Boundedness of Reset P/T Nets

C. Dufourd[1], P. Jančar[2*], and Ph. Schnoebelen[1]

[1] Lab. Spécification et Vérification, CNRS UMR 5522 & ENS de Cachan, France
[2] Dept. of Computer Science, FEI VŠB-TU, Tech. Univ. Ostrava - Czech Rep.
email: {dufourd,phs}@lsv.ens-cachan.fr, Petr.Jancar@vsb.cz

Abstract. P/T nets with reset and transfer arcs can be seen as counter-machines with some restricted set of operations. Surprisingly, several problems related to boundedness are harder for Reset nets than for the more expressive Transfer nets. Our main result is that boundedness is undecidable for nets with three reset arcs, while it is decidable for nets with two resetable places.

1 Introduction

"Infinite-state transition systems" is a generic name for a large collection of computational models that go beyond finite-state systems but where some decidability results exist. These *infinite systems* (for short) are currently under active study for several reasons. First some applications use models that cannot easily be abstracted into finite-state systems (e.g., hybrid systems). Then, some models only go beyond finite-state in a very specific way (e.g., parameterized systems, lossy channel systems). Finally, verification methods designed for infinite-state systems may help understand where and how to avoid the famous *state explosion problem* in the finite-state framework.

P/T nets are a classical infinite-state model which remains partially analyzable. Many properties (reachability, safety properties, termination, etc.) are decidable and many are not (bisimilarity, language equivalence, etc.). Here we are not concerned with their use in modelling concurrency: we see them as *weak* counter machines that do not have a full-fledged test for zero (i.e., transitions can only test negatively for emptiness of a counter).

Reset nets are P/T nets extended with the possibility of resetting (setting to zero) the content of places. This adds expressive power and makes reachability undecidable [AK77]. *Transfer nets* are even more powerful: they can transfer the whole content of some place to some other one. [DFS98,Duf98] investigate decidability issues in the verification of Reset and Transfer nets : compared to P/T nets, the decidability of reachability is lost but termination, coverability and inevitability remain decidable.

* P. Jančar is partially supported by the Grant Agency of the Czech Republic, Grant No. 201/97/0456

The most surprising result in [DFS98] [1] is the undecidability of boundedness for Reset nets, knowing it is clearly decidable for Transfer nets [FS98]. Boundedness is a fundamental question: it is to *space* what termination is to *time*. Indeed, termination and boundedness are closely related problems. Boundedness has been neglected in model-checking of infinite systems because most approaches only refer to the observable behaviour of systems. However, when studying potentially infinite systems, a first question is to check if they really are infinite !

Obviously, boundedness issues for Reset and Transfer nets are difficult questions. [DFS98] uses an involved reduction from Hilbert's 10th problem. [KCK+97] gives a faulty decidability proof for a more general model.

Our contribution. We give a new and simpler proof that boundedness is undecidable for Reset nets *even for nets with at most 3 reset arcs* [2].

We prove that this result is tight and that *boundedness is decidable for Reset nets with 2 resetable places.* This threshold value is surprising since 2 reset arcs are enough to simulate strong counter machines quite faithfully (see Theorem 11 in [DFS98]). Exhibiting the exact threshold value is our main result. The proof is based on the possibility of producing a "semilinear" witness of unboundedness. Since we are close to the boundary of this possibility, it is natural that the proof is technically nontrivial. A similar situation appears, e.g., in the case of VASS's: in dimension 2 they are always semilinear, unlike in dimension 3 [HP79].

We also investigate related problems and complete the following table:

	Reset nets	Transfer nets	P/T nets
boundedness	Σ_1-complete	Decidable	Decidable
place-boundedness	Σ_1-complete	Σ_1-complete	Decidable
structural boundedness	Π_2-complete	Π_1-complete	Decidable (PTIME !)
termination	Decidable	Decidable	Decidable
structural termination	Π_1-complete	Π_1-complete	Decidable (PTIME !)

Our results and techniques have applications to models related to Reset net (see below).

Related Works. [AK77] shows that reachability is undecidable when P/T nets are extended with reset arcs. [AL78] shows that coverability is decidable for VASS's with reset arcs. [May98] gives several undecidability results for lossy counter machines with applications to, e.g., Reset and Transfer nets.

Reset and Transfer nets are a subset of Valk's self-modifying nets [Val78]. [LC94,Cia94] consider them from an expressivity perspective.

For many purposes, Reset and Transfer nets are very close to other infinite-state models currently under study. The undecidability of boundedness for Reset

[1] Another surprise is that place-boundedness ("is a given place bounded ?") is not decidable for Transfer nets while boundedness is. Boundedness and place-boundedness are very similar problems and many papers confuse them.

[2] For lack of space, most proofs are sketched or omitted. They appear in the full version available from the authors.

nets translates into undecidability of boundedness for Lossy VASS's with zero test [BM99]. *Parameterized broadcast protocols* can be encoded into Transfer nets and [EFM99] uses a crucial counter-example inspired from our Figure 1. [BO98] introduces *reconfigurable nets* and shows that boundedness is decidable. These nets can in fact be seen as Transfer nets.

2 Reset and Transfer nets

A P/T net is a tuple $N = \langle P, T, F\rangle$ where $P = \{p_1, ...\}$ and $T = \{t_1, ...\}$ are two disjoint finite sets (of *places*, of *transitions*), and where $F \subseteq P \times T \cup T \times P$ is a set of *arcs*; we denote ${}^\bullet t = \{p \mid (p,t) \in F\}$, and $t^\bullet = \{p \mid (t,p) \in F\}$. We assume the usual graphical presentation.

A *marking* m of N can be identified with a multiset of places; we also say that $m(p)$ denotes the number of *tokens* on place p in m. By ($m < m'$) $m \leq m'$ we denote the (proper) multiset inclusion. A *marked net* is a net with some initial marking m_0.

We write $m \xrightarrow{t} m'$ iff $m \geq {}^\bullet t$ and $m' = m - {}^\bullet t + t^\bullet$ (first the enabling tokens are removed and then the output tokens are added). We use $m \rightarrow m'$ when abstracting the transition; similarly we use $m \xrightarrow{u} m'$ (with $u \in T^*$) and $m \xrightarrow{*} m'$.

A *Reset net* is a P/T net with special "reset arcs". Formally, it is a tuple $\langle P, T, F, F_\mathrm{R}\rangle$ where $\langle P, T, F\rangle$ is a P/T net and $F_\mathrm{R} \subseteq T \times P$ is a set of *reset arcs*. Reset arcs do not change the requirements for enabling a transition. But when $m \xrightarrow{t} m'$, a reset arc (t, p) sets p to zero so that $m'(p) = 0$ ($m'(p) = 1$ if (t,p) is in both F and F_R). We depict reset arcs by crossed edges as in Fig. 1 where $F_\mathrm{R} = \{(t_2, p_3), (t_4, p_4)\}$ and $m_0 = (1, 0, 1, 0)$. Observe that $(1, 0, n, 0) \xrightarrow{w(n)} (1, 0, n+1, 0)$ for $w(n) = t_1^n t_2 t_3^n t_4$. Hence the net is unbounded. However no fixed $u, u' \in T^*$ exist s.t. $m_0 \xrightarrow{u} m_1 \xrightarrow{u'} m_2 \xrightarrow{u'} m_3 \cdots$ leads to unboundedness.

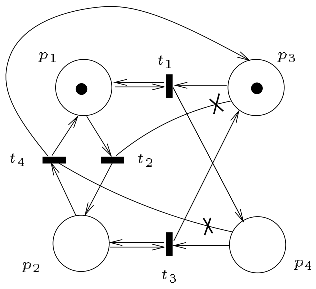

Fig. 1. An unbounded Reset net

A *Transfer net* is a P/T net with special "transfer arcs". Formally, it is a tuple $\langle P, T, F, F_\mathrm{T}\rangle$ where $\langle P, T, F\rangle$ is a P/T net and $F_\mathrm{T} \subseteq P \times T \times P$ is a set of *transfer arcs* (with the condition that for any p, t there is at most one $(p, t, p') \in F_\mathrm{T}$). Transfer arcs do not change the requirements for enabling a transition. But when $m \xrightarrow{t} m'$, m' is obtained by (1) removing the enabling tokens, then (2) transferring all tokens from p to p' when $(p, t, p') \in F_\mathrm{T}$, and finally (3) adding the usual output tokens. We omit the tedious formal definitions.

Boundedness and termination problems. A place p is *bounded* in a marked net if there is a bound $k \in \mathbb{N}$ s.t. $m(p) \leq k$ for all m reachable from m_0. A marked net is *bounded* if all its places are. A net is *structurally bounded* if it is bounded for any initial marking (note that the bound k may vary with m_0).

A marked net is *terminating* if it admits no infinite run. A net is *structurally terminating* if it is terminating for any initial marking.

Reset (and Transfer) nets enjoy the monotonicity property: $m_1 \xrightarrow{t} m_2$ and $m'_1 > m_1$ imply $m'_1 \xrightarrow{t} m'_2$ with $m'_2 \geq m_2$ ($m'_2 > m_2$ for Transfer nets). Hence

Lemma 1. *A marked Reset net is unbounded iff it has an unbounded nondecreasing run.*

where an *unbounded run* is an infinite sequence of moves $m_0 \xrightarrow{t_1} m_1 \xrightarrow{t_2} m_2 \cdots$ s.t. some place(s) are unbounded along the run (get arbitrarily large values in the m_i's), and where a run is *nondecreasing* if $i < j$ implies $m_i \not\geq m_j$.

Lemma 2. *A marked Transfer net is unbounded iff there exists two sequences $u, u' \in T^*$ s.t. $m_0 \xrightarrow{u} m_1 \xrightarrow{u'} m_2$ and $m_1 < m_2$.*

There exists an effective procedure searching for such u, u', and its termination relies on Dickson's lemma: every infinite sequence of elements of $\mathbb{N}^k$ has an infinite ascending subsequence. The corollary is that boundedness of Transfer nets is decidable.

Transfer nets can simulate Reset nets. If we add a dummy place *dum*, any reset arc (t, p) can be simulated by a transfer arc (p, t, dum). [3] This simulation is quite faithful and preserve almost all behavioural aspects of the nets. That is, except boundedness ! Hence decidability of boundedness for Transfer nets does not entail decidability of boundedness for Reset nets. Figure 1 shows that Lemma 2 does not hold for Reset nets.

Counting reset arcs. If (t, p) is a reset arc, we say p is a *resetable place*. For boundedness issues, there is no need for several reset arcs on the same place.

Lemma 3. *If N is a Reset net with k resetable places, then it is possible to construct a modified N' with k reset arcs s.t. for any initial marking m_0, N with m_0 is bounded iff N' with the corresponding m'_0 is.*

3 Undecidability of boundedness for Reset nets

An *m-counter machine* M, with *nonnegative* counters $c_1, c_2, \ldots, c_m$, is a sequence of (labelled) instructions $\mathtt{0} : \mathtt{instr_0}$; $\mathtt{1} : \mathtt{instr_1}$; $\ldots$; $\mathtt{n} : \mathtt{instr_n}$ where $\mathtt{instr_n} = \mathtt{HALT}$ and $\mathtt{instr_i}$, $\mathtt{i} = \mathtt{0}, \mathtt{1}, \ldots, \mathtt{n} - \mathtt{1}$, are of the following two types (assuming $\mathtt{0} \leq \mathtt{k}, \mathtt{k_1}, \mathtt{k_2} \leq \mathtt{n}$ and $\mathtt{c} \in \{\mathtt{c_1}, \mathtt{c_2}, \ldots, \mathtt{c_m}\}$)

[3] It is believed that Transfer nets are strictly more expressive (in terms of generated languages) than Reset nets but we know of no formal proof for this. Indeed, the example given to that purpose in [Cia94] (his Fig. 14) is flawed and can be encoded into a Reset net.

– $\mathtt{c := c + 1;\ goto\ k}$
– $\mathtt{if\ c > 0\ then\ (c := c - 1;\ goto\ k_1)\ else\ goto\ k_2}$

We shall refer to the obvious notion of the (unique) *computation* of M *on given input values* of the counters (the computation is either finite, i.e., halting, or infinite). Note that the order of instructions does not matter when we fix that $\mathtt{0 : instr_0}$ is the initial (starting) instruction.

We say that M is (space-) *bounded for given input values* iff there is $k \in \mathbb{N}$ s.t. the sum $\mathtt{c_1} + \cdots + \mathtt{c_m}$ of the counters never exceeds k during the computation on the given input values. Since even 2-counter machines can simulate Turing machines, it is clear that the problem whether a given 2-counter machine is bounded for input values $0, 0$, is undecidable.

Lemma 4. *Given any 2-counter machine M, we can construct a marked Reset net N_M with 3 reset arcs s.t. M is bounded for inputs $0, 0$ iff N_M is bounded.*

Proof. We first transform M into a 3-counter machine M_1 (with a slightly richer set of allowed instructions) as follows: we add a third "capacity" counter $\mathtt{K}$, replace every command (subinstruction) $\mathtt{c := c - 1}$ by $(\mathtt{c := c - 1;\ K := K + 1})$, and every $\mathtt{c := c + 1}$ by a "controlled incrementation" :

$$(\mathtt{if\ K > 0\ then\ (K := K - 1;\ c := c + 1)\ else\ goto\ n{+}1}).$$

Finally we add 3 instructions

$\mathtt{n{+}1 : if\ c_1 > 0\ then\ (c_1 := c_1 - 1;\ K := K + 1;\ goto\ n{+}1)\ else\ goto\ n{+}2},$
$\mathtt{n{+}2 : if\ c_2 > 0\ then\ (c_2 := c_2 - 1;\ K := K + 1;\ goto\ n{+}2)\ else\ goto\ n{+}3},$
$\mathtt{n{+}3 : K := K + 1;\ goto\ 0}.$

In M_1 only $\mathtt{instr_{n+3}}$ changes (increases by 1) the sum $\mathtt{c_1{+}c_2{+}K}$ of the counters. M is bounded for inputs $0, 0$ iff M_1 is bounded for inputs $0, 0, 0$ (an unbounded run of M yields an unbounded run of M_1: once $\mathtt{K}$ is empty, a controlled incrementation has M_1 restarting the computation with one more token in $\mathtt{K}$).

Now we also allow commands (subinstructions) $\mathtt{c := 0}$, and introduce the nondeterministic choice operator $\mathtt{[]}$. In M_1, we replace every (sub)instruction $\mathtt{if\ c > 0\ then\ comm_1\ else\ comm_2}$ $(\mathtt{c} \in \{\mathtt{c_1, c_2, K}\})$ by $\mathtt{comm_1\ []\ (c := 0; comm_2)}$, assuming that any attempt to perform $\mathtt{c := c - 1}$ fails when $\mathtt{c = 0}$. Thus we get a *nondeterministic* machine M_2. Again only $\mathtt{instr_{n+3}}$ increases the sum of the counters while the other instructions decrease or do not change it.

One can now verify that M_1 is bounded for inputs $0, 0, 0$ iff M_2 is bounded for inputs $0, 0, 0$, i.e., *all* computations of M_2 are bounded. Indeed, any unbounded run of M_2 uses $\mathtt{instr_{n+3}}$ infinitely often as well as $\mathtt{instr_{n+2}}$, $\mathtt{instr_{n+1}}$ and $\mathtt{instr_0}$. After these instructions, the same initial state is reached, possibly with one more token in $\mathtt{K}$. Thus, if there is an unbounded computation, there is one which performs $\mathtt{c := 0}$ only when $\mathtt{c = 0}$ and this gives an unbounded run of M_1.

Now there only remains to check the easy fact that M_2 can be faithfully simulated by a Reset net N_M, with the place set consisting of "control places" $e_0, e_1, e_2, \ldots, e_{n+3}$ (there is always just one token in one of them, the other being empty) and "counter places" c_1, c_2, K. Since just 3 places are resetable in N_M, we can transform it so that it contains just 3 reset arcs (cf. Lemma 3).

Corollary 1. *Boundedness of Reset nets with three reset arcs is Σ_1-complete.*

Another corollary is that place-boundedness is Σ_1-complete for Reset and Transfer nets (membership in Σ_1 follows from the fact that coverability, "given some m, is there a reachable $m' \geq m$?", is decidable [DFS98]).

We define (an m-counter machine) M to be *totally bounded* iff it is bounded for all inputs.

Lemma 5. *The problem whether a given 2-counters machine is totally bounded is reducible to the structural boundedness problem for Reset nets.*

Lemma 6. *The problem whether a given 2-counters machine does not halt is reducible to the structural boundedness problem for Transfer nets.*

This uses reductions similar to Lemma 4 (see full version). Now, since structural boundedness clearly is in Π_1 for Transfer nets (boundedness being decidable), and in Π_2 for Reset nets (boundedness being in Σ_1), we get

Corollary 2. *Structural boundedness is Π_1-complete for Transfer nets and Π_2-complete for Reset nets.*

This completes the table from the introduction (noting that [May98] shows structural termination is Π_1-hard).

4 Decidability of boundedness for nets with 2 reset arcs

First observe that the boundedness problem is semidecidable: a standard (breadth-first) construction of the reachability set is guaranteed to terminate when this set is finite. Therefore the *semidecidability of unboundedness* is what should be demonstrated. We start with an easy subcase.

Definition 1. *For a marked Reset net, a* simple witness *is a 4-tuple m_1, m_2, u, p such that m_1 is reachable, $m_1 \xrightarrow{u} m_2$, $m_1 \leq m_2$, $m_1(p) < m_2(p)$, and p is not reset by any transition in the sequence u.*

Repeating u *ad infinitum* (from m_1) yields a run R along which p is unbounded. Hence a simple witness witnesses unboundedness.

Note that a simple witness—if there is any—is surely encountered (and recognized) when constructing the reachability tree. Therefore, unboundedness is semidecidable for nets with simple witnesses. They are characterized by:

Lemma 7. *A marked Reset net admits a simple witness iff it has a run along which either (1) a place is unbounded and reset only finitely many times, or (2) only one place is unbounded (then it can be reset infinitely often).*

easily proven using Dickson's Lemma, the pigeonhole principle, and a bounded-change property: there is $c \in \mathbb{N}$ s.t. $m \xrightarrow{t} m'$ implies $|m(p) - m'(p)| \leq c$ when (p, t) is not a reset arc ($c = 1$ in our case of ordinary arcs).

Thus we further restrict our attention only to the unbounded nets with two reset arcs and without simple witnesses, i.e., to the unbounded nets with two reset arcs—connected to different places—where along every unbounded run precisely the two resetable places are unbounded and are reset infinitely often. (Recall that Figure 1 shows a relevant example.) We now fix such a marked net N, denoting the resetable places by p_1, p_2 and the transition set by T. First observe that the set $\{m' \mid m \xrightarrow{*}_n m'\}$ is finite for any reachable m and any $n \in \mathbb{N}$, where the subscript n imposes that at most n (occurrences of) reset transitions are used (in the respective transition sequence).

We now also fix an unbounded nondecreasing run R in N and restrict our attention to the markings along R. Such a marking is presented as an (unordered) tuple $(r, p_1 \leftarrow x, p_2 \leftarrow y)$ where r ranges over submarkings on nonresetable places, while x, y are values of the resetable places. The (finite!) range of r is denoted by *fcs*; it can be viewed as a *finite control states* set. We use p, p' for denoting elements of $\{p_1, p_2\}$; by $\bar{p}$ we mean p_2 (p_1) when $p = p_1$ ($p = p_2$). Using $(r, p \leftarrow x, -)$ when denoting a marking, we mean that the value for $\bar{p}$ is unimportant ("don't care").

Our decidability proof is based on the fact that R can be supposed to have a "regular" structure. Then a finite amount of information, called a *regular witness*, demonstrates the unboundedness (as was exemplified by the scheme of $w(n)$'s for the net of Figure 1). Now we give the relevant definitions and statements.

Definition 2. *Given a pair* $\langle r, p \leftarrow b_1\rangle$, $\langle r', p' \leftarrow b_2\rangle$, *where* $b_1, b_2 \in \mathbb{N}$ *(and* $p = p'$ *is allowed), by a* path scheme, *of order* n, *we mean a triple* (w, f, x_0), *where* $w : \mathbb{N} \to T^+$, $f : \mathbb{N} \to \mathbb{N}$, $x_0 \in \mathbb{N}$, *such that* $\forall x \geq x_0 : (r, p \leftarrow b_1, \bar{p} \leftarrow x) \xrightarrow{w(x)}_n (r', p' \leftarrow b_2, \bar{p'} \leftarrow f(x))$ *(note that every* $w(x)$ *contains at most* n *reset transitions). The path scheme has the* maximum property *if for every* $x \geq x_0$ *there is no* $y > f(x)$ *s.t.* $(r, p \leftarrow b_1, \bar{p} \leftarrow x) \xrightarrow{*}_n (r', p' \leftarrow b_2, \bar{p'} \leftarrow y)$.

Note that if $p = p'$ and (w, f, x_0) has the maximum property then $x_0 \leq x \leq x'$ implies $f(x) \leq f(x')$.

A tuple $\langle r, p \leftarrow b_1\rangle$ is *sensible* iff for any $m \in \mathbb{N}$ there is $b_1' \geq b_1$, $m' \geq m$ s.t. $(r, p \leftarrow b_1', \bar{p} \leftarrow m')$ is reachable. When considering path schemes, we implicitly suppose that $\langle r, p \leftarrow b_1\rangle$ is sensible. In a straightforward way (putting $f(x) = -1$, etc.) we could handle the case of nonreachability ($(r, p \leftarrow b_1, \bar{p} \leftarrow x) \not\xrightarrow{*}_n (r', p' \leftarrow b_2, -)$). Then sensibility guarantees the existence of a path scheme with the maximum property – for any order n (and $x_0 = 0$).

By the *composition* of path schemes we mean the natural notion (implicitly assuming some obvious consistency assumptions). We now define regular path schemes – the essence of regular witnesses; it is technically straightforward to show the important fact that *composition of regular path schemes is regular.*

Definition 3. *A function* $g : \mathbb{N} \longrightarrow \mathbb{N}$ *is* (i, d)-regular, *where* $0 \leq i < d$, $i, d \in \mathbb{N}$ *iff there are rational constants* ρ_1, ρ_2 *such that: for every* $x \in \mathbb{N}$, $x \mod d = i$ *implies* $g(x) = \rho_1(x) + \rho_2$ *(*$\in \mathbb{N}$*; note that it imposes* $\rho_1 d \in \mathbb{N}$*). We refer to* ρ_1 *(*ρ_2*) as the* first (second) coefficient *of* g.

*A function $f : \mathbb{N} \longrightarrow \mathbb{N}$ is d-*regular*, where $d \in \mathbb{N}$, $0 < d$, iff there are functions $f_0, f_1, \ldots, f_{d-1}$ s.t. f_i is (i,d)-regular ($i = 0, 1, \ldots, d-1$), all f_i's have the same first coefficient ρ and $f(x) = f_i(x)$ for $x \mod d = i$; ρ is then the* coefficient *of f. We call f* regular *if it is d-regular for some d.*

Definition 4. *A* path scheme (w, f, x_0) *is* regular *if f is d-regular for some d, and for each $i \in \{0, 1, 2, \ldots, d-1\}$ we have $m \in \mathbb{N}$, $u_1, v_1, u_2, v_2, \ldots, u_m, v_m, u_{m+1} \in T^*$, and (i,d)-regular functions $g_1, g_2, \ldots, g_m$ such that: for every $x \geq x_0$, $x \mod d = i$ implies $w(x) = u_1 v_1^{g_1(x)} u_2 v_2^{g_2(x)} \ldots u_m v_m^{g_m(x)} u_{m+1}$.*

Definition 5. *A* regular witness *(for the net N) is a reachable marking $(r, p_1 \leftarrow x_0, p_2 \leftarrow 0)$ together with a regular path scheme $\langle w, f, x_0 \rangle$ (of order n for some $n \in \mathbb{N}$) which is related to the pair $\langle r, p_2 \leftarrow 0 \rangle$, $\langle r, p_2 \leftarrow 0 \rangle$ and has the property $\forall x \geq x_0 : x < f(x)$.*

A regular witness surely witnesses unboundedness. Moreover, such a witness, for convenience accompanied with a sequence demonstrating reachability of $(r, p_1 \leftarrow x_0, p_2 \leftarrow 0)$, can be naturally presented as a string in a fixed alphabet. Whether or not a given string represents a regular witness can be checked by using obvious algorithms. Therefore when a regular witness exists, it can be found by generating and checking all strings. So it remains to show that a regular witness is guaranteed to exist. For this, the next lemma is crucial.

Lemma 8. *For any ("sensible") pair $\langle r, p \leftarrow b_1 \rangle$, $\langle r', p' \leftarrow b_2 \rangle$, and $n \in \mathbb{N}$, there is a path scheme of order n which has the maximum property and is regular.*

Proof. (Sketch.) The key point is the case $n = 0$ (no reset transition used, i.e., we consider $\xrightarrow{*}_0$). Roughly speaking, this handles the segments between two neighbouring resets; that is why the ("sensible") pairs $\langle r, p \leftarrow 0 \rangle$ (p has just been reset), $\langle r', p' \leftarrow - \rangle$ (p' will be reset afterwards) are of interest.

For such a pair, consider a (maybe not regular) path scheme (w, f, x_0) of order 0 which has the maximum property and suppose that $w(x)$ is the shortest possible (among those yielding $f(x)$). Therefore the (finite) run corresponding to $w(x)$ is non-decreasing; it can neither increase (no simple witness exists).

We fix a sufficiently large constant $b \in \mathbb{N}$ (e.g., $b = k!$ where $k = |fcs|$), and imagine now the run $(r, p \leftarrow 0, \bar{p} \leftarrow x) \xrightarrow{w(x)}_0 (r', p' \leftarrow -, \bar{p}' \leftarrow f(x))$ as the corresponding *tour* in the plane – current values of p_1, p_2 give coordinates, current control state is additional (bounded) information. Note that the tour (when long) uses many *cycles* $(r'', p_1 \leftarrow x_1, p_2 \leftarrow y_1) \xrightarrow{*}_0 (r'', p_1 \leftarrow x_2, p_2 \leftarrow y_2)$ – necessarily with $(x_2 - x_1)/(y_2 - y_1) < 0$; a cycle is *elementary* when it contains no shorter cycles (and thus has length k at most). It is also clear that the tour can visit both the initial vertical and initial horizontal belt of breadth b (some of the coordinates being less or equal to b) only boundedly many times (the bound depending on k and b, not on x). When the tour leaves the belts—leaving from either the horizontal or the vertical belt—it can either *return* to the same belt or *zig (zag)* to the other. Now the idea of pumping or deleting elementary cycles helps to realize that lengths of returns are bounded. This idea also implies that a zig (or zag) segment can be supposed to use a "best-rate" elementary cycle – possibly except of a bounded prefix and suffix.

So we have boundedly many possibilities, *patterns*, how to zig-zag and how to arrange the (bounded) segments in between. To each pattern pat, there is an associated regular path scheme $\langle w_{pat}, f_{pat}, -\rangle$; the coefficient of the d_{pat}-regular function f_{pat} is determined by the rates of elementary cycles used in zigs and zags, the second coefficients reflecting their bounded prefixes and suffixes as well as the rest of the tour obeying that pattern. For a large x, the (main) coefficient is decisive for the "quality"—the size of $f_{pat}(x)$—of a pattern. If more patterns have the same coefficient, their qualitative order for (large) x depends solely on $x \mod d$ where d is, e.g., the multiple of all d_{pat}'s.

In the general case, when order $n > 0$ is allowed, we have boundedly many patterns how to arrange the resets – and use regular path schemes of order 0 in between. So we can use the same idea as above (using also the fact that composition of regular path schemes is regular).

Now call a pair (r, n), $r \in fcs$, $n \in \mathbb{N}$, *increasing* iff for every $m \in \mathbb{N}$ there are $x_1 < x_2 < \cdots < x_{m+1}$ s.t. $(r, p_1 \leftarrow x_1, p_2 \leftarrow 0) \xrightarrow{*}_n (r, p_1 \leftarrow x_2, p_2 \leftarrow 0) \xrightarrow{*}_n \ldots \xrightarrow{*}_n (r, p_1 \leftarrow x_{m+1}, p_2 \leftarrow 0)$. Note that, since R is nondecreasing, there cannot be more than $k = |fcs|$ resets of p in R without any reset of $\bar{p}$ in between. Using the pigenhole principle (at most k times in succession), we can easily show that there is an increasing pair (r, n) s.t. $(r, p_1 \leftarrow -, p_2 \leftarrow 0)$ appears infinitely often in R. Lemma 8 guarantees a regular path scheme (w, f, x_0) of order n which is related to $\langle r, p_2 \leftarrow 0\rangle$, $\langle r, p_2 \leftarrow 0\rangle$ and has the maximum property. f is d-regular for some d; let ρ be its coefficient and $\rho_0, \rho_1, \ldots \rho_{d-1}$ its second coefficients. The case $\rho < 1$ as well as the case $\rho = 1$ and $\rho_i \leq 0$ for some $i \in \{0, 1, \ldots, d-1\}$ would contradict the choice of r, n (for large x and $m > d$). Thus we have shown

Proposition 1. *For an unbounded marked Reset net with at most two reset arcs and without a simple witness, there is a regular witness.*

5 Conclusion

In this paper we investigated boundedness problems for Reset and Transfer nets, after several surprising results appeared in [DFS98].

We proved that boundedness is undecidable for Reset nets with 3 reset arcs and decidable for Reset nets with 2 resetable places. This is quite surprising since 2 resetable places are enough to recognize any recursive language. The proof technique can be adapted to related problems like structural boundedness and structural termination.

We showed structural boundedness is Π_2-complete for Reset nets, Π_1-complete for Transfer nets. This shows that the paradoxical result "the more powerful Transfer nets have simpler boundedness problems" carries over to structural boundedness and is not just a strange accidental quirk.

Results on Reset and Transfer nets (and similar extensions of the P/T model) have applications to various computational models investigated in the infinite systems community, as we pointed out in the Introduction. E.g., our decidability result hints at interesting questions for stack automata (and extensions), lossy systems, parameterized systems.

References

[AK77] T. Araki and T. Kasami. Some decision problems related to the reachability problem for Petri nets. *Theor. Comp. Sci.*, 3(1):85–104, 1977.

[AL78] A. Arnold and M. Latteux. Récursivité et cônes rationnels fermés par intersection. *Calcolo*, XV(IV):381–394, 1978.

[BM99] A. Bouajjani and R. Mayr. Model checking lossy vector addition systems. In *Proc. 16th Ann. Symp. Theoretical Aspects of Computer Science (STACS'99), Trier, Germany, Mar. 1999*, vol. 1563 of *LNCS*, pp 323–333. Springer, 1999.

[BO98] E. Badouel and J. Oliver. Reconfigurable nets, a class of high level Petri nets supporting dynamic changes within workflow systems. Research Report 3339, INRIA, January 1998.

[Cia94] G. Ciardo. Petri nets with marking-dependent arc cardinality: Properties and analysis. In *Proc. 15th Int. Conf. Applications & Theory of Petri Nets, Zaragoza, Spain, June 1994*, vol. 815 of *LNCS*, pp 179–198. Springer, 1994.

[DFS98] C. Dufourd, A. Finkel, and Ph. Schnoebelen. Reset nets between decidability and undecidability. In *Proc. 25th Int. Coll. Automata, Languages, and Programming (ICALP'98), Aalborg, Denmark, July 1998*, vol. 1443 of *LNCS*, pp 103–115. Springer, 1998.

[Duf98] C. Dufourd. Réseaux de Petri avec reset/transfert: Décidabilité et indécidabilité. Thèse de Docteur en Sciences de l'École Normale Supérieure de Cachan, October 1998.

[EFM99] J. Esparza, A. Finkel, and R. Mayr. On the verification of broadcast protocols. In *Proc. 14th IEEE Symp. Logic in Computer Science (LICS'99), Trento, Italy, July 1999*, 1999.

[FS98] A. Finkel and Ph. Schnoebelen. Fundamental structures in well-structured infinite transition systems. In *Proc. 3rd Latin American Theoretical Informatics Symposium (LATIN'98), Campinas, Brazil, Apr. 1998*, vol. 1380 of *LNCS*, pp 102–118. Springer, 1998.

[HP79] J. Hopcroft and J.-J. Pansiot. On the reachability problem for 5-dimensional vector addition systems. *Theor. Comp. Sci.*, 8(2):135–159, 1979.

[KCK+97] M. Kishinevsky, J. Cortadella, A. Kondratyev, L. Lavagno, A. Taubin, and A. Yakovlev. Coupling asynchrony and interrupts: Place chart nets. In *Proc. 18th Int. Conf. Application & Theory of Petri Nets, Toulouse, France, June 1997*, vol. 1248 of *LNCS*, pp 328–347. Springer, 1997.

[LC94] C. Lakos and S. Christensen. A general approach to arc extensions for coloured Petri nets. In *Proc. 15th Int. Conf. Applications & Theory of Petri Nets, Zaragoza, Spain, June 1994*, vol. 815 of *LNCS*, pp 338–357. Springer, 1994.

[May98] R. Mayr. Lossy counter machines. Tech. Report TUM-I9830, Institut für Informatik, TUM, Munich, Germany, October 1998.

[Val78] R. Valk. Self-modifying nets, a natural extension of Petri nets. In *Proc. 5th Int. Coll. Automata, Languages, and Programming (ICALP'78), Udine, Italy, Jul. 1978*, vol. 62 of *LNCS*, pp 464–476. Springer, 1978.