, \mathsf{inp}, \mathsf{res}, \mathsf{swap}\} \cup \omega$ *and* $\emptyset \neq \omega \subseteq \{\mathsf{free}, \mathsf{used}\}$,
4. $\tau = \{\mathsf{nop}, \mathsf{inp}, \mathsf{res}, \mathsf{swap}\}$ *or* $\tau = \{\mathsf{nop}, \mathsf{out}, \mathsf{set}, \mathsf{swap}\}$,

Notice that, by the discussion above, for the types of Theorem 1, NP-completeness of DRτS follows by the NP-completeness of τ-synthesis [19, p. 3]. The proof of Theorem 1 bases on parameterized reductions of the problem *Hitting Set*, which is known to be $W[2]$-complete (see e.g. [9]). The problem *Hitting Set* is defined as follows:

Hitting Set (HS)

Input: a finite set $\mathfrak{U}$, a set $M = \{M_1, \dots, M_m\}$ of subsets of $\mathfrak{U}$ with $M_i = \{X_{i_1}, \dots, X_{i_{m_i}}\}$ and $i_1 < \dots < i_{m_i}$ for all $i \in \{1, \dots, m\}$, a natural number κ.

Parameter: κ

Decide: whether there is a set $S \subseteq \mathfrak{U}$ such that $|S| \leq \kappa$ and $S \cap M_i \neq \emptyset$ for every $i \in \{1, \dots, m\}$.

The General Reduction Idea. An input $I = (\mathfrak{U}, M, \kappa)$ of HS, where $M = \{M_1, \dots, M_m\}$, is reduced to an instance (A_I^τ, d) of DRτS with TS A_I^τ and $d = f(\kappa)$, for some linear function f. For every $i \in \{1, \dots, m\}$, the TS A_I^τ has a directed labeled path

$$P_i = \quad s_{i,0} \xrightarrow{X_{i_1}} \cdots \xrightarrow{X_{i_{\ell-1}}} s_{i,i_{\ell-1}} \xrightarrow{X_{i_\ell}} s_{i,i_\ell} \xrightarrow{X_{i_{\ell+1}}} \cdots \xrightarrow{X_{i_{m_i}}} s_{i,i_{m_i}}$$

that represents the set $M_i = \{X_{i_1}, \dots, X_{i_{m_i}}\}$ and uses its elements as events. The TS A_I^τ is then composed in such a way that for some ESSP atom α of A_I^τ the following is satisfied: If $R = (sup, sig)$ is a d-restricted τ-region that solves α, then $sup(s_{i,0}) \neq sup(s_{i,i_{m_i}})$ for all $i \in \{1, \dots, m\}$. Since the image P_i^R of P_i is a directed path in τ, by $sup(s_{i,0}) \neq sup(s_{i,i_{m_i}})$, there has to be an element $X \in M_i$ such that $s \xrightarrow{X} s' \in P_i$ implies $sup(s) \neq sup(s')$. That is, the image $sig(X)$ of X causes a state change on P_i^R in τ. In particular, this implies $sig(X) \neq \mathsf{nop}$. The following visualisation of P_i^R sketches the situation for a region $R = (sup, sig)$, where $sup(s_{i,0}) = \dots = sup(s_{i,i_{\ell-1}}) = 0$ and $sup(s_{i,i_\ell}) = \dots = sup(s_{i,i_{m_i}}) = 1$ and $sig(X_{i_\ell}) = \mathsf{set}$ and $sig(X_{i_k}) = \mathsf{nop}$ for all $k \in \{1, \dots, m_i\} \setminus \{\ell\}$:

$$P_i^R = \quad \underset{0}{sup(s_{i,0})} \xrightarrow[\textbf{nop}]{sig(X_{i_1})} \cdots \xrightarrow[\textbf{nop}]{sig(X_{i_{\ell-1}})} \underset{0}{sup(s_{i,i_{\ell-1}})} \xrightarrow[\textbf{set}]{sig(X_{i_\ell})} \underset{1}{sup(s_{i,i_\ell})} \xrightarrow[\textbf{nop}]{sig(X_{i_{\ell+1}})} \cdots \xrightarrow[\textbf{nop}]{sig(X_{i_{m_i}})} \underset{1}{sup(s_{i,i_{m_i}})}$$

It is simultaneously true for all paths $P_1, \dots, P_m$ representing the sets $M_1, \dots, M_m$, that on each path there is a (not necessarily unique) X satisfying $sig(X) \neq \mathsf{nop}$. Moreover, the reduction ensures that $|\{X \in \mathfrak{U} \mid sig(X) \neq \mathsf{nop}\}| \leq \kappa$. In other words, $S = \{X \in \mathfrak{U} \mid sig(X) \neq \mathsf{nop}\}$ defines a sought hitting set of I. Thus, if (A_I^τ, d) is a yes-instance of DRτS, implying the solvability of α, then $I = (\mathfrak{U}, M, \kappa)$ is a yes-instance of HS.

Conversely, if $I = (\mathfrak{U}, M, \kappa)$ is a yes-instance, then there is a fitting τ-region of A_I^τ that solves α. The reduction ensures that the d-restricted τ-solvability of α implies that all (E)SSP atoms of A_I^τ are solvable by d-restricted τ-regions. Thus, (A_I^τ, d) is a yes-instance, too.

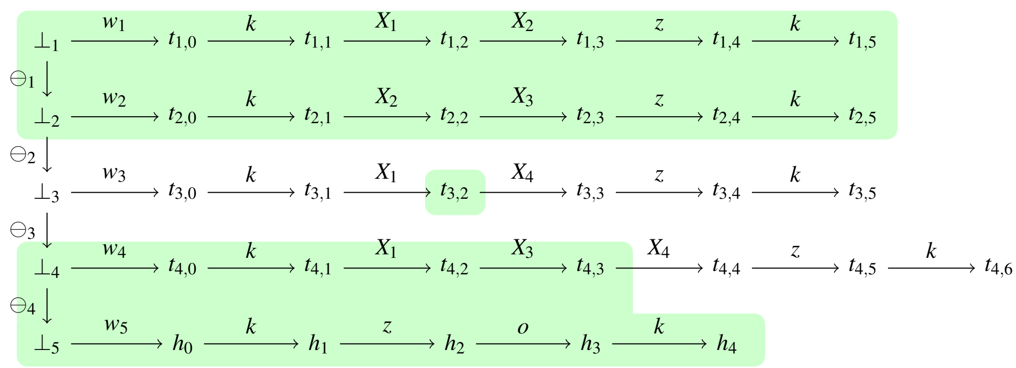

Figure 6: The TS A^τ_I, where $\tau \supseteq \{\mathsf{nop}, \mathsf{inp}, \mathsf{set}\}$ and I originates from Example 4. The green colored area sketches the states that are mapped to 1 by the region $R^{X,2}_{3,2}$ solving (X_4, s) for all $s \in \{\bot_3, t_{3,0}, t_{3,1}\}$.

In the following, we present the corresponding reductions and show that the solvability of α implies the existence of a sought-for hitting set. Moreover, we argue that the existence of a sought set implies the τ-solvability of α and, finally, the τ-solvability of A^τ_I.

As an instance, the following (running) example serves for all concrete reductions that we present, to simplify the understanding of the reductions' formal descriptions.

Example 4. *The input* $I = (\mathfrak{U}, M, \kappa)$ *is defined by* $\mathfrak{U} = \{X_1, X_2, X_3, X_4\}$ *and* $M = \{M_1, M_2, M_3, M_4\}$*, where* $M_1 = \{X_1, X_2\}$*,* $M_2 = \{X_2, X_3\}$*,* $M_3 = \{X_1, X_4\}$ *and* $M_4 = \{X_1, X_3, X_4\}$*, and* $\kappa = 2$*. A fitting hitting set of* M *is given by* $S = \{X_1, X_3\}$*.*

3.1 The Proof of Theorem 1.1

Theorem 1.1: The Reduction. In accordance to our general approach, we first define $d = \kappa + 2$. Next, we introduce the TS A^τ_I. Figure 6 provides a concrete example of A^τ_I, where I corresponds to Example 4. The TS A^τ_I has the following gadget H that applies the events k, z and o and provides the atom $\alpha = (k, h_2)$:

$$\bot_{m+1} \xrightarrow{w_{m+1}} h_0 \xrightarrow{k} h_1 \xrightarrow{z} h_2 \xrightarrow{o} h_3 \xrightarrow{k} h_4$$

For all $i \in \{1, \dots, m\}$, the TS A^τ_I has the following gadget T_i that applies w_i, k, z and the elements of $M_i = \{X_{i_1}, \dots, X_{i_{m_i}}\}$ as events:

$$\bot_i \xrightarrow{w_i} t_{i,0} \xrightarrow{k} t_{i,1} \xrightarrow{X_{i_1}} \dots \xrightarrow{X_{i_{m_i}}} t_{i,m_i+1} \xrightarrow{z} t_{i,m_i+2} \xrightarrow{k} t_{i,m_i+3}$$

The TS A^τ_I has the events $\ominus_1, \dots, \ominus_m$ to connect the gadgets $T_1, \dots, T_m$ and H by $\bot_1 \xrightarrow{\ominus_1} \dots \xrightarrow{\ominus_m} \bot_{m+1}$. The initial state of A^τ_I is $\bot_1$.

Theorem 1.1: The Solvability of α Implies a Hitting Set. We argue for $\tau \supseteq \{\mathsf{nop}, \mathsf{inp}, \mathsf{set}\}$, the hardness of the other types follows by symmetry. In the following, we argue that if there is a d-restricted τ-region $R = (sup, sig)$ that solves α, then I has a hitting set of size at most κ. Let $R = (sup, sig)$ be such a τ-region. Since R solves α, we have either $sig(k) \in \{\mathsf{inp}, \mathsf{used}\}$ and $sup(h_2) = 0$ or $sig(k) \in \{\mathsf{out}, \mathsf{free}\}$ and $sup(h_2) = 1$. In what follows, we consider to the former case. The proof for the latter case is symmetrical.

If $sig(k) = \mathsf{inp}$ and $sup(h_2) = 0$, then $s \xrightarrow{k} s'$ implies $sup(s) = 1$ and $sup(s') = 0$. By $sup(h_2) = 0$ and $sup(h_3) = 1$, we get $sig(o) \in \{\mathsf{out}, \mathsf{set}, \mathsf{swap}\}$. In particular, since R is d-restricted, there are at most κ events left that have a signature different from nop. By $sup(h_1) = sup(h_2) = 0$ and $h_1 \xrightarrow{z} h_2$, we have $sig(z) \in \{\mathsf{nop}, \mathsf{res}, \mathsf{free}\}$. Moreover, by $sup(t_{i,m_i+2}) = 1$ and $\xrightarrow{z} t_{i,m_i+2}$, we have $sig(z) = \mathsf{nop}$. By $sig(k) = \mathsf{inp}$ and $sig(z) = \mathsf{nop}$, we conclude $sup(t_{i,1}) = 0$ and $sup(t_{i,m_i+1}) = 1$ for all $i \in \{1, \dots, m\}$. Consequently, for every $i \in \{1, \dots, m\}$, there is $X \in M_i$ such that $sig(X) \in \{\mathsf{out}, \mathsf{set}, \mathsf{swap}\}$. Otherwise a state change from 0 to 1 would not be possible. Since R is d-restricted and $sig(k) \neq \mathsf{nop} \neq sig(o)$, we get $|\{X \in \mathfrak{U} \mid sig(X) \neq \mathsf{nop}\}| \leq \kappa$. This implies that $S = \{X \in \mathfrak{U} \mid sig(X) \neq \mathsf{nop}\}$ is a fitting hitting set of I.

If $sig(k) = \mathsf{used}$ and $sup(h_2) = 0$, then $s \xrightarrow{k} s'$ implies $sup(s) = sup(s') = 1$. By $sup(h_1) = sup(h_3) = 1$ and $sup(h_2) = 0$, we get $sig(z) \in \{\mathsf{inp}, \mathsf{res}, \mathsf{swap}\}$ and $sig(o) \in \{\mathsf{out}, \mathsf{set}, \mathsf{swap}\}$. By $sup(t_{i,m_i+2}) = 1$ and $\xrightarrow{z} t_{i,m_i+2}$, we get $sig(z) = \mathsf{swap}$. Since R is d-restricted, there are at most $\kappa - 1$ events left whose signature is different from nop. Moreover, by $sig(k) = \mathsf{used}$ and $sig(z) = \mathsf{swap}$, we have $sup(t_{i,1}) = 1$ and $sup(t_{i,m_i+1}) = 0$ for all $i \in \{1, \dots, m\}$. Just like before, we conclude that $S = \{X \in \mathfrak{U} \mid sig(X) \neq \mathsf{nop}\}$ is a sought hitting set of I.

Conversely, a κ-HS of $(\mathfrak{U}, M, \kappa)$ implies the τ-solvability of A_I^τ, which is the statement of the following lemma. Due to space restrictions, we omit the proof which can be found in [23].

Lemma 2. *Let τ be a type of nets in correspondence of Theorem 1.1. If $(\mathfrak{U}, M, \kappa)$ has a κ-HS, then there is a d-restricted admissible set of A_τ^I.*

3.2 The Proof of Theorem 1.2

Theorem 1.2: The Reduction. Let τ be a type of Theorem 1.2. According to our general approach, we first define $d = \kappa + 4$. Next we introduce the TS A_I^τ. Figure 7 provides an example of A_I^τ, where I corresponds to Example 4. The TS A_I^τ has the following gadget H_1 that provides the atom $\alpha = (k, h_{1,2})$:

$$\begin{tikzcd} \bot_{m+1} \arrow[r, "w_{m+1}"] & h_{1,0} \arrow["w_{m+1}", loop above] \arrow[r, "k"] & h_{1,1} \arrow["k", loop above] \arrow[r, "o_1"] & h_{1,2} \arrow["o_1", loop above] \arrow[r, "o_2"] & h_{1,3} \arrow["o_2", loop above] \arrow[r, "k"] & h_{1,4} \arrow["k", loop above] \end{tikzcd}$$

Moreover, the TS A_I^τ has the following gadgets H_2 and H_3:

$$H_2 = \begin{tikzcd} \bot_{m+2} \arrow[r, "w_{m+2}"] & h_{2,0} \arrow["w_{m+2}", loop above] \arrow[r, "k"] & h_{2,1} \arrow["k", loop above] \arrow[r, "z_1"] & h_{2,2} \arrow["{z_1, o_1}", loop above] \end{tikzcd} \qquad H_3 = \begin{tikzcd} \bot_{m+3} \arrow[r, "w_{m+3}"] & h_{3,0} \arrow["{w_{m+3}, o_1, z_2}", loop above] \end{tikzcd}$$

For all $i \in \{1, \dots, m\}$, TS A_I^τ has the following gadget T_i that applies w_i, k, z_1, z_2 and the elements of $M_i = \{X_{i_1}, \dots, X_{i_{m_i}}\}$ as events:

$$\begin{tikzcd} \bot_i \arrow[r, "w_i"] & t_{i,0} \arrow["w_i", loop above] \arrow[r, "k"] & t_{i,1} \arrow["k", loop above] \arrow[r, "z_1"] & t_{i,2} \arrow["z_1", loop above] \arrow[r, "X_{i_1}"] & {} \arrow["X_{i_1}", loop above] & \dots \arrow[r, "X_{i_{m_i}}"] & t_{i,m_i+2} \arrow["X_{i_{m_i}}", loop above] \arrow[r, "z_2"] & t_{i,m_i+3} \arrow["z_2", loop above] \arrow[r, "k"] & t_{i,m_i+4} \arrow["k", loop above] \end{tikzcd}$$

Finally, the TS A_I^τ uses the events $\ominus_1, \dots, \ominus_{m+2}$ and applies for all $i \in \{1, \dots, m\}$ the edges $\bot_i \xrightarrow{\ominus_i} \bot_{i+1}$ and $\bot_{i+1} \xrightarrow{\ominus_i} \bot_{i+1}$ to join the gadgets $T_1, \dots, T_m$ and H_1, H_2, H_3.

Theorem 1.2: The τ-solvability of α Implies a Hitting Set. Let $R = (sup, sig)$ be a τ-region that solves α, that is, either $sig(k) = \mathsf{used}$ and $sup(h_{1,2}) = 0$ or $sig(k) = \mathsf{free}$ and $sup(h_{1,2}) = 1$. In the following, we assume that $sig(k) = \mathsf{used}$ and $sup(h_{1,2}) = 0$. The arguments for the case $sig(k) = \mathsf{free}$

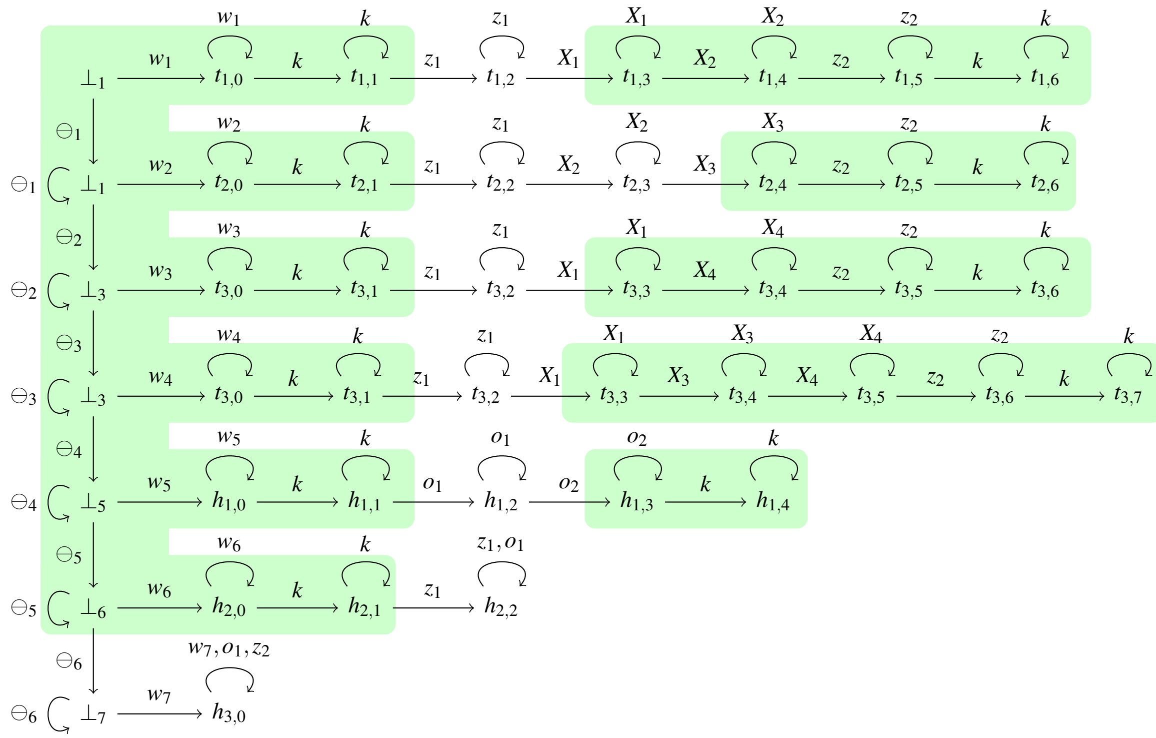

Figure 7: The TS A_I^τ where τ corresponds to Theorem 1.2 and I to Example 4 with the HS $S = \{X_1, X_3\}$. The green colored area sketches the τ-region $R = (sup, sig)$ that solves α, where, for all $e \in E(A_I^\tau)$, if $e = k$, then $sig(e) = \mathsf{used}$; if $e \in \{o_2\} \cup S$, then $sig(e) = \mathsf{set}$; if $e \in \{o_1, z_1, \ominus_6\}$, then $sig(e) = \mathsf{res}$; otherwise $sig(e) = \mathsf{nop}$.

and $sup(h_{1,2}) = 1$ are symmetrical. Notice that if $s \xrightarrow{e} s' \in A_I^\tau$, then $s' \xrightarrow{e} s' \in A_I^\tau$. Thus, for all $e \in E(A_I^\tau)$ holds $sig(e) \neq \mathsf{swap}$.

Since $sig(k) = \mathsf{used}$, if $s \xrightarrow{k} s'$, then $sup(s) = sup(s') = 1$. In particular, we have $sup(t_{i,m_i+3}) = 1$ for all $i \in \{1, \dots, m\}$. Moreover, by $sup(h_{1,1}) = 1$ and $sup(h_{1,2}) = 0$, we have $sig(o_1) = \mathsf{res}$ and $sig(o_2) = \mathsf{set}$. This implies $sup(h_{2,2}) = sup(h_{3,0}) = 0$. By $sup(h_{2,1}) = 1$ and $sup(h_{2,2}) = 0$, we get $sig(z_1) = \mathsf{res}$; by $sup(h_{3,0}) = 0$ and $sup(t_{1,m_i+3}) = 1$, we get $sig(z_2) = \mathsf{nop}$. Thus, by $sig(z_1) = \mathsf{res}$ and $sig(z_2) = \mathsf{nop}$, we get $sup(t_{i,2}) = 0$ and $sup(t_{i,m_i+2}) = 1$ for all $i \in \{1, \dots, m\}$. Consequently, for all $i \in \{1, \dots, m\}$, there is $X \in M_i$ such that $sig(X) = \mathsf{set}$. Since $sig(e) \neq \mathsf{nop}$ for all $e \in \{k, o_1, o_2, z_1\}$ and R is d-restricted, it holds $|\{X \in \mathfrak{U} \mid sig(X) \neq \mathsf{nop}\}| \leq \kappa$. This implies that $S = \{X \in \mathfrak{U} \mid sig(X) \neq \mathsf{nop}\}$ is a sought-for hitting set of I.

In return, if $(\mathfrak{U}, M, \kappa)$ has a κ-HS, then A_τ^I is τ-solvable, which is the statement of the following lemma. Due to space restrictions, we omit the proof which can be found in [23].

Lemma 3. *Let τ be a type of nets in correspondence of Theorem 1.2. If $(\mathfrak{U}, M, \kappa)$ has a κ-HS, then there is a d-restricted admissible set of A_τ^I.*

3.3 Proof of Theorem 1.3

Theorem 1.3: The Reduction. We restrict ourselves to the case where $\tau = \{\mathsf{nop},\mathsf{set},\mathsf{swap}\} \cup \omega$ or $\tau = \{\mathsf{nop},\mathsf{out},\mathsf{set},\mathsf{swap}\} \cup \omega$ and $\emptyset \neq \omega \subseteq \{\mathsf{free},\mathsf{used}\}$. The hardness for the other types follows by symmetry. First, we define $d = \kappa+4$. Next, we introduce the TS A_I^τ. Figure 8 provides a full example of A_I^τ where I corresponds to Example 4.

The TS A_I^τ has the following gadgets H_0 and H_1 that provide the atom $\alpha = (k,h_{0,3})$:

$$H_0 = \bot_{m+1} \overset{w_{m+1}}{\longleftrightarrow} h_{0,1} \overset{k}{\longleftrightarrow} h_{0,2} \overset{o_1}{\longleftrightarrow} h_{0,3} \overset{o_2}{\longleftrightarrow} h_{0,4} \overset{k}{\longleftrightarrow} h_{0,5}$$

$$H_1 = \bot_{m+2} \overset{w_{m+2}}{\longleftrightarrow} h_{1,1} \overset{k}{\longleftrightarrow} h_{1,2} \overset{z_1}{\longleftrightarrow} h_{1,3} \overset{o_1}{\longleftrightarrow} h_{1,4} \overset{z_2}{\longleftrightarrow} h_{1,5} \overset{k}{\longleftrightarrow} h_{1,6}$$

Moreover, for every $i \in \{1,\dots,m\}$, the TS A_I^τ has the following gadget T_i that has the elements of $M_i = \{X_{i_1},\dots,X_{i_{m_i}}\}$ as events:

$$\bot_i \overset{w_i}{\longleftrightarrow} t_{i,0} \overset{k}{\longleftrightarrow} t_{i,1} \overset{z_1}{\longleftrightarrow} t_{i,2} \overset{a_{i,1}}{\longleftrightarrow} t_{i,3} \overset{X_{i_1}}{\longrightarrow} t_{i,4} \overset{X_{i_1}}{\longleftrightarrow} t_{i,5} \overset{a_{i,1}}{\longleftrightarrow} t_{i,6}$$

$$\vdots$$

$$t_{i,4m_i+4} \overset{k}{\longleftrightarrow} t_{i,4m_i+3} \overset{z_2}{\longleftrightarrow} t_{i,4m_i+2} \overset{a_{i,m_i}}{\longleftrightarrow} t_{i,4m_i+1} \overset{X_{i_{m_i}}}{\longleftrightarrow} t_{i,4m_i} \overset{X_{i_{m_i}}}{\longleftarrow} t_{i,4m_i-1} \overset{a_{i,m_i}}{\longleftrightarrow} t_{i,4m_i-2}$$

Notice that, for all $\ell \in \{1,\dots,m_i\}$, the event $a_{i,\ell}$ that encompasses the event X_{i_ℓ} of M_i is bounded to the occurrence of X_{i_ℓ} in T_i. In particular, if two distinct sets M_i and M_j share an event $X \in \mathfrak{U}$, that is, there are indices $\ell \in \{1,\dots,m_i\}$ and $n \in \{1,\dots,m_j\}$ such that $X = X_{i_\ell} = X_{j_n}$, then $a_{i,\ell}$ embraces X in T_i and $a_{j,n}$ embraces X in T_j but $a_{i,\ell}$ and $a_{j,n}$ are distinct. Finally, to obtain A_I^τ, we use fresh events $\ominus_1,\dots,\ominus_{m+1}$ and connect $T_1,\dots,T_m,H_0$ and H_1 by $\bot_1 \overset{\ominus_1}{\longleftrightarrow} \dots \overset{\ominus_{m+1}}{\longleftrightarrow} \bot_{m+2}$. The initial state of A_I^τ is $\bot_1$. Notice that for every region R of A_I^τ, holds that $s \overset{e}{\longleftrightarrow} s' \in A_I^\tau$ and $sup(s) \neq sup(s')$ implies $sig(e) = \mathsf{swap}$. Moreover, if $s \overset{e}{\longrightarrow} s' \in A_I^\tau$, then, by construction, $s' \overset{e}{\longrightarrow}$. By the definition of out, this implies $sig(e) \neq \mathsf{out}$ for all $e \in E(A_I^\tau)$.

Theorem 1.3: The τ-Solvability of α Implies a Hitting Set. Let $R = (sup,sig)$ be a τ-region that solves α. Since R solves α, we have either $sig(k) = \mathsf{used}$ and $sup(h_{0,3}) = 0$ or $sig(k) = \mathsf{free}$ and $sup(h_{0,3}) = 1$. In the following, we consider the former case, the arguments for the latter are symmetrical. Please note Figure 8 during the following considerations. By $sig(k) = \mathsf{used}$, we have that $sup(s) = sup(s') = 1$ for all $s \overset{k}{\longleftrightarrow} s' \in A_I^\tau$. In particular, we have $sup(h_{0,2}) = sup(h_{0,4}) = 1$ which, by $sup(h_{0,3}) = 0$, implies $sig(o_1) = sig(o_2) = \mathsf{swap}$. Moreover, we have $sup(h_{1,2}) = sup(h_{1,5}) = 1$. Consequently, the number of state changes on the image P^R of the path $P = h_{1,2} \overset{z_1}{\longrightarrow} \dots \overset{z_2}{\longrightarrow} h_{1,5}$ is even. Since $sig(o_1) = \mathsf{swap}$, this implies that there is exactly one event $e \in \{z_1,z_2\}$ such that $sig(e) = \mathsf{swap}$. We consider the case $sig(z_1) = \mathsf{swap}$. The arguments for the case $sig(z_2) = \mathsf{swap}$ are similar. The region R is d-restricted, and k,o_1,o_2,z_1 have signatures different from nop. There are at most κ events left whose signatures are not nop.

Let $i \in \{1,\dots,m\}$ be arbitrary but fixed. By $sig(k) = \mathsf{used}$, we have $sup(t_{i,1}) = sup(t_{i,4m_i+3}) = 1$. By $sig(z_1) = \mathsf{swap}$ and $sig(z_2) \neq \mathsf{swap}$, this implies $sup(t_{i,2}) = 0$ and $sup(t_{i,m_i+2}) = 1$. Hence the image P^R of the path $P =$

$$t_{i,2} \overset{a_{i,1}}{\longleftrightarrow} t_{i,3} \overset{X_{i_1}}{\longrightarrow} t_{i,4} \overset{X_{i_1}}{\longleftrightarrow} t_{i,5} \overset{a_{i,1}}{\longleftrightarrow} t_{i,6} \quad \dots \quad t_{i,4m_i-2} \overset{a_{i,m_i}}{\longleftrightarrow} t_{i,4m_i-1} \overset{X_{i_{m_i}}}{\longrightarrow} t_{i,4m_i} \overset{X_{i_{m_i}}}{\longleftrightarrow} t_{i,4m_i+1} \overset{a_{i,m_i}}{\longleftrightarrow} t_{i,4m_i+2}$$

is a path from 0 to 1 in τ. Thus, there is an event $e \in \{X_{i_1},\dots,X_{i_{m_i}}\} \cup \{a_{i,1},\dots,a_{i,m_i}\}$ whose signature causes the state change from 0 to 1. This implies $sig(e) \neq \mathsf{nop}$. Assume, for a contradiction, that $sig(e) = \mathsf{nop}$ for

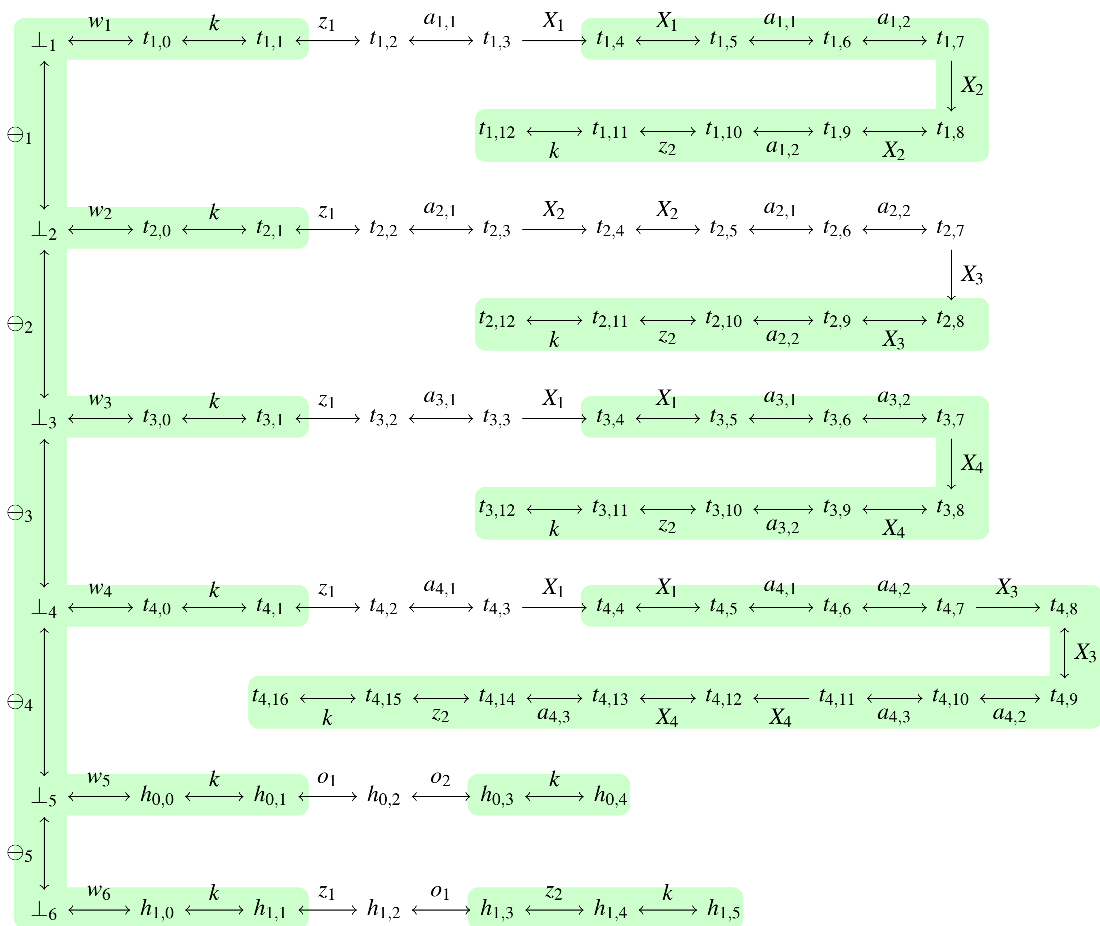

Figure 8: A full example of A^{τ}_{I}, where τ belongs to the types of Theorem 1.3 and I originates from Example 4. Green colored area: A sketch of the $\{\mathsf{nop},\mathsf{set},\mathsf{swap},\mathsf{used}\}$-region $R^k = (sup, sig)$, based on the HS $S = \{X_1, X_3\}$, that satisfies $sig(k) = \mathsf{used}$ and $sup(h_{0,2}) = 0$ and solves α.

all $e \in \{X_{i_1},\dots,X_{i_{m_i}}\}$. Let $\ell \in \{1,\dots,m_i\}$ be arbitrary but fixed. By $sig(X_\ell) = \mathsf{nop}$, we get $sup(t_{i,4\ell-1}) = sup(t_{i,4\ell}) = sup(t_{i,4\ell+1})$. Recall that $sup(s) \neq sup(s')$ implies $sig(e) = \mathsf{swap}$ for all $s \overset{e}{\longleftrightarrow} s' \in A^{\tau}_{I}$. Thus, if $sig(a_{i,\ell}) \neq \mathsf{swap}$, then $sup(t_{i,4\ell-2}) = sup(t_{i,4\ell-1}) = sup(t_{i,4\ell}) = sup(t_{i,4\ell+1}) = sup(t_{i,4\ell+2})$. Otherwise, if $sig(a_{i,\ell}) = \mathsf{swap}$, then $sup(t_{i,4\ell-2}) \neq sup(t_{i,4\ell-1}) = sup(t_{i,4\ell}) = sup(t_{i,4\ell+1}) \neq sup(t_{i,4\ell+2})$. Consequently, both cases imply $sup(t_{i,4\ell-2}) = sup(t_{i,4\ell+2})$. Since ℓ was arbitrary, this implies $sup(t_{i,2}) = sup(t_{i,4m_i+2})$, a contradiction. Hence, there is an event $e \in \{X_{i_1},\dots,X_{i_{m_i}}\}$ such that $sig(e) \neq \mathsf{nop}$. Since i was arbitrary, this is simultaneously true for all $T_1,\dots,T_m$. Moreover, since R respects the parameter, the cardinality of $S = \{X \in \mathfrak{U} \mid sig(X) \neq \mathsf{nop}\}$ is at most κ. Thus, S is a fitting hitting set of I.

The next lemma completes the proof of Theorem 1.3 and states that a sought HS of I implies a d-restricted admissible set of A^{τ}_{I}. Due to space restrictions, its proof can be found in [23].

Lemma 4. *Let τ be a type of net corresponding to Theorem 1.3. If $I = (\mathfrak{U}, M, \kappa)$ has a fitting HS, then A^{τ}_{I} has a d-restricted admissible set.*

3.4 The Proof of Theorem 1.4

Theorem 1.4: The Reduction In the following, we argue for $\tau = \{\mathsf{nop}, \mathsf{inp}, \mathsf{res}, \mathsf{swap}\}$. The hardness for $\tau = \{\mathsf{nop}, \mathsf{out}, \mathsf{set}, \mathsf{swap}\}$ then follows by symmetry. For a start, we define $d = \kappa + 4$. The TS A_I^τ has the following gadgets $H_0, \dots, H_4$ that provide the atom $\alpha = (k, h_{0,2})$:

$$H_0 = \bot_{m+1} \xrightarrow{w_{m+1}} h_{0,0} \xrightarrow{k} h_{0,1} \xrightarrow{o_1} h_{0,2} \xrightarrow{o_2} h_{0,3} \xrightarrow{k} h_{0,4}$$

$$H_1 = \bot_{m+2} \xrightarrow{w_{m+2}} h_{1,0} \xrightarrow{k} h_{1,1} \xrightarrow{z_1} h_{1,2} \xrightarrow{o_2} h_{1,3} \xrightarrow{k} h_{1,4}$$

$$H_2 = \bot_{m+3} \xrightarrow{w_{m+3}} h_{2,0} \xrightarrow{k} h_{2,1} \xrightarrow{z_2} h_{2,2} \xrightarrow{o_2} h_{2,3} \xrightarrow{k} h_{2,4}$$

$$H_3 = \bot_{m+4} \xrightarrow{w_{m+4}} h_{3,0} \xrightarrow{k} h_{3,1} \xrightarrow{z_1} h_{3,2} \xrightarrow{z_3} h_{3,3} \xrightarrow{z_2} h_{3,4} \xrightarrow{k} h_{3,5}$$

$$H_4 = \bot_{m+5} \xrightarrow{w_{m+5}} h_{4,0} \xrightarrow{k} h_{4,1} \xrightarrow{z_1} h_{4,2} \xrightarrow{z_4} h_{4,3} \xrightarrow{z_2} h_{4,4} \xrightarrow{k} h_{4,5}$$

Moreover, for every $i \in \{1, \dots, m\}$, the TS A_I^τ has the following gadget T_i that uses the elements of $M_i = \{X_{i_1}, \dots, X_{i_{m_i}}\}$ as events:

$$t_{i,0} \xrightarrow{k} t_{i,1} \xrightarrow{z_3} t_{i,2} \xrightarrow{X_{i_1}} \cdots \xrightarrow{X_{i_{m_i}}} t_{i,m_i+2} \xrightarrow{z_4} t_{i,m_i+3} \xrightarrow{k} t_{i,m_i+4}$$

The Joining of A_I^τ by Relevant Paths. Similar to the previous reductions, we essentially want to connect all gadgets by a simple directed path on which every event occurs exactly once. However, since we want to ensure that if α is τ-solvable then all (E)SSP atoms of A_I^τ are also τ-solvable (by d-restricted regions), this is not directly possible for the gadgets $T_1, \dots, T_m$. Instead, we complete the construction of A_I^τ through two further steps. Firstly, for all $i \in \{1, \dots, m\}$, we extend the gadget T_i to a (path-) gadget $G_i = \bot_i \rightsquigarrow T_i$ with starting state $\bot_i$. Secondly, we use the events $\ominus_1, \dots, \ominus_{m+4}$ and connect the gadgets $G_1, \dots, G_m$ and $H_0, \dots, H_4$ by $\bot_1 \xrightarrow{\ominus_1} \bot_2 \xrightarrow{\ominus_2} \dots \xrightarrow{\ominus_{m+4}} \bot_{m+5}$. The resulting TS is A_I^τ, and its initial state is $\bot_1$. Before we introduce the definition of G_i, in the following, we briefly outline which obstacles arise and, in order to overcome them, in which way they lead to G_i.

Let $i \in \{1, \dots, m\}$ and $\ell \in \{1, \dots, m_i\}$ be arbitrary but fixed. Similar to the approach of region $R_{i,\ell}^{X,2}$ of Theorem 1.1, which is sketched for $i = 3$ and $\ell = 2$ by Figure 6, our aim is to solve X_{i_ℓ} "gadget-wise". In particular, to solve (X_{i_ℓ}, s) for all predecessor states s of $t_{i,\ell+1}$ in G_i, that is, $\bot_i, \dots, t_{i,\ell}$, we want to construct a region $R = (sup, sig)$ such that as few events as possible are not mapped to nop. (Independent of A_I^τ's size, the region $R_{i,\ell}^{2,X}$ of Theorem 1.1 maps four events not to nop.) First of all, look at the following definition: $sup(\bot_1) = 0$; for all $e \in E(A_I^\tau)$, if $e = X_{i_\ell}$, then $sig(e) = \mathsf{inp}$; if e is X_{i_ℓ}'s direct predecessor, that is, $\xrightarrow{e} t_{i,\ell+1}$, then $sig(e) = \mathsf{swap}$; otherwise $sig(e) = \mathsf{nop}$. In Figure 9, the red colored area sketches this region for $X_{1_1} = X_1$ and its direct predecessor z_3; the green colored area sketches this region for $X_{3_2} = X_4$ and its direct predecessor X_1. Actually, R is always well defined if $X_{i_\ell} \in E(T_j)$ implies that X_{i_ℓ}'s direct predecessor $\xrightarrow{e} t_{i,\ell+1}$ also belongs to $E(T_j)$. This is not true if there is an occurrence of X_{i_ℓ} in a gadget T_j, say at $t_{j,\ell'}$, such that X_{i_ℓ}'s predecessor does not belong to T_j's event set. For example, consider in Figure 9 the event $X_{4_2} = X_3$ of T_4 that occurs as X_{2_2} in T_2. In T_4, X_3 is directly preceded by X_1, but X_1 does not occur in T_2. The following problem arises. Since $sig(X_{i_\ell}) = \mathsf{inp}$, there has to be an event e on the unambiguous path $\bot_1 \longrightarrow \dots \longrightarrow t_{j,\ell'}$ such that $sig(e) = \mathsf{swap}$. Otherwise, X_{i_ℓ}'s source $t_{i,\ell'}$ in T_j would not satisfy $sup(t_{i,\ell'}) = 1$. At first glance, a possible solution might be to implement an additional (unique) event y_j on the path $\bot_j \rightsquigarrow t_{j,0}$ for all $j \in \{1, \dots, m\}$ where X_{i_ℓ} belongs to $E(T_j)$

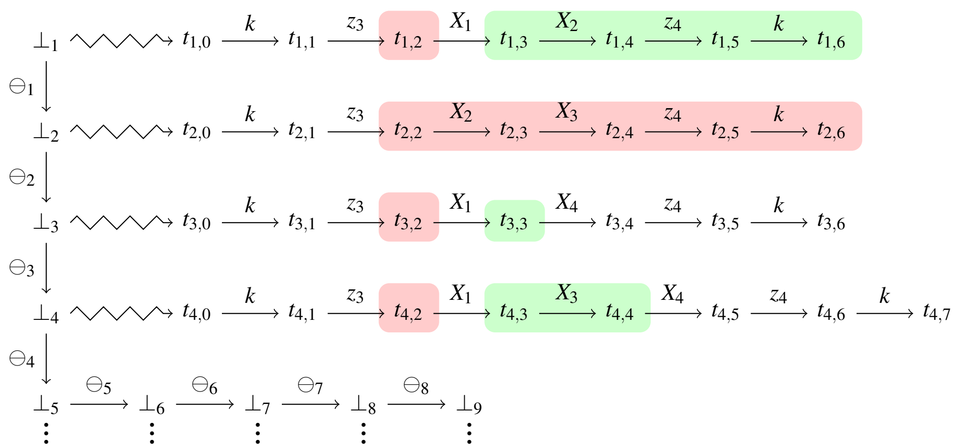

Figure 9: A snippet of A_I^τ ($\tau = \{\mathsf{nop}, \mathsf{inp}, \mathsf{res}, \mathsf{swap}\}$) built from Example 4 and showing the gadgets $T_1, \ldots, T_4$. Red colored area: the region $R = (sup, sig)$ where $sup(\bot_1) = 0$; $sig(X_1) = \mathsf{inp}$; $sig(z_3) = \mathsf{swap}$; $sig(e) = \mathsf{nop}$ for all $e \in E(A_I^\tau) \setminus \{z_3, X_1\}$. Green colored area: the region $R = (sup, sig)$ where $sup(\bot_1) = 0$; $sig(X_4) = \mathsf{inp}$; $sig(X_1) = \mathsf{swap}$; $sig(e) = \mathsf{nop}$ for all $e \in E(A_I^\tau) \setminus \{X_1, X_4\}$.

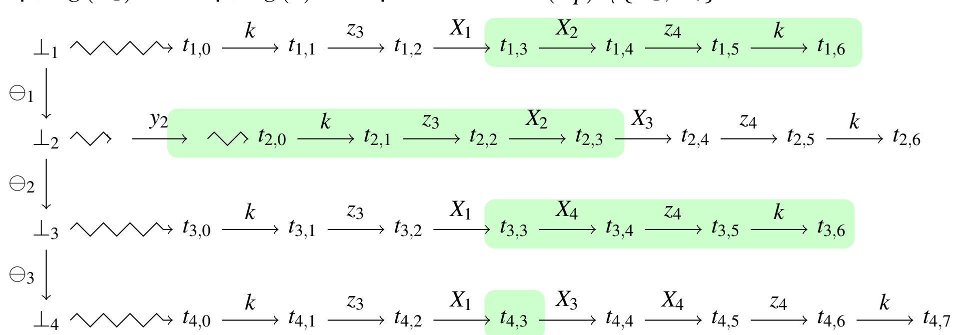

Figure 10: A sketch of the "first-glance" solution for A_I^τ ($\tau = \{\mathsf{nop}, \mathsf{inp}, \mathsf{res}, \mathsf{swap}\}$), where I corresponds to Example 4. Green colored area: the region $R = (sup, sig)$ where $sup(\bot_1) = 0$; $sig(X_3) = \mathsf{inp}$; $sig(X_1) = sig(y_2) = \mathsf{swap}$; $sig(e) = \mathsf{nop}$ for all $e \in E(A_I^\tau) \setminus \{X_1, X_3, y_2\}$.

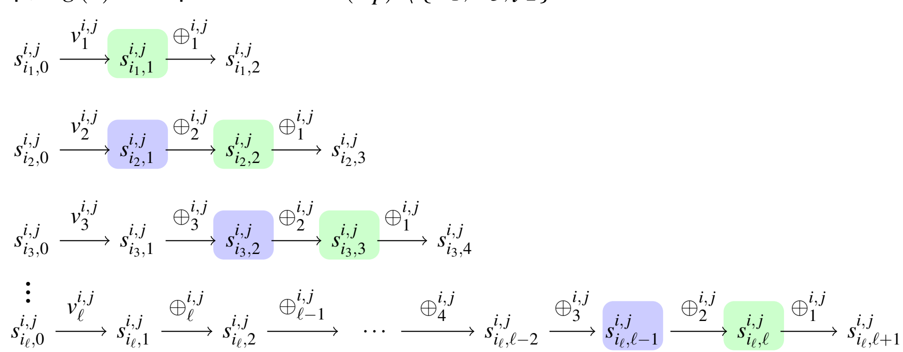

Figure 11: The pyramidal approach of the relevant paths ensures that $\oplus$-events are solvable by regions independent of the size of $(\mathfrak{U}, M, \kappa)$. Green colored area: a region $R = (sup, sig)$ solving $(\oplus_1^{i,j}, s)$ for all relevant $s \in S(A_I^\tau)$: $sup(\bot_1) = 0$; for all $e \in E(A_I^\tau)$, if $e = \oplus_1^{i,j}$, then $sig(e) = \mathsf{inp}$; if $e \in \{v_1^{i,j}, \oplus_2^{i,j}\}$, then $sig(e) = \mathsf{swap}$; otherwise $sig(e) = \mathsf{nop}$. Blue colored area: a corresponding region solving $\oplus_2^{i,j}$. These regions are independent from the positions of $G_{i_1}, \ldots, G_{i_\ell}$ in A_I^τ or P_{i_n} in G_{i_n}, where $n \in \{1, \ldots, \ell\}$.

but X_{i_ℓ}'s direct predecessor event does not. Then we would modify the region $R=(sup,sig)$ in a way, that $sig(y_j)=\mathsf{swap}$ for all relevant j. Figure 10 sketches the situation for y_2.

Unfortunately, for this construction and the sketched region, $|\{e\in E(A_I^\tau)\mid sig(e)\neq \mathsf{nop}\}|\geq n+2$ holds, where n is the number of gadgets in which X_{i_ℓ} occurs but its predecessor does not. Since X_{i_ℓ} could occur in numerous sets, in general, n depends on the size of M and does not necessarily respect the parameter d. Thus, this approach yields not a parameterized reduction. The next inelaborate solution to overcome this obstacle is to ensure that there is the same event, say y, on every path $\bot_j \rightsquigarrow t_{j,0}$ for all $j\in\{1,\dots,m\}\setminus\{i\}$ such that $X_{i_\ell}\in E(T_j)$ but X_{i_ℓ}'s predecessor is not in $E(T_j)$. However, one has to ensure that the already discussed difficulties are not transferred from X_{i_ℓ} to y. Our solution uses *relevant paths* to realize a pyramidal approach that is sketched by Figure 11. Instead of one single event y (whose role is played by $\oplus_1^{i,j}$ in Figure 11), this approach implements for every corresponding T_j a unique directed path.

Let $i\in\{1,\dots,m\}$ be arbitrary but fixed. We extend the gadget T_i to $G_i=\bot_i\xrightarrow{w_i}P_i\xrightarrow{u_i}T_i$ with starting state $\bot_i$ and events w_i,u_i that embrace the path P_i, to be defined next. To be able to refer uniformly to the events $X_{i_1},\dots,X_{i_{m_i}}$ and z_4, we define $e_1^i=X_{i_1},\dots,e_{m_i}^i=X_{i_{m_i}}$ and $e_{m_i+1}^i=z_4$. Let $j\in\{2,\dots,m_i+1\}$ be arbitrary but fixed and let $i_1<\dots<i_\ell\in\{1,\dots,m\}\setminus\{i\}$ be exactly the indices different from i such that for the gadgets $T_{i_1},\dots,T_{i_\ell}$ we have $e_j^i\in E(T_{i_n})$ and $e_{j-1}^i\notin E(T_{i_n})$, for all $n\in\{1,\dots,\ell\}$. For all $n\in\{1,\dots,\ell\}$, we say that e_j^i is relevant for G_{i_n} and

$$P_{i_n,n}^{i,j}=s_{i_n,0}^{i,j}\xrightarrow{v_n^{i,j}}s_{i_n,1}^{i,j}\xrightarrow{\oplus_n^{i,j}}s_{i_n,2}^{i,j}\xrightarrow{\oplus_{n-1}^{i,j}}\dots\xrightarrow{\oplus_1^{i,j}}s_{i_n,n+1}^{i,j}$$

is the relevant path of G_{i_n} that originates from e_j^i.

Example 5. *The event $e_3^1=z_4$ of T_1 of Figure 9 is preceded by $e_2^1=X_2$. While the event z_4 occurs in T_2,T_3 and T_4, the event X_2 occurs in T_2 but not in T_3 and not in T_4. Thus, e_3^1 is (only) relevant for $T_3=T_{i_1}$ and $T_4=T_{i_2}$, where $i_1=3$ and $i_2=4$. The corresponding relevant paths are*

$$P_{3,1}^{1,3}=s_{3,0}^{1,3}\xrightarrow{v_1^{1,3}}s_{3,1}^{1,3}\xrightarrow{\oplus_1^{1,3}}s_{3,2}^{1,3}\text{ and }P_{4,2}^{1,3}=s_{4,0}^{1,3}\xrightarrow{v_2^{1,3}}s_{4,1}^{1,3}\xrightarrow{\oplus_2^{1,3}}s_{4,2}^{1,3}\xrightarrow{\oplus_1^{1,3}}s_{4,3}^{1,3}.$$

Equipped with these definitions, we are prepared to define the gadget G_i. If there are no relevant events for G_i, then $G_i=\bot_i\xrightarrow{w_i}q_i\xrightarrow{u_i}T_i$. In particular, $P_i=q_i$. Otherwise, let $e_{j_1}^{i_1},\dots,e_{j_n}^{i_n}$ be the events that are relevant for G_i where $i_1\leq i_2\leq\dots\leq i_n$ and $j_1\leq j_2\leq\dots\leq j_n$. Let $P_{i,\ell_1}^{i_1,j_1},P_{i,\ell_2}^{i_2,j_2},\dots,P_{i,\ell_n}^{i_n,j_n}$ be the relevant paths of G_i that origin from $e_{j_1}^{i_1},\dots,e_{j_n}^{i_n}$, respectively. The path P_i then originates from G_i's relevant paths:

$$G_i=\bot_i\xrightarrow{w_i}P_{i,\ell_1}^{i_1,j_1}\xrightarrow{c_1^i}P_{i,\ell_2}^{i_2,j_2}\xrightarrow{c_2^i}\dots\xrightarrow{c_n^i}P_{i,\ell_n}^{i_n,j_n}\xrightarrow{u_i}T_i$$

See [23] for a full example.

Theorem 1.4: The τ-Solvability of α Implies a Hitting Set. Let $R=(sup,sig)$ be a d-restricted τ-region of A_I^τ that solves α. Since R solves α, one easily finds that $sig(k)=\mathsf{inp}$ and $sup(h_{0,2})=0$. By $sig(k)=\mathsf{inp}$, we have $sup(h_{0,3})=1$; and $sup(h_{0,2})=0$ implies $sig(o_2)=\mathsf{swap}$. Moreover, by $sig(k)=\mathsf{inp}$ and $sig(o_2)=\mathsf{swap}$, we obtain that $sup(h_{1,1})=sup(h_{1,2})=sup(h_{2,1})=sup(h_{2,2})=0$. This implies $sig(z_1),sig(z_2)\in\{\mathsf{nop},\mathsf{res}\}$. By $sig(k)=\mathsf{inp}$ and $sig(z_1),sig(z_2)\in\{\mathsf{nop},\mathsf{res}\}$, we get $sup(h_{3,2})=sup(h_{4,2})=0$ and $sup(h_{3,3})=sup(h_{4,3})=1$. This implies $sig(z_3)=sig(z_4)=\mathsf{swap}$. Since $d=\kappa+4$ and R is d-restricted, there are at most κ events left whose signature is different from nop. Let $i\in\{1,\dots,m\}$ be arbitrary but fixed. By $sig(k)=\mathsf{inp}$, we get $sup(t_{i,1})=0$ and $sup(t_{i,m_i+3})=1$. Moreover, by $sig(z_3)=sig(z_4)=\mathsf{swap}$, we get $sup(t_{i,2})=1$ and $sup(t_{i,m_i+2})=0$. Thus, there is an event $X\in E(T_i)$ such that

$sig(X) \in \{\mathsf{inp}, \mathsf{res}, \mathsf{swap}\}$. Since i was arbitrary and R is d-restricted, the set $S = \{X \in \mathfrak{U} \mid sig(X) \neq \mathsf{nop}\}$ is a sought-for HS of I.

Theorem 1.4: A Hitting Set Implies the τ-Solvability of A_I^τ. We argue for the τ-solvability of k, implying the τ-solvability of α. The following d-restricted τ-region $R = (sup, sig)$ solves α and solves (k,s) for all relevant $s \in \bigcup_{i=1}^m S(H_i) \setminus \{\bot_{m+1}, \dots, \bot_{m+5}\}$, too: $sup(\bot_1) = 1$; for all $e \in E(A_I^\tau)$, if $e = k$, then $sig(e) = \mathsf{inp}$; if $e \in \{o_2, z_3, z_4\}$, then $sig(e) = \mathsf{swap}$; if $e \in S$, then $sig(e) = \mathsf{res}$; otherwise, $sig(e) = \mathsf{nop}$.

Let $i \in \{1, \dots, m\}$ be arbitrary but fixed. The following region $R = (sup, sig)$ solves (k,s) for all relevant $s \in S(G_i)$: If $i = 1$, then $sup(\bot_1) = 0$, otherwise $sup(\bot_1) = 1$; for all $e \in E(A_I^\tau)$, if $e \in \{k, \ominus_{i-1}\}$, then $sig(k) = \mathsf{inp}$; if $e \in \{\ominus_i, o_1, z_1, z_2, z_4\}$, then $sig(e) = \mathsf{swap}$; if $e = z_3$, then $sig(e) = \mathsf{res}$; otherwise, $sig(e) = \mathsf{nop}$. It is easy to see that, for any $s \in \{\bot_{m+1}, \dots, \bot_{m+5}\}$, this region can be modified to a d-restricted region that solves (k,s).

Let $i \in \{1, \dots, m_i\}$ be arbitrary but fixed. The separability of $X_{i_1}, \dots, X_{i_{m_i}}, z_4$ in G_i has already been sketched in the explanation of the relevant paths. Clearly, these events are separable in the gadgets in which they do not occur. Also the helper events of the relevant paths are separable. We omit the proofs for the sake of readability.

4 Conclusion

In this paper, we investigate the parameterized complexity of DRτS parameterized by d and show $W[2]$-hardness for a range of Boolean types. As a result, d is ruled out for fpt-approaches for the considered types of nets. As future work, it remains to classify DRτS exactly in the W-hierarchy. Moreover, one may look for other more promising parameters: If $N = (P, T, M_0, f)$ is a Boolean net, $p \in P$ and if the *occupation number* o_p of p is defined by $o_p = |\{M \in RS(N) \mid M(p) = 1\}|$ then the *occupation number* o_N of N is defined by $o_N = \max\{o_p \mid p \in P\}$. If $\mathscr{R}$ is a τ-admissible set (of a TS A) and $R \in \mathscr{R}$, then the support of R determines the number of markings of $N_A^{\mathscr{R}}$ that occupy R, that, is, $o_R = |\{s \in S(A) \mid sup(s) = 1\}|$. Thus, searching for a τ-net where $o_N \leq n$, $n \in \mathbb{N}$, corresponds to searching for a τ-admissible set $\mathscr{R}$ such that $|\{s \in S(A) \mid sup(s) = 1\}| \leq n$ for all $R \in \mathscr{R}$. As a result, for each (E)SSP atom α there are at most $\mathscr{O}(\binom{|S|}{o_N})$ fitting supports for τ-regions solving α. Thus, the corresponding problem *o_N-restricted τ-synthesis* parameterized by o_N is in XP if, in a certain sense, τ-regions are fully determined by a given support sup.

References

[1] Wil M. P. van der Aalst (2011): *Process Mining - Discovery, Conformance and Enhancement of Business Processes*. Springer, doi:10.1007/978-3-642-19345-3.

[2] Eric Badouel, Luca Bernardinello & Philippe Darondeau (1997): *The Synthesis Problem for Elementary Net Systems is NP-Complete*. Theor. Comput. Sci. 186(1-2), pp. 107–134, doi:10.1016/S0304-3975(96)00219-8.

[3] Eric Badouel, Luca Bernardinello & Philippe Darondeau (2015): *Petri Net Synthesis*. Texts in Theoretical Computer Science. An EATCS Series, Springer, doi:10.1007/978-3-662-47967-4.

[4] Eric Badouel, Benoît Caillaud & Philippe Darondeau (2002): *Distributing Finite Automata Through Petri Net Synthesis*. Formal Asp. Comput. 13(6), pp. 447–470, doi:10.1007/s001650200022.

[5] Eric Badouel & Philippe Darondeau (1995): *Trace Nets and Process Automata*. Acta Inf. 32(7), pp. 647–679, doi:10.1007/BF01186645.

[6] Eike Best (1986): *Structure Theory of Petri Nets: the Free Choice Hiatus*. In: *Advances in Petri Nets*, *Lecture Notes in Computer Science* 254, Springer, pp. 168–205, doi:10.1007/BFb0046840.

[7] Eike Best, Raymond R. Devillers & Uli Schlachter (2018): *Bounded choice-free Petri net synthesis: algorithmic issues*. *Acta Inf.* 55(7), pp. 575–611, doi:10.1007/s00236-017-0310-9.

[8] Jordi Cortadella, Michael Kishinevsky, Alex Kondratyev, Luciano Lavagno & Alexandre Yakovlev (1997): *A region-based theory for state assignment in speed-independent circuits*. *IEEE Trans. on CAD of Integrated Circuits and Systems* 16(8), pp. 793–812, doi:10.1109/43.644602.

[9] Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk & Saket Saurabh (2015): *Parameterized Algorithms*. Springer, doi:10.1007/978-3-319-21275-3.

[10] Raymond R. Devillers, Evgeny Erofeev & Thomas Hujsa (2019): *Synthesis of Weighted Marked Graphs from Circular Labelled Transition Systems*. In Wil M. P. van der Aalst, Robin Bergenthum & Josep Carmona, editors: *ATAED@Petri Nets/ACSD 2019, Aachen, Germany, June 25, 2019*, *CEUR Workshop Proceedings* 2371, CEUR-WS.org, pp. 6–22. Available at `http://ceur-ws.org/Vol-2371/ATAED2019-6-22.pdf`.

[11] Raymond R. Devillers & Thomas Hujsa (2019): *Analysis and Synthesis of Weighted Marked Graph Petri Nets: Exact and Approximate Methods*. *Fundam. Inform.* 169(1-2), pp. 1–30, doi:10.3233/FI-2019-1837.

[12] Javier Esparza & Mogens Nielsen (1994): *Decidability Issues for Petri Nets - a survey*. *Bulletin of the EATCS* 52, pp. 244–262.

[13] Lawrence E. Holloway, Bruce H. Krogh & Alessandro Giua (1997): *A Survey of Petri Net Methods for Controlled Discrete Event Systems*. *Discrete Event Dynamic Systems* 7(2), pp. 151–190, doi:10.1023/A:1008271916548.

[14] Marta Pietkiewicz-Koutny (1997): *Transition Systems of Elementary Net Systems with Inhibitor Arcs*. In: *ICATPN*, *Lecture Notes in Computer Science* 1248, Springer, pp. 310–327, doi:10.1007/3-540-63139-9_43.

[15] Wolfgang Reisig (1992): *A primer in Petri net design*. Springer Compass International, Springer, doi:10.1007/978-3-642-75329-9.

[16] Uli Schlachter (2016): *Bounded Petri Net Synthesis from Modal Transition Systems is Undecidable*. In Josée Desharnais & Radha Jagadeesan, editors: *27th International Conference on Concurrency Theory, CONCUR 2016, August 23-26, 2016, Québec City, Canada*, *LIPIcs* 59, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, pp. 15:1–15:14, doi:10.4230/LIPIcs.CONCUR.2016.15.

[17] Vincent Schmitt (1996): *Flip-Flop Nets*. In: *STACS*, *Lecture Notes in Computer Science* 1046, Springer, pp. 517–528, doi:10.1007/3-540-60922-9_42.

[18] Robert Endre Tarjan (1977): *Finding optimum branchings*. *Networks* 7(1), pp. 25–35, doi:10.1002/net.3230070103.

[19] Ronny Tredup (2019): *The Complexity of Synthesizing nop-Equipped Boolean Nets from g-Bounded Inputs (Technical Report)*. *CoRR* abs/1911.05834. Available at `http://arxiv.org/abs/1911.05834`.

[20] Ronny Tredup (2019): *Hardness Results for the Synthesis of b-bounded Petri Nets*. In Susanna Donatelli & Stefan Haar, editors: *Application and Theory of Petri Nets and Concurrency - 40th International Conference, PETRI NETS 2019, Aachen, Germany, June 23-28, 2019, Proceedings*, *Lecture Notes in Computer Science* 11522, Springer, pp. 127–147, doi:10.1007/978-3-030-21571-2_9.

[21] Ronny Tredup (2019): *Synthesis of Structurally Restricted b-bounded Petri Nets: Complexity Results*. In Emmanuel Filiot, Raphaël M. Jungers & Igor Potapov, editors: *Reachability Problems - 13th International Conference, RP 2019, Brussels, Belgium, September 11-13, 2019, Proceedings*, *Lecture Notes in Computer Science* 11674, Springer, pp. 202–217, doi:10.1007/978-3-030-30806-3_16.

[22] Ronny Tredup & Evgeny Erofeev (2020): *On the Parameterized Complexity of d-Restricted Boolean Net Synthesis*. In: *Theory and Applications of Models of Computation*, *Theoretical Computer Science and General Issues* 12337, doi:10.1007/978-3-030-59267-7.

[23] Ronny Tredup & Evgeny Erofeev (2020): *On the Parameterized Complexity of Synthesizing Boolean Petri Nets With Restricted Dependency (Technical Report)*. *CoRR* abs/2007.12372. Available at `https://arxiv.org/abs/2007.12372`.

[24] Ronny Tredup & Christian Rosenke (2018): *Narrowing down the Hardness Barrier of Synthesizing Elementary Net Systems*. In: *CONCUR*, *LIPIcs* 118, Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, pp. 16:1–16:15, doi:10.4230/LIPIcs.CONCUR.2018.16.

[25] Ronny Tredup, Christian Rosenke & Karsten Wolf (2018): *Elementary Net Synthesis Remains NP-Complete Even for Extremely Simple Inputs*. In: *Petri Nets*, *Lecture Notes in Computer Science* 10877, Springer, pp. 40–59, doi:10.1007/978-3-319-91268-4_3.

[26] Harro Wimmel (2019): *Synthesis of Reduced Asymmetric Choice Petri Nets*. *CoRR* abs/1911.09133. Available at `http://arxiv.org/abs/1911.09133`.

[27] Harro Wimmel (2020): *Presynthesis of bounded choice-free or fork-attribution nets*. *Inf. Comput.* 271, p. 104482, doi:10.1016/j.ic.2019.104482.