LIPIcs Leibniz International Proceedings in Informatics

Modelchecking counting properties of 1-safe nets with buffers in paraPSPACE

M. Praveen and Kamal Lodaya
The Institute of Mathematical Sciences, Chennai, India

ABSTRACT. We consider concurrent systems that can be modelled as 1-safe Petri nets communicating through a fixed set of buffers (modelled as unbounded places). We identify a parameter K, which we call "benefit depth", formed from the communication graph between the buffers. We show that for our system model, the coverability and boundedness problems can be solved in polynomial space assuming K to be a fixed parameter, that is, the space requirement is $f(K)p(n)$, where f is an exponential function and p is a polynomial in the size of the input. We then obtain similar complexity bounds for modelchecking a logic based on such counting properties. This means that systems that have sparse communication patterns can be analyzed more efficiently than using previously known algorithms for general Petri nets.

1 Introduction

Many theoretical models exist for concurrent, infinite-state systems. Petri nets [19], process rewrite systems [4], lossy channel systems (LCS) [5] and networks of pushdown systems [1] are some of them. The power to express properties of the original system in sufficient detail and existence of efficient algorithms for analysis are often conflicting goals in these models. Reachability in LCS is non-primitive recursive [22] and reachability for Petri nets is decidable but with no known upper bound [18, 15].

More structure is sometimes imposed on the models to handle these conflicting goals. Communicating automata with buffers [3] is one such model. In this paper we consider a small generalization where 1-safe Petri nets (which we call components) communicate via buffers. Thus we have a system model which allows both asynchronous and synchronous communication, since 1-safe Petri nets can model the latter.

The diagram shown in Fig. 1 illustrates the kind of systems we are interested in. The boxes labelled as line 1, line 2 etc. can be thought of as assembly lines represented by 1-safe Petri nets, drawing raw materials from buffers ib_1, ib_2 etc. Output of these assembly lines are deposited into buffers ob_1, ob_2 etc. Boxes labelled master line 1 and master line 2 can be thought of as master assembly lines that use output of earlier assembly lines as their input. They deposit their output in buffers pr_1 and pr_2 respectively. We are concerned with verifying properties like $\exists c : pr_1 \leq c$ *in all reachable configurations* (boundedness) or $ob_1 + ob_2 \geq 100$ *in some reachable configuration* (coverability). For instance, the latter property might show that the two buffers are dealing with enough throughput. Karp and Miller examined these properties in the context of Petri nets [14] and Lipton and Rackoff showed them to be EXPSPACE-complete [17, 20].

As Esparza notes in his survey article [10], verification of a "logic" based on such properties, for instance LTL or CTL extended with counting properties, quickly becomes undecidable. Modalities of the form $\mathbf{EF}(M \geq M_c)$ (where M, M_c are markings) can be handled

Foundations of Software Technology and Theoretical Computer Science (Kanpur) 2009.
Editors: Ravi Kannan and K. Narayan Kumar; pp 347–358
Leibniz International Proceedings in Informatics (LIPIcs), Schloss Dagstuhl - Leibniz-Zentrum für Informatik, Germany.
Digital Object Identifier: 10.4230/LIPIcs.FSTTCS.2009.2331

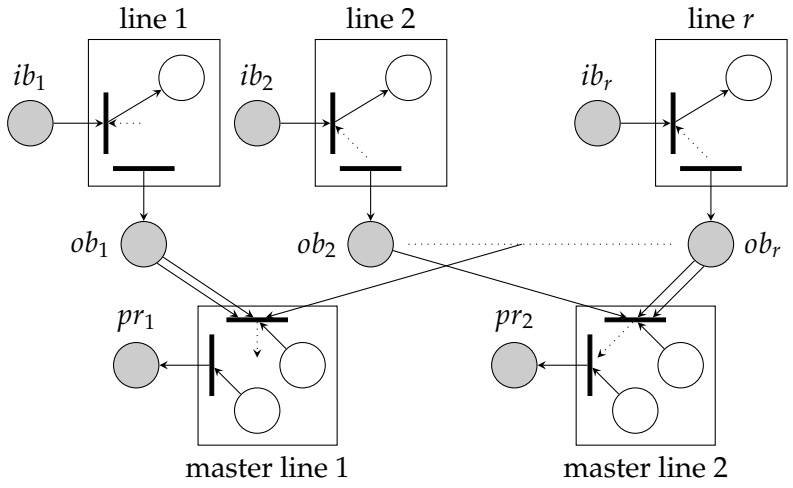

Figure 1: Illustration of communicating automata with buffers

without getting into undecidability [24]. However, a "usual" definition of a logic based on these modalities can express reachability, as in Howell, Rosier and Yen's logic [13] and in Yen's logic [24] (as was recently shown by Atig and Habermehl [2]). So we are left with positive Boolean combinations of formulae of the form $\mathbf{EF}(M \geq M_c)$ [24] for which modelchecking is EXPSPACE-hard. Rosier and Yen analyzed boundedness [21] using what we today call parameterized complexity [9] to show that the space requirement is exponential in the number of unbounded places and polynomial in the number of bounded places. If we give up counting properties, Habermehl shows that the full linear time μ-calculus can be reduced to the problem of repeated control state reachability [12] and is PSPACE-complete in the size of the formula and EXPSPACE-complete in the size of the model.

An EXPSPACE lower bound in the size of the model is not very encouraging for potential verifiers. Our first contribution is the identification of a parameter K, which we call benefit depth. A buffer p_1 can benefit by another buffer p_2 if there is a sequence of transitions that decrease tokens in p_2 and increase tokens in p_1. Benefit depth is the maximum number of buffers benefited by any one buffer. It seems reasonable that, in a sparsely communicating system, benefit depth can be low.

We show that boundedness and coverability in our models, when parameterized by benefit depth, are solvable in paraPSPACE [11]. That is, the space requirement is of the form $\mathcal{O}(f(K)p(n))$, where f is an exponential function of benefit depth and p is some polynomial of the size of the model and the marking to be covered. For constant benefit depth, boundedness and coverability can be solved in PSPACE. Thus, our results are refinements of Rosier and Yen's [21], improving them if benefit depth is less than the number of buffers (as happens in sparsely communicating systems).

As our final contribution, we define a logic which can express counting properties such as coverability *and* show that it can be modelchecked on Petri nets in paraPSPACE.

The full version of this paper may be consulted at http://www.imsc.res.in/%7Epraveen/ for detailed proofs. This conference version attempts a more intuitive treatment without

compromising precision.

2 Problem definitions

Let $\mathbb{Z}$ be the set of integers and $\mathbb{N}$ the set of natural numbers. A Petri net is a 4-tuple $N = (P, T, Pre, Post)$ where P is a set of places, T is a set of transitions and Pre and $Post$ are the incidence functions: $Pre : P \times T \to [0 \dots W]$ (arcs going from places to transitions), $Post : P \times T \to [0 \dots W]$ (arcs going from transitions to places), where $W \geq 1$.

DEFINITION 1. *Given a place p, the set of places $Ben(p) \subseteq P$ and the set of transitions $T_{ben}(p) \subseteq T$ benefited by p are those connected to p by a sequence of arcs with weight ≥ 1. Formally they are the smallest sets satisfying:*

1. *$p \in Ben(p)$.*
2. *If some $p' \in Ben(p)$ and there is a transition t with $Pre(p', t) \geq 1$, then $t \in T_{ben}(p)$.*
3. *If some transition $t \in T_{ben}(p)$ and there is a place p'' such that $Post(p'', t) \geq 1$, then $p'' \in Ben(p)$.*

$Ind(p) = P \setminus Ben(p)$ and $T_{ind}(p) = T \setminus T_{ben}(p)$ are the places and transitions not benefiting from p.

We call a function $M : P \to \mathbb{Z}$ a vector. For two vectors M_1 and M_2, we say M_1 covers M_2 (written $M_1 \geq M_2$) if for every place p, $M_1(p) \geq M_2(p)$. $M_1 > M_2$ means that M_1 covers M_2 but they are not the same.

If the range of the vector is $\mathbb{N}$, it is called a marking. At a marking M, a place p is said to have $M(p)$ tokens. A pair (N, M_0) consisting of a Petri net N and an initial marking M_0 is called a **system**. We assume a net is presented as two matrices for Pre and $Post$. In the rest of this paper, we will assume that a Petri net N has m places, n transitions and that W is the maximum of the range of Pre and $Post$. We define the size of the net to be $2mn \log W$ bits. The system has size $2mn \log W + \log |M_0|$ bits.

A transition t may be taken as a step at the vector M yielding a new vector M' given by the equation $M'(p) = M(p) - Pre(p, t) + Post(p, t)$ for all $p \in P$. The transition t is said to be fired at M if, in addition, t is enabled at M, that is, for all $p \in P$, $M(p) \geq Pre(p, t)$. Thus firing a transition leads from a marking to another marking, while stepping is a more general notion leading from a vector to a vector.

A finite transition sequence $\sigma = t_1 t_2 \dots t_r$ is a walk from an initial vector M_0 to a vector M_r if there exist intermediate vectors $M_1, M_2, \dots, M_r$ such that for all i with $1 \leq i \leq r$, we have a step from M_{i-1} to M_i using the transition t_i. We write $M_0 \xrightarrow{\sigma} M_r$. σ is a firing sequence enabled at some initial marking M_0 if the transitions are enabled at the intermediate vectors, so that $M_1, M_2, \dots, M_r$ are all markings. We write $M_0 \stackrel{\sigma}{\Longrightarrow} M_r$ and say that the marking M_r is reachable from M_0. $\mathcal{R}(N, M_0)$ is the set of markings reachable from M_0. A place is said to be c-bounded, $c \in \mathbb{N}$, in the system (N, M_0), if for all its reachable markings M, $M(p)$ is in $\{0, \dots, c\}$. The system is c-bounded if all its places are. A 1-bounded system is commonly called a 1-Safe net.

DEFINITION 2. *[Reachability, coverability, boundedness] Given a system (N, M_0) and a marking M as input data, the reachability problem is to decide if the marking M is in $\mathcal{R}(N, M_0)$; the coverability problem is to decide if there is an M' in $\mathcal{R}(N, M_0)$ such that M' covers M.*

Given a system (N, M_0), the boundedness problem is to decide if there is some $c \in \mathbb{N}$ such that the system is c-bounded.

Given a c-bounded system, the reachability and coverability problems are known to be PSPACE-complete [6]. For systems in general, which can be unbounded, Lipton showed that all three problems are EXPSPACE-hard [17]. Rackoff showed that boundedness and coverability are in EXPSPACE[20]. Reachability has been shown to be decidable [18, 15], obtaining an upper bound is a famous open problem.

2.1 A logic of properties

Inspired by Yen [24], we now formulate a logic of properties such that its model checking can be reduced to coverability (κ) and boundedness (β) problems, but is designed to avoid expressing reachability. In particular, a κ formula of the form $\tau \leq c,\ c \in \mathbb{N}$, is *not* provided and the κ and ϕ formulas are *not* closed under negation.

$$\begin{aligned}
\tau &::= p,\ p \in P \mid \tau_1 + \tau_2 \mid c\tau,\ c \in \mathbb{N} \\
\kappa &::= \tau \geq c,\ c \in \mathbb{N} \mid \kappa_1 \wedge \kappa_2 \mid \kappa_1 \vee \kappa_2 \mid \mathbf{EF}\kappa \\
\beta &::= \{\tau_1, \ldots, \tau_r\} < \omega \mid \neg\beta \mid \beta_1 \vee \beta_2 \\
\phi &::= \beta \mid \kappa \mid \phi_1 \wedge \phi_2 \mid \phi_1 \vee \phi_2
\end{aligned}$$

The satisfaction of a formula ϕ by a system (N, M_0) (denoted as $N, M_0 \models \phi$) is defined below. The boolean operators work as usual. Note that every term (of type τ) gives a function $L_\tau : P \to \mathbb{N}$ such that τ is syntactically equivalent to $\sum_{p \in P} L_\tau(p)p$.

- $N, M_0 \models \tau \geq c$ if $\sum_{p \in P} L_\tau(p) M_0(p) \geq c$.
- $N, M_0 \models \mathbf{EF}\kappa$ if $\exists M \in \mathcal{R}(N, M_0)$ such that $N, M \models \kappa$.
- $N, M_0 \models \{\tau_1, \ldots \tau_r\} < \omega$ if $\exists c \in \mathbb{N} : \forall M \in \mathcal{R}(N, M_0)\ \exists j \in \{1, \ldots, r\}$ such that $\sum_{p \in P} L_{\tau_j}(p) M(p) \leq c$.

We use $\{\tau_1, \ldots, \tau_r\} = \omega$ as an abbreviation for $\neg(\{\tau_1, \ldots, \tau_r\} < \omega)$.

The formula $\{p_1, \ldots, p_r\} < \omega$ says that the given set of places is bounded according to Valk and Vidal-Naquet [23, Section 4.1]. On the other hand, $\{p_1 + \cdots + p_r\} < \omega$ says that the same set of places is uniformly bounded according to the same authors [23].*

2.2 System model

Though our results work for any Petri net, we work with the model defined below to emphasize the fact that our problem formulation strictly generalizes reachability for 1-bounded systems. The model of concurrent systems we consider in this paper consists of some 1-safe nets, called components, which can add or remove tokens to/from a set of unbounded places that we refer to as buffers.

*We thank an anonymous FSTTCS referee for pointing out this subtlety. Following their suggestion, we have slightly extended our logic beyond the submitted version to cover both kinds of boundedness.

DEFINITION 3. *A **net communicating with buffers** (we just use the word "net" below) is a Petri net $N = (C, B, T, Pre, Post)$ where the set of places $P = C \cup B$ is partitioned into a set of buffers B and component places $C = P \setminus B$, such that all places in C remain 1-bounded (regardless of the number of tokens in the buffers in an initial marking).*

In the rest of the paper, we will assume that $|C| = a$, $|B| = b$ and that $a + b = m$, where m is the total number of places. In our model, the components do not contribute to exponential space complexity. Our results can be generalized to the case where the components are declared to be c-bounded (for a constant c) rather than 1-bounded.

DEFINITION 4. *The **benefit depth** of a net is defined as $K = max\{|Ben(p) \cap B| - 1 \mid p \in B\}$.*

Benefit depth depends only on the communication pattern among buffers, even though the communication link may involve some component places. It can be computed efficiently (in NLOGSPACE).

The communication graph of the system of Fig. 1 is shown in Fig. 2. Irrespective of the number of assembly lines, benefit depth is 3 since only ob_i, pr_1 and pr_2 can benefit by decreasing tokens from ib_i. If there are interdependencies among the assembly lines, such

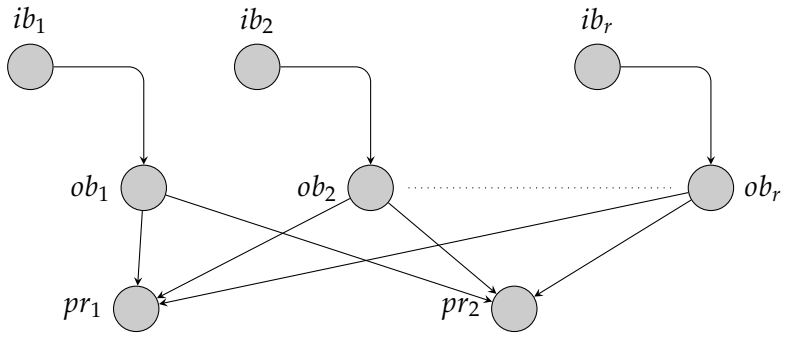

Figure 2: Communication graph of buffers of the system in Fig. 1

as a byproduct of one being the raw material of another (not shown in the figure), then benefit depth will increase. The more such dependencies (i.e., more dense the communication graph among the buffers is), the higher will be the benefit depth. Intuitively, the number of tokens in a place in $Ben(p)$ can be increased by decreasing some tokens in p through a sequence of transitions in $T_{ben}(p)$. Only those transitions use the extra tokens from p.

Our earlier definitions are modified to be well-behaved on the components. A vector will now be given by a pair of functions $C \to \{0, 1\}$ and $B \to \mathbb{Z}$; it is a marking if the second function has range $\mathbb{N}$. Walks and firing sequences will now be defined with these kinds of intermediate vectors and markings.

3 Benefit depth and coverability

Let $Q \subseteq P$ be a subset of places. For this paper we will need the inbetween notion (due to Rackoff) of σ being a Q-**run** where for the vectors $M_i, 0 \leq i < r$, $M_i(p) \geq Pre(p, t_{i+1})$ for every place p in Q. Thus a walk is a $\emptyset$-run and a firing sequence is a P-run. For two vectors

M_1 and M_2, we say $M_1 \geq_Q M_2$ if for every $p \in Q, M_1(p) \geq M_2(p)$ and $M_1(p) = M_2(p)$ for every $p \in C$. A walk σ from M_1 is said to Q-**cover** a marking M_{cov} if it is a Q-run and the final vector M_2 obtained by walking σ at M_1 satisfies $M_2 \geq_Q M_{cov}$. We say σ **covers** a marking if σ P-covers it.

We will fix for this section M_{cov} as the marking to be covered. For the purpose of complexity analysis, we will denote the maximum of the range of M_{cov} by R.

DEFINITION 5. *A Q-**covering** run is a Q-run that Q-covers M_{cov}. Let $Q_0 \subseteq Q$. A Q-run from M_0 to M_r is said to be c-**bounded for** Q_0, $c \in \mathbb{N}$, if for all intermediate vectors $M_i, 0 \leq i < r$, $M_i(p)$ is in $\{0, \ldots, c\}$ for every place p in Q_0.*

DEFINITION 6. *[20, Rackoff] Let $C \subseteq Q \subseteq P$. Define $lencov(Q, M, M_{cov})$ to be the length of the shortest Q-covering run from the vector M. If there is no such sequence, define $lencov(Q, M, M_{cov})$ to be 0. For $0 \leq i \leq b$, $\ell(i, M_{cov})$ is defined to be $max\{lencov(Q, M, M_{cov}) \mid M$ a vector, $C \subseteq Q \subseteq P$ and $|Q \setminus C| = i\}$. In this section we abbreviate $\ell(i, M_{cov})$ to $\ell(i)$. In section 5 we will abbreviate $\ell(b, M)$ to $\ell'(M)$.*

DEFINITION 7. *Let $C \subseteq Q \subseteq P$ and $p \in B$ be a buffer. Define $covind^p(Q, M, M_{cov})$ to be the length of the shortest Q-covering run in $T_{ben}(p)^*$ from the vector M. If there is no such sequence, define $covind^p(Q, M, M_{cov})$ to be 0. Let $\ell_1(i) = max\{covind^p(Q, M, M_{cov}) \mid M$ a vector, p a buffer, $|Q \cap Ben(p) \cap B| = i\}$.*

Our strategy is to segregate covering sequences into two parts, the first made of transitions in $T_{ind}(p)$ and the second one made of transitions in $T_{ben}(p)$. We need the following technical lemma, which is a generalization of the exchange lemma [7, Lemma 2.14] to Petri nets with weighted arcs.

LEMMA 8. *Let p be a place, transitions $t_{ben} \in T_{ben}(p)$ and $t_{ind} \in T_{ind}(p)$. Let $Q \subseteq P$ be some subset of places. If $t_{ben}t_{ind}$ is a Q-run from some vector M, then so is $t_{ind}t_{ben}$.*

LEMMA 9. *If $K \leq i < b$, then $\ell(i+1) \leq (W\ell_1(K) + R)^{i+1}2^a + \ell(i) + \ell_1(K)$.*

PROOF. Suppose σ is a Q_{i+1}-covering run from some vector M, with $Q_{i+1} \subseteq P$ and $|Q_{i+1} \cap B| = i + 1$. If some buffer $p \in Q_{i+1}$ has more than $W\ell_1(K) + R$ tokens at some intermediate marking M', rest of the sequence can be replaced by a Q_i-covering run σ_2' of length at most $\ell(i)$, where $Q_i = Q_{i+1} \setminus \{p\}$. Now, apply Lemma 8 repeatedly to rearrange σ_2' into $\tau_1\tau_2$, with $\tau_1 \in T_{ind}(p)^*$ and $\tau_2 \in T_{ben}(p)^*$ (see Fig. 3). Since τ_2 is a covering sequence made of transitions in $T_{ben}(p)$, it can be replaced by another one of length at most $\ell_1(K)$. ∎

The bound on $\ell(i+1)$ given by Rackoff in [20] is similar to the one in Lemma 9 but uses $\ell(i)$ in place of $\ell_1(K)$. Since $\ell_1(K)$ can be much smaller than $\ell(i)$, the bound in Lemma 9 is better. This is the fact that enables us to restrict exponential space complexity to K. The following lemma gives a recurrence relation for length of covering sequences made of transitions in $T_{ben}(p)$.

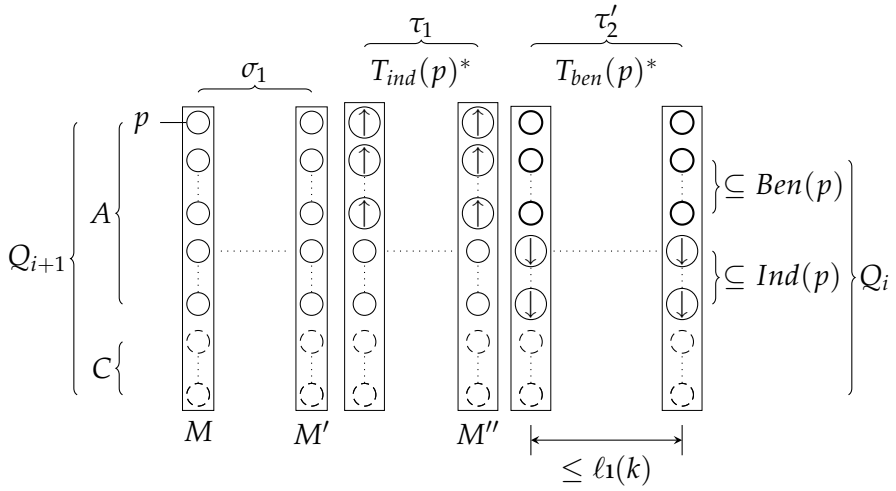

Figure 3: Sequences and bounds used in the proof of Lemma 9
↑ (resp. ↓) inside places indicates that tokens are non-decreasing (resp. non-increasing).

LEMMA 10. *$\ell_1(0) \leq 2^a$ and $\ell_1(i+1) \leq (W\ell_1(i)+R)^{i+1}2^a + \ell_1(i)$.*

PROOF. (Following [20].) For any $Q \subseteq P$, buffer p and Q-run $\sigma \in T_{ben}(p)^*$, if two intermediate vectors of the run are equal when restricted to $Q \cap Ben(p)$, the subsequence between these two vectors can be removed and the remaining sequence will still be a Q-run and retains the covering properties of σ. This is because the removed subsequence doesn't affect places in $Q \cap Ben(p)$ and doesn't increase tokens in any place in $Q \cap Ind(p)$.

The bound on $\ell_1(0)$ is due to the above observation and the fact that component places are 1-bounded and there are 2^a possible distinct vectors when restricted to C. For $\ell_1(i+1)$, suppose there is a Q_{i+1}-covering run $\sigma \in T_{ben}(p)^*$ for some buffer p with $|Q_{i+1} \cap B \cap Ben(p)| = i+1$. If some buffer $p' \in Ben(p) \cap Q_{i+1}$ has more than $W\ell_1(i)+R$ tokens at some intermediate vector M, we can apply the same kind of reasoning used in Lemma 9. ∎

It now only remains to solve the recurrence relations we have obtained and use them in a nondeterministic algorithm that guesses covering sequences to get our first main theorem.

DEFINITION 11. *Let $W' = \max\{W,2\}$, $R' = \max\{R,2\}$. Define a growth function $g : \mathbb{N} \to \mathbb{N}$ as $g(0) = W'R'2^a$ and $g(i+1) = (g(i))^{3(i+1)}2^a$.*

LEMMA 12. *$\ell(K+j) \leq (K+j)(W\ell_1(K)+R)^{K+j}2^a + j\ell_1(K) + \ell(K)$.*

LEMMA 13. *$\ell_1(i), \ell(i) \leq g(i) \leq (W'R')^{3^i i!}2^{6^i i! a}$ and $\ell(K+j) \leq (K+j)(g(K))^{3(K+j)}2^a$.*

THEOREM 14. *Suppose a net under consideration has benefit depth K. There is a nondeterministic algorithm that decides if there is a firing sequence covering M_{cov} from M_0 in space $\mathcal{O}(\log|M_0| + \log n + (\log W' + \log R')6^{K+2}K!m^3 \log m)$.*

PROOF. Since there are b buffers in the net, $\ell(b)$ gives an upper bound on the length of the shortest P-covering run. Therefore, there exists a P-covering run iff there is one of length at most $\ell(b)$. From Lemma 13 we get

$$\ell(b) \leq b(g(K))^{3b}2^a \leq m(g(K))^{3m}2^a \leq m\left((W'R')^{3^K K!}2^{6^K K!a}\right)^{3m}2^a \leq m\left((W'R')^{6^{K+1}K!a}\right)^{3m}2^a$$

Hence $\ell(b) \leq m(W'R')^{6^{K+2}K!m^2}$. A nondeterministic algorithm can guess a sequence of transitions of this length and verify that it is P-covering from M_0. The memory needed is dominated by a counter to count up to maximum $\ell(b)$ and the memory needed to store intermediate markings. The memory needed for the counter is $\mathcal{O}((\log W' + \log R')6^{K+2}K!m^2 \log m)$ and to store markings we need $\mathcal{O}(\log |M_0| + \log n + (\log W' + \log R')6^{K+2}K!m^3 \log m)$. ∎

Given a net, its benefit depth K can be computed in polynomial time. Hence, the upper bound on the memory requirement in the above theorem is space constructible and the well known Savitch's theorem can be applied to determinize the above algorithm (see any standard text on complexity theory). The memory required will still be polynomial in the size of the input net and this gives us the paraPSPACE algorithm.

For later use in section 5, we name the exponent $6^{K+2}K!m^2$ used in the above proof $expcov(1)$, and let $expcov(i) = expcov(1)^i$.

4 Benefit depth and boundedness

In this section, we will tighten Rosier and Yen's analysis [21] and prove that the complexity of boundedness problem is paraPSPACE when parameterized by benefit depth. As in coverability, we segregate transitions that reduce tokens from a place and those that do not.

DEFINITION 15. *Let $U \subseteq B$ be a subset of buffers, $Q \subseteq P$ a subset of places and M a vector. A Q-run σ from M is said to be U-**self-covering** if it can be decomposed as $\sigma_1\sigma_2$ with $M \xrightarrow{\sigma_1} M_1 \xrightarrow{\sigma_2} M_2$, $M_2 \geq M_1$ and for all $p \in U$, $M_2(p) > M_1(p)$. We call σ_2 as the **pumping portion** of the self-covering sequence.*

It is well known that a place p is unbounded iff there is a firing sequence that is U-self-covering from the initial marking[†] for some $U \subseteq P$ with $p \in U$. In the rest of this section, we will fix a non-empty subset U of places and refer to U-self-covering sequences as self-covering sequences. Let $T_{dep}(p) = \{t \in T_{ben}(p) \mid \forall p' \in Ind(p) : Pre(p', t) = 0\}$.

DEFINITION 16. *Let $C \subseteq Q \subseteq P$ and $p \in B$ be a buffer. Let $scov^p(Q, M)$ be the length of the shortest Q-run in $T_{ben}(p)^*$ that is self-covering from the vector M with the pumping portion of the sequence in $T_{dep}(p)^*$. If there is no such sequence, define $scov^p(Q, M)$ to be 0. Let $s_1(i) = \max\{scov^p(Q, M) \mid M$ a vector, $|Q \cap Ben(p) \cap B| = i\}$. Also, let $scov(Q, M)$ be the length of the shortest self-covering Q-run from the vector M and 0 if there is no such sequence. Let $s(i) = \max\{scov(Q, M) \mid M$ a vector, $|Q \cap B| = i\}$.*

LEMMA 17. *For $0 \leq i < b$, $s(i+1) \leq (W^2 s_1(K))^{poly(m)} + s_1(K) + (Ws_1(K) + 2)s(i)$ for $poly(m)$ a polynomial in m with degree independent of W, m, K.*

PROOF. Suppose that $Q = Q_{i+1} = C \cup A$ with $|A| = i + 1$ and that there is a self-covering Q_{i+1}-run σ from some vector M. If this run is $Ws_1(K)$-bounded for Q_{i+1}, the required result

[†]We thank an anonymous FSTTCS referee for pointing out a mistake here.

is a consequence of Lemma 2.2 in [21]. Otherwise some buffer $p \in Q_{i+1}$ has more than $Ws_1(K)$ tokens at some intermediate vector M'. The sequence occurring after M' can be replaced by a self-covering Q_i-run σ_2 of length at most $s(i)$, with $Q_i = Q_{i+1} \setminus \{p\}$. By repeated application of Lemma 8, rearrange the non-pumping portion of σ_2 into $\tau_1\tau_2$ and the pumping portion into $\tau_1'\tau_2'$, with $\tau_1, \tau_1' \in T_{ind}(p)^*$ and $\tau_2, \tau_2' \in T_{ben}(p)^*$ (see Fig. 4). τ_2'

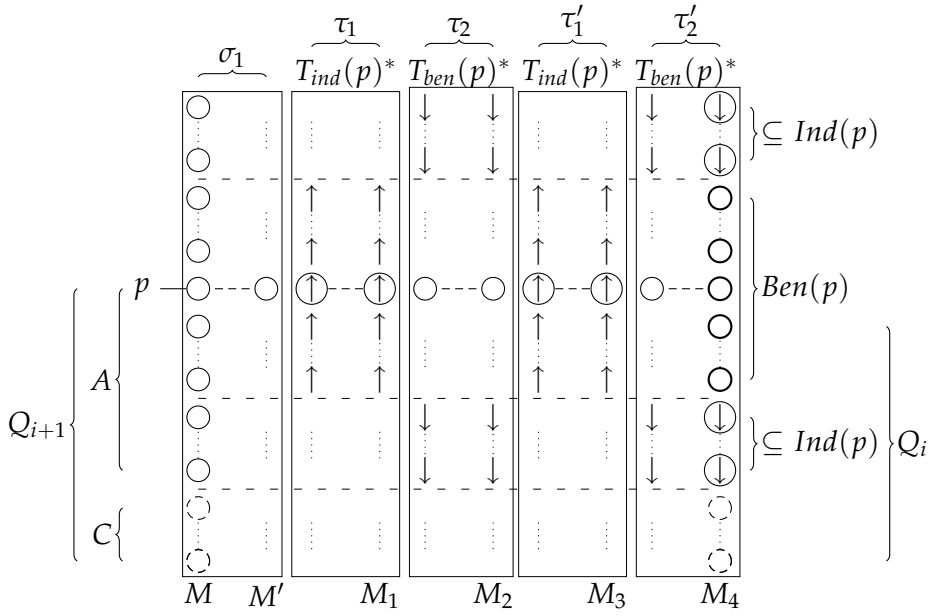

Figure 4: Sequences and bounds used in the proof of Lemma 17

is a sequence in $T_{ben}(p)^*$ that "pumps up" tokens in some of the places and hence can be replaced by another one τ_2'' of length at most $s_1(K)$. τ_2'' can however decrease tokens from places that are pumped up by τ_1', so we compensate for it by firing τ_1' $Ws_1(K)+1$ times. Putting everything together, we get $\tau_1\tau_2\tau_1'^{Ws_1(K)}\tau_2''$ is a self-covering Q_{i+1}-run from M'. ∎

The following lemmas give recurrence relations for length of self-covering sequences in $T_{ben}(p)^*$. The proofs are similar to those of corresponding lemmas in [21] with the additional fact that transitions in $T_{ben}(p)$ don't increase tokens in $Ind(p)$. As before, $W' = \max\{W, 2\}$.

LEMMA 18. *Let $C \subseteq Q \subseteq P$ and $p \in B$ a buffer. For $c \in \mathbb{N}$, suppose there is a self-covering Q-run in $T_{ben}(p)^*$ from some vector M which is c-bounded for $Q \cap Ben(p) \cap B$. If its pumping portion is in $T_{dep}(p)^*$, then a similar sequence exists whose length is at most $(W'c2^a)^{poly(K)}$ for $poly(K)$ some polynomial in K whose degree is independent of W, c, a, K.*

LEMMA 19. *$s_1(0) \leq (W'2^a)^{poly(K)}$ and $s_1(i+1) \leq (W'^2 s_1(i) 2^a)^{poly(K)}$.*

Now we give upper bounds for these recurrence relations and use them in a nondeterministic algorithm. A technical point is that the recurrence relation in Lemma 17 for $s(i)$ starts from $i = 1$ (unlike that in Lemma 9). This avoids the calculation of an upper bound for $s(u)$ using Lemma 20 below from containing terms m^K in the exponent, which is not acceptable in paraPSPACE algorithms.

LEMMA 20. *For $0 < i < b$, we have $s_1(i) \leq W'^{2(i+1)poly(K^{i+1})}2^{a(i+1)poly(K^{i+1})}$, as also $s(i) \leq 2^{i-1}(4Ws_1(K))^{i-1}(W^2s_1(K))^{poly(m)} + (4Ws_1(K))^i s(0)$.*

THEOREM 21. *There is a nondeterministic algorithm that decides if a net is bounded in space $\mathcal{O}(\log|M_0| + \log W'K^{cK}m^c a + m\log n)$ where c is some constant.*

5 The model checking algorithm

We now show that checking whether a given system (N, M_0) satisfies a given formula ϕ of the logic defined in sub-section 2.1 can be done in paraPSPACE with benefit depth as the parameter. This requires a lot of technical work. First of all, we simplify the kind of formulas that our algorithm has to handle by nondeterministically choosing a disjunct from a disjunctive subformula. We then end up with ϕ a sequence of conjuncts $\beta_1, \ldots, \beta_c, \kappa$, where each β_i is of the form $\{\tau_1, \cdots \tau_r\} < \omega$ or $\{\tau_1, \cdots \tau_r\} = \omega$ and κ consists of conjunctions of nested **EF** modalities over $\tau \geq c$ formulas. If we can check such formulas in paraPSPACE, Savitch's theorem ensures that satisfiability of ϕ can be checked in paraPSPACE.

For checking β_i, we need the following lemma. The proof of this lemma relies on some results on Karp-Miller trees, in particular on [8, Theorems 21 and 22]. Recall that every term τ gives a function $L_\tau : P \to \mathbb{N}$ such that τ is syntactically equivalent to $\sum_{p\in P} L_\tau(p)p$.

LEMMA 22. *$N, M_0 \models \{\tau_1, \ldots, \tau_r\} = \omega$ iff there exists a U-self-covering sequence for some $U \subseteq P$ such that for every $j \in \{1.\ldots, r\}$, there is a $p_j \in U$ with $L_{\tau_j}(p_j) \geq 1$.*

Hence, checking of β_i can be done in paraPSPACE by using results of section 4.

We now consider verifying the formulas κ, which are of the form $\gamma \wedge \mathbf{EF}(\kappa_1) \wedge \cdots \wedge \mathbf{EF}(\kappa_r)$, with γ having only conjunctions of $\tau \geq c$ formulas. We call γ the **content** of κ and $\kappa_1, \ldots, \kappa_r$ as the **children** of κ. Each of the children may have their own content and children, thus generating a tree with nodes Γ, with κ at the root of this tree. We will represent nodes of this tree by sequences of natural numbers, 0 being the root.

The maximum length of sequences in Γ is one more than the nesting depth of the **EF** modality in κ and we denote it by D. Let $[D] = \{0, 1, \ldots, D-1\}$. If $\alpha \in \Gamma$ is a tree node that represents the formula $\kappa(\alpha) = \gamma \wedge \mathbf{EF}(\kappa_1) \wedge \cdots \wedge \mathbf{EF}(\kappa_r)$, $content(\alpha) = \gamma$ denotes the content of the node α. Let $ratio(\tau \geq c) = max\{\lceil c/L_\tau(p)\rceil \mid L_\tau(p) \neq 0, p \in P\}$. Defining $\max(\emptyset) = 0$, we define the maximum ratio at height i in the tree by $ratio(i) = \max\{ratio(\tau \geq c) \mid \tau \geq c$ appears as a conjunct in $content(\alpha)$ for some $\alpha \in \Gamma, |\alpha| = i+1\}$. Recall from Definition 6 that b is the number of buffers and $\ell'(M)$ the length of the shortest run covering M using all the buffers $\ell(b, M)$.

DEFINITION 23. *Given a formula κ and a system (N, M_0), the bound function $f : [D] \times P \to \mathbb{N}$ is defined as follows. We use $f(j)$ for the marking defined by $f(j)(p) = f(j, p)$.*

- $f(D-1, p) = ratio(D-1)$,
- $f(D-i, p) = \max\{ratio(D-i), W\ell'(f(D-i+1)) + f(D-i+1, p)\}, 1 < i < D$,
- $f(0, p) = M_0(p)$.

A guess function $h : \Gamma \times P \to \mathbb{N}$ is any function that satisfies $h(\alpha, p) \leq f(|\alpha| - 1, p)$ for all $\alpha \in \Gamma$ and $p \in P$. If h is a guess function, $h(\alpha)$ is the marking defined by $h(\alpha)(p) = h(\alpha, p)$.

If a given system satisfies the formula $\kappa = \gamma \wedge \mathbf{EF}(\kappa_1) \wedge \cdots \wedge \mathbf{EF}(\kappa_r)$, then there exist firing sequences $\sigma_{01}, \ldots, \sigma_{0r}$ that are all enabled at the initial marking M_0 such that $M_0 \stackrel{\sigma_{0i}}{\Longrightarrow} M_{0i}$ and M_{0i} satisfies κ_i. In general, if κ generates a tree with set of nodes Γ, then there is a set of sequences $\{\sigma_\alpha \mid \alpha \in \Gamma \setminus \{0\}\}$ and set of markings $\{M_\alpha \mid \alpha \in \Gamma\}$ such that $M_\alpha \stackrel{\sigma_{\alpha j}}{\Longrightarrow} M_{\alpha j}$ for all $\alpha, \alpha j \in \Gamma$ and M_α satisfies $content(\alpha)$ for all $\alpha \in \Gamma$.

LEMMA 24. *There exist sequences $\{\mu_\alpha \mid \alpha \in \Gamma \setminus \{0\}\}$ and markings $\{M_\alpha \mid \alpha \in \Gamma\}$ such that $M_\alpha \stackrel{\mu_{\alpha j}}{\Longrightarrow} M_{\alpha j}$ for all $\alpha, \alpha j \in \Gamma$ with M_α satisfying $content(\alpha)$ and $|\mu_\alpha| \leq \ell'(f(|\alpha| - 1))$ iff there exist sequences $\{\sigma_\alpha \mid \alpha \in \Gamma \setminus \{0\}\}$ and markings $\{M'_\alpha \mid \alpha \in \Gamma\}$ (M'_0 should be equal to M_0) such that $M'_\alpha \stackrel{\sigma_{\alpha j}}{\Longrightarrow} M'_{\alpha j}$ for all $\alpha, \alpha j \in \Gamma$ with M'_α satisfying $content(\alpha)$.*

To derive an upper bound for $f(i)$ to use in a nondeterministic algorithm, let $R = \max\{ratio(\tau \geq c) \mid \tau \geq c \text{ is a subformula of } \kappa\}$, $R' = \max\{R, 2\}$ and $W' = \max\{W, 2\}$. Recall that $D - 1$ is the nesting depth of **EF** and note that boundedness and coverability can be expressed with $D \leq 2$.

LEMMA 25. *For $i \geq 2$, $f(D - i, p) \leq (i + 1)R'W\ell'(f(D - i + 1))$.*

LEMMA 26. *Recall from the end of section 3 that $expcov(i) = (6^{K+2}K!m^2)^i$. Then $\ell'(f(D - 1)) \leq m(W'R')^{expcov(1)}$ and $\ell'(f(D - i)) \leq m \prod_{j=D-i}^{D} \left((D - j + 1)W'^2R'm\right)^{expcov(i+j+1-D)}$.*

THEOREM 27. *Given a net and a formula ϕ, if the benefit depth of the net is treated as a parameter and the nesting depth D of **EF** modality in the formula is treated as a constant, then there is a* paraPSPACE *algorithm that checks if the net satisfies the given formula.*

PROOF. By Lemma 24, it is enough for a nondeterministic algorithm to guess sequences $\sigma_{\alpha j}$, $\alpha j \in \Gamma$ of lengths at most $\ell'(f(|\alpha j| - 1))$ and verify that they satisfy the formula. Using bounds given by Lemma 26 and an argument similar to the one in the proof of Theorem 14, it can be shown that the space used is exponential in K and polynomial in the size of the net and numeric constants in the formula. This gives the paraPSPACE algorithm. ∎

The space requirement of the above algorithm will have terms like m^{2D} and hence it will not be paraPSPACE if D is treated as a parameter instead of a constant.

6 Conclusion

We considered nets communicating with buffers. These are infinite-state concurrent systems allowing 1-safe Petri net components communicating through synchronization, which in turn communicate asynchronously through a fixed set of buffers. We identified the parameter benefit depth that measures the maximum number of other buffers that any one buffer can influence. We showed that based on this parameter, paraPSPACE algorithms can be obtained for the coverability and boundedness problems. Note that this does *not* yield a paraPSPACE algorithm for the reachability problem. Whether benefit depth can yield such an algorithm is open; for work of this kind we refer to Kostin [16]. We then extended the above technique to show that satisfiability of formulas of the logic given in sub-section 2.1 can be checked in paraPSPACE if the nesting depth of **EF** quantifiers in such formulas is treated as a constant.

References

[1] M. F. Atig, A. Bouajjani, and T. Touili. On the reachability analysis of acyclic networks of pushdown systems. In *CONCUR*, volume 5201 of *LNCS*, pages 356–371, 2008.

[2] M. F. Atig and P. Habermehl. On Yen's path logic for Petri nets. In *RP 2009*, volume 5797 of *LNCS*, pages 51–63, 2009.

[3] D. Brand and P. Zafiropulo. On communicating finite-state machines. *JACM*, 30, 1983.

[4] D. Caucal. On the regular structure of prefix rewriting. *TCS*, 106:61–86, 1992.

[5] G. Cécé, A. Finkel, and S. Purushothaman Iyer. Unreliable channels are easier to verify than perfect channels. *Inf. Comput.*, 124(1):20–31, 1996.

[6] A. Cheng, J. Esparza, and J. Palsberg. Complexity results for 1-safe nets. *Theoret. Comp. Sci.*, 147(1-2):117–136, 1995.

[7] J. Desel and J. Esparza. *Free choice Petri nets*, volume 40 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, 1995.

[8] J. Desel and W. Reisig. *Place transition Petri nets*, volume 1491 of *LNCS*. 1998.

[9] R. G. Downey and M. R. Fellows. *Parameterized complexity*. Springer-Verlag, 1999.

[10] J. Esparza. *Decidability and complexity of Petri net problems — An introduction*, volume 1491 of *LNCS*, pages 374–428. 1998.

[11] J. Flum and M. Grohe. Describing parameterized complexity classes. *Inf. Comput.*, 187(2):291–319, 2003.

[12] P. Habermehl. On the complexity of the linear-time μ-calculus for Petri-nets. In *ATPN '97*, volume 1248 of *LNCS*, pages 102–116, 1997.

[13] R. Howell, L.E. Rosier, and H.-C. Yen. A taxonomy of fairness and temporal logic problems for petri nets. *Theoret. Comp. Sci.*, 82(2):341–372, 1991.

[14] R.M. Karp and R.E. Miller. Parallel program schemata. *JCSS*, 3(2):147–195, May 1969.

[15] S.R. Kosaraju. Decidability of reachability in vector addition systems. In *Proc. 14th STOC*, pages 267–281. ACM, 1982.

[16] A.E. Kostin. Using transition invariants for reachability analysis of Petri nets. In V. Kordic, editor, *Petri net: theory and applications*, pages 435–458. I-Tech Edu. Pub., 2008.

[17] R. Lipton. The reachability problem requires exponential space. Yale university, 1975.

[18] E.W. Mayr. An algorithm for the general Petri net reachability problem. *SIAM J. Comput.*, 13(3):441–460, 1984.

[19] C.A. Petri. *Kommunikation mit Automaten*. PhD thesis, Inst. Instrumentelle Math., 1962.

[20] C. Rackoff. The covering and boundedness problems for vector addition systems. *Theoret. Comp. Sci.*, 6:223–231, 1978.

[21] L.E. Rosier and H.-C. Yen. A multiparameter analysis of the boundedness problem for vector addition systems. *J. Comput. Syst. Sci.*, 32(1):105–135, 1986.

[22] P. Schnoebelen. Verifying lossy channel systems has nonprimitive recursive complexity. *Inf. Proc. Lett.*, 83(5):251–261, 2002.

[23] R. Valk and G. Vidal-Naquet. Petri nets and regular languages. *JCSS*, 23:299–325, 1981.

[24] H.-C. Yen. A unified approach for deciding the existence of certain petri net paths. *Inf. Comput.*, 96(1):119–137, 1992.