TECHNISCHE
UNIVERSITÄT
MÜNCHEN

INSTITUT FÜR INFORMATIK

Sonderforschungsbereich 342:
Methoden und Werkzeuge für die Nutzung
paralleler Rechnerarchitekturen

Lossy Counter Machines

Richard Mayr

TUM-I9827
SFB-Bericht Nr. 342/10/98 A
Oktober 98

TUM–INFO–10-I9827-50/1.–FI

Anforderungen an: Prof. Dr. A. Bode
Sprecher SFB 342
Institut für Informatik
Technische Universität München
D-80290 München, Germany

Druck: Fakultät für Informatik der
Technischen Universität München

Lossy Counter Machines

Richard Mayr*

Abstract

We consider lossy counter machines, i.e. counter machines with counters whose contents can spontaneously decrease at any time. They are not Turing-powerful, since reachability is decidable for them, but they still have interesting undecidable properties: For a lossy counter machine it is undecidable if there exists an initial configuration s.t. there is an infinite run.

Lossy counter machines can be used as a general tool to show the undecidability of many problems for lossy and non-lossy systems, e.g. verification of lossy FIFO-channel systems, model checking lossy Petri nets, problems for reset and transfer Petri nets, and parametric problems like fairness of broadcast communication protocols.

1 Introduction

Lossy systems were introduced to model communication through unreliable channels. The main example are lossy FIFO-channel systems. These are systems of finite-state processes who communicate through lossy FIFO-channels (buffers) of unbounded length. These lossy FIFO-channels are unreliable, because they can spontaneously loose messages. Since normal (non-lossy) FIFO-channel systems are Turing-powerful, automatic analysis of them is restricted to special cases [4]. Lossy FIFO-channel systems are not Turing-powerful, since reachability and some safety-properties are decidable for them [2, 6, 1]. However, some liveness-properties like the so-called 'recurrent-state problem' are undecidable even for lossy FIFO-channel systems [3].

Here we generalize these negative results by introducing lossy counter machines (LCM). These are counter machines where the numbers in the counters

*Institut für Informatik, TU-München, D-80290 München, Germany, mayrri@informatik.tu-muenchen.de, www7.informatik.tu-muenchen.de/~mayrri

can spontaneously become smaller at any time. We show that it is undecidable for an LCM if there is an initial configuration s.t. it has an infinite run. The result does not depend on the notion of lossiness. For example one can define that a counter can only spontaneously decrease by 1, or only be set to zero, or become any smaller value. This result subsumes the result on lossy FIFO-channel systems in [3], since lossy counter machines are a weaker model (at least as far as infinite runs are concerned). Moreover, our proof is much shorter and simpler.

Lossy counter machines can be used as a general tool to show the undecidability of many problems. This is because there are many systems who cannot simulate counter machines, but who can simulate lossy counter machines.

In Section 2 we define lossy counter machines. In Section 3 we briefly cite some positive decidability results for LCM, while in Section 4 we show the main negative decidability result for LCM. In Section 5 we apply this result to show the undecidability of several verification problems and in Section 6 we draw some general conclusions.

2 Lossy Counter Machines

Definition 2.1 A n-counter machine M is described by a finite set of states Q, an initial state $q_0 \in Q$, an accepting state $accept \in Q$, a rejecting state $reject \in Q$, n counters $c_1, \dots, c_n$ and a finite set of instructions of the form $(q: \ c_i := c_i + 1; \mathsf{goto}\ q')$ or $(q: \ \mathsf{If}\ c_i = 0\ \mathsf{then\ goto}\ q'\ \mathsf{else}\ c_i := c_i - 1; \mathsf{goto}\ q'')$ where $i \in \{1, \dots, n\}$ and $q, q', q'' \in Q$.

A configuration of M is described by a tuple $(q, m_1, \dots, m_n)$ where $q \in Q$ and $m_i \in \mathbb{N}$ is the content of the counter c_i for $1 \leq i \leq n$. The size of a configuration is defined by

$$size((q, m_1, \dots, m_n)) := \sum_{i=1}^{n} m_i$$

The possible computation steps are defined as follows:

1. $(q, m_1, \dots, m_n) \to (q', m_1, \dots, m_i + 1, \dots, m_n)$ if there is an instruction $(q: \ c_i := c_i + 1; \mathsf{goto}\ q')$.
2. $(q, m_1, \dots, m_n) \to (q', m_1, \dots, m_n)$ if there is an instruction $(q: \ \mathsf{If}\ c_i = 0\ \mathsf{then\ goto}\ q'\ \mathsf{else}\ c_i := c_i - 1; \mathsf{goto}\ q'')$ and $m_i = 0$.
3. $(q, m_1, \dots, m_n) \to (q'', m_1, \dots, m_i - 1, \dots, m_n)$ if there is an instruction $(q: \ \mathsf{If}\ c_i = 0\ \mathsf{then\ goto}\ q'\ \mathsf{else}\ c_i := c_i - 1; \mathsf{goto}\ q'')$ and $m_i > 0$.

A *run* of a counter machine is a (possibly infinite) sequence of configurations $s_0, s_1, s_2, \ldots$ with $s_0 \rightarrow s_1 \rightarrow s_2 \rightarrow s_3 \rightarrow \ldots$.

Now we define 'lossiness relations', which describe spontaneous changes in the configurations of lossy counter machines.

Definition 2.2 Let $\stackrel{s}{\rightarrow}$ (for 'sum') be a relation on configurations of n-counter machines which is defined as follows.

$$(q, m_1, \ldots, m_n) \stackrel{s}{\rightarrow} (q', m'_1, \ldots, m'_n) :\Leftrightarrow (q, m_1, \ldots, m_n) = (q', m'_1, \ldots, m'_n) \vee \left(q = q' \wedge \sum_{i=1}^{n} m_i > \sum_{i=1}^{n} m'_i\right)$$

This relation means that either nothing is changed or the sum of all counters strictly decreases. Let *id* be the identity relation. A relation $\stackrel{l}{\rightarrow}$ is a *lossiness relation* iff

$$id \subseteq \stackrel{l}{\rightarrow} \subseteq \stackrel{s}{\rightarrow}$$

A *lossy counter machine* (LCM) is given by a counter machine M and a lossiness relation $\stackrel{l}{\rightarrow}$. Let $\rightarrow$ be the normal transition relation of M. The lossy transition relation $\Longrightarrow$ of the lossy counter machine is defined by

$$s_1 \Longrightarrow s_2 :\Leftrightarrow \exists s'_1 s'_2.\ s_1 \stackrel{l}{\rightarrow} s'_1 \rightarrow s'_2 \stackrel{l}{\rightarrow} s_2$$

An *arbitrary lossy counter machine* is a lossy counter machine with an arbitrary (unspecified) lossiness relation.
The following relations are lossiness relations:

Perfect The relation *id* is a lossiness relation. Thus arbitrary lossy counter machines subsume normal counter machines.

Classic Lossiness The *classic lossiness relation* $\stackrel{cl}{\rightarrow}$ is defined by

$$(q, m_1, \ldots, m_n) \stackrel{cl}{\rightarrow} (q', m'_1, \ldots, m'_n) :\Leftrightarrow q = q' \wedge \forall i.\, m_i \geq m'_i$$

Here the contents of the counters can become smaller in any possible way, but the state q cannot change. A relation $\stackrel{l}{\rightarrow}$ is called a *subclassic lossiness relation* iff $id \subseteq \stackrel{l}{\rightarrow} \subseteq \stackrel{cl}{\rightarrow}$.

Bounded Lossiness A counter can loose at most $x \in \mathbb{N}$ before and after every computation step. Here the lossiness relation $\stackrel{l(x)}{\rightarrow}$ is defined by

$$(q, m_1, \ldots, m_n) \stackrel{l(x)}{\rightarrow} (q', m'_1, \ldots, m'_n) :\Leftrightarrow q = q' \wedge \forall i.\, m_i \geq m'_i \geq max\{0, m_i - x\}$$

Note that $\stackrel{l(x)}{\rightarrow}$ is a subclassic lossiness relation.

Reset Lossiness If a counter is to be tested for zero, then it can suddenly become zero. The lossiness relation $\xrightarrow{rl}$ is defined as follows: $(q, m_1, \ldots, m_n) \xrightarrow{rl} (q', m'_1, \ldots, m'_n)$ iff $q = q'$ and for all i either $m_i = m'_i$ or $m'_i = 0$ and there is an instruction $(q: \ \textsf{If}\ c_i = 0\ \textsf{then goto}\ q'\ \textsf{else}\ c_i := c_i - 1; \textsf{goto}\ q'')$. Note that $\xrightarrow{rl}$ is a subclassic lossiness relation.

The definition of these lossiness relations carries over to other models like Petri nets. There, places are considered instead of counters and the control-states q are ignored.

Definition 2.3 For any arbitrary lossy n-counter machine and any configuration s let $runs(s)$ be the set of runs that start at configuration s. (There can be more than one run if the counter machine is nondeterministic or lossy.) Let $runs^\omega(s)$ be the set of infinite runs that start at configuration s. A run $r = \{(q^i, m_1^i, \ldots, m_n^i)\}_{i=0}^{\infty} \in runs^\omega(s)$ is *space-bounded* iff

$$\exists c \in \mathbb{N}. \forall i. \sum_{j=1}^{n} m_j^i \leq c$$

Let $runs_b^\omega(s)$ be the space-bounded infinite runs that start at s. An arbitrary lossy n-counter machine M is

input-bounded iff in every run from any configuration the size of every reached configuration is bounded by the size of the input.

$$\forall s. \forall r \in runs(s). \forall s' \in r.\, size(s') \leq size(s)$$

strongly-cyclic iff every infinite run from any configuration visits the initial state q_0 infinitely often.

$$\forall q \in Q, m_1, \ldots, m_n \in \mathbb{N}.\ \forall r \in runs^\omega((q, m_1, \ldots, m_n)).$$
$$\exists m'_1, \ldots, m'_n.\, (q_0, m'_1, \ldots, m'_n) \in r$$

bounded-strongly-cyclic iff every space-bounded infinite run from any configuration visits the initial state q_0 infinitely often.

$$\forall q \in Q, m_1, \ldots, m_n \in \mathbb{N}.\ \forall r \in runs_b^\omega((q, m_1, \ldots, m_n)).$$
$$\exists m'_1, \ldots, m'_n.\, (q_0, m'_1, \ldots, m'_n) \in r$$

If M is strongly-cyclic then it is also bounded-strongly-cyclic. If M is input-bounded and bounded-strongly-cyclic then it is also strongly-cyclic.

3 Decidable Properties

Since arbitrary LCM subsume normal counter machines, nothing is decidable for them. However, some problems are decidable for classic LCM. They are not Turing-powerful. The following results are special cases of positive decidability results in [5, 6, 2].

Lemma 3.1 *Let M be a classic LCM and s a configuration of M. The set $pre^*(s) := \{s' \mid s' \Longrightarrow^* s\}$ of predecessors of s is effectively constructible.*

Proof Since M is a classic LCM, the set $pre^*(s)$ is upward closed and can thus be characterized by finitely many minimal elements. These minimal elements can be effectively constructed by Dickson's Lemma (see [5]). ■

Theorem 3.2 *Reachability is decidable for classic LCM.*

Proof Directly from Lemma 3.1. ■

Theorem 3.3 *Let M be a classic LCM with initial configuration s_0. It is decidable if there is an infinite run that starts at s_0.*

Proof Check for all runs if it terminates or it reaches a configuration s s.t. there is a previous configuration s' with $s \geq s'$. There is an infinite cyclic run from s' to s' in the latter case, because M is classical lossy. These checks are effective, because of Dickson's Lemma. ■

4 The Undecidability Result

In this section we show that the following problem is undecidable for every lossiness relation.

$\exists(\alpha, q)$LCM$^\omega$

Instance: A strongly-cyclic, input-bounded LCM M with five counters and initial state q_0.

Question: Do there exist $m_1, \ldots, m_5 \in \mathbb{N}$ and a state $q \in Q$ s.t. there is an infinite run that starts at $(q, m_1, \ldots, m_5)$?

The proof proceeds in three steps:

1. We consider a normal 2-counter machine M. It is undecidable if there exists an $n \in \mathbb{N}$ s.t. M accepts the input $(n, 0)$, i.e. if the computation from the configuration $(q_0, n, 0)$ is accepting.

2. We reduce this problem to the problem $\exists\alpha$BSC-CM$_b^\omega$. This problem is for a bounded-strongly-cyclic 4-counter machine M', if there exist $m_1, \dots, m_4 \in \mathbb{N}$ s.t. there is an infinite space-bounded run from $(q_0, m_1, \dots, m_4)$.

3. We reduce $\exists\alpha$BSC-CM$_b^\omega$ to $\exists(\alpha, q)$LCM$^\omega$ for every lossiness relation.

$\exists n$CM
Instance: A 2-counter machine M with initial state q_0.
Question: Does there exist an $n \in \mathbb{N}$ s.t. M accepts $(q_0, n, 0)$?

Lemma 4.1 *$\exists n CM$ is undecidable.*

Proof By a simple reduction from the problem if a 2-counter machine accepts any input. ■

$\exists\alpha$BSC-CM$_b^\omega$
Instance: A bounded-strongly-cyclic 4-counter machine M with initial state q_0.
Question: Do there exist $m_1, \dots, m_4 \in \mathbb{N}$ s.t. M has an infinite space-bounded run that starts at $(q_0, m_1, m_2, m_3, m_4)$?

Lemma 4.2 *$\exists\alpha BSC$-CM_b^ω is undecidable.*

Proof We reduce $\exists n$CM to $\exists\alpha$BSC-CM$_b^\omega$. Let M be a 2-counter machine with initial state q_0. We construct a 3-counter machine M' that does the same as M, except that after every instruction it increases the third counter c_3 by 1. Every instruction of M of the form $(q: \ c_i := c_i + 1; \ \mathsf{goto}\ q')$ with $(1 \le i \le 2)$ is replaced by

$$\begin{array}{ll} q: & c_i := c_i + 1; \ \mathsf{goto}\ q_2 \\ q_2: & c_3 := c_3 + 1; \ \mathsf{goto}\ q' \end{array}$$

where q_2 is a new state. Every instruction of the form

$$(q: \ \mathsf{If}\ c_i = 0\ \mathsf{then\ goto}\ q'\ \mathsf{else}\ c_i := c_i - 1; \mathsf{goto}\ q'')$$

with $(1 \le i \le 2)$ is replaced by

$$\begin{array}{ll} q: & \mathsf{If}\ c_i = 0\ \mathsf{then\ goto}\ q_2\ \mathsf{else}\ c_i := c_i - 1; \mathsf{goto}\ q_3 \\ q_2: & c_3 := c_3 + 1; \ \mathsf{goto}\ q' \\ q_3: & c_3 := c_3 + 1; \ \mathsf{goto}\ q'' \end{array}$$

where q_2, q_3 are new states. Then we construct a 4-counter machine M'' with initial state q_0'' as follows: First copy c_4 to c_1 and set c_2 and c_3 to zero. ($c_1 := c_4; c_2 := 0; c_3 := 0;$) Then do the same as M'. Finally, we replace the accepting state *accept* of M, M' by q_0'', i.e. we replace every instruction (goto *accept*) by ($\mathsf{goto}\ q_0''$). M'' is bounded-strongly-cyclic, because c_3 is increased after every instruction and only set to zero at the initial state q_0''.

$\Rightarrow$ If M is a positive instance of $\exists n$CM then there exists an $n \in \mathbb{N}$ s.t. M has an accepting run from $(q_0, n, 0)$. This run has finite length and is therefore space-bounded. Then M'' has an infinite space-bounded cyclic run that starts at $(q_0'', 0, 0, 0, n)$ (or $(q_0'', m_1, m_2, m_3, n)$ for any $m_1, m_2, m_3 \in \mathbb{N}$) and thus M'' is a positive instance of $\exists\alpha$BSC-CM$_b^\omega$.

$\Leftarrow$ If M'' is a positive instance of $\exists\alpha$BSC-CM$_b^\omega$ then there exist $m_1, \dots, m_4 \in \mathbb{N}$ s.t. M'' has an infinite space-bounded run that starts at the configuration $(q_0'', m_1, \dots, m_4)$. By the construction of M'' it also has an infinite space-bounded run that starts at $(q_0'', 0, 0, 0, m_4)$. Since M'' is bounded-strongly-cyclic this run must visit q_0'' again. By the construction of M'' this is only possible if the included computation of M reaches the accepting state. Then M has an accepting run that starts at $(q_0, m_4, 0)$, and thus M is a positive instance of $\exists n$CM.

The result follows from Lemma 4.1. ∎

Now we can prove the main result.

Theorem 4.3 *$\exists(\alpha, q)LCM^\omega$ is undecidable for every lossiness relation.*

Proof We reduce $\exists\alpha$BSC-CM$_b^\omega$ to $\exists(\alpha, q)$LCM$^\omega$ with any lossiness relation $\xrightarrow{l}$. For any bounded-strongly-cyclic 4-counter machine M we construct a strongly-cyclic, input-bounded lossy 5-counter machine M' with lossiness relation $\xrightarrow{l}$ as follows: The fifth counter c_5 holds the 'capacity'. In every operation it is changed in a way s.t. the sum of all counters never increases. (More exactly, the sum of all counters can increase by 1, but only if it was decreased by 1 in the previous step.) Every instruction of M of the form ($q: \ c_i := c_i + 1;\ \mathsf{goto}\ q'$) with $(1 \le i \le 4)$ is replaced by

$$\begin{array}{ll} q: & \mathsf{If}\ c_5 = 0\ \mathsf{then\ goto}\ \mathit{halt}\ \mathsf{else}\ c_5 := c_5 - 1;\ \mathsf{goto}\ q_2 \\ q_2: & c_i := c_i + 1;\ \mathsf{goto}\ q' \end{array}$$

where *halt* is a final state and q_2 is a new state. Every instruction of the form ($q: \ \mathsf{If}\ c_i = 0\ \mathsf{then\ goto}\ q'\ \mathsf{else}\ c_i := c_i - 1; \mathsf{goto}\ q''$) with $(1 \le i \le 4)$ is replaced by

$$\begin{array}{ll} q: & \mathsf{If}\ c_i = 0\ \mathsf{then\ goto}\ q'\ \mathsf{else}\ c_i := c_i - 1; \mathsf{goto}\ q_2 \\ q_2: & c_5 := c_5 + 1;\ \mathsf{goto}\ q'' \end{array}$$

where q_2 is a new state.

M' is bounded-strongly-cyclic, because M is bounded-strongly-cyclic. M' is input-bounded, because every run from a configuration $(q, m_1, \dots, m_5)$ is space-bounded by $m_1 + m_2 + m_3 + m_4 + m_5$. Thus M' is also strongly-cyclic.

$\Rightarrow$ If M is a positive instance of $\exists\alpha$BSC-CM$_b^\omega$ then there exist $m_1, \dots, m_4, c \in \mathbb{N}$ s.t. there is an infinite run that starts at $(q_0, m_1, \dots, m_4)$, visits q_0 infinitely often and always satisfies $c_1 + c_2 + c_3 + c_4 \leq c$. Since $id \subseteq \xrightarrow{l}$, there is also an infinite run of M' that starts at $(q_0, m_1, m_2, m_3, m_4, c - m_1 - m_2 - m_3 - m_4)$, visits q_0 infinitely often and always satisfies $c_1 + c_2 + c_3 + c_4 + c_5 \leq c$. Thus M' is a positive instance of $\exists(\alpha, q)$LCM$^\omega$.

$\Leftarrow$ If M' is a positive instance of $\exists(\alpha, q)$LCM$^\omega$ then there exist $m_1, \dots, m_5 \in \mathbb{N}$ and $q \in Q$ s.t. there is an infinite run that starts at the configuration $(q, m_1, m_2, m_3, m_4, m_5)$. This run is space-bounded, because it always satisfies $c_1 + c_2 + c_3 + c_4 + c_5 \leq m_1 + m_2 + m_3 + m_4 + m_5$. By the construction of M', the sum of all counters can only increase by 1 if it was decreased by 1 in the previous step. By the definition of lossiness (see Def. 2.2) we get the following: If lossiness occurs (when the contents of the counters spontaneously change) then this strictly and permanently decreases the sum of all counters. It follows that lossiness can only occur at most $m_1 + m_2 + m_3 + m_4 + m_5$ times in this infinite run and the sum of all counters is bounded by $c := m_1 + m_2 + m_3 + m_4 + m_5$. Thus there is an infinite suffix of this run where lossiness does not occur. Thus there exist $q' \in Q$, $m_1', \dots, m_5' \in \mathbb{N}$ s.t. an infinite suffix of this run without lossiness starts at $(q', m_1', \dots, m_5')$. It follows that there is an infinite space-bounded run of M that starts at $(q', m_1', \dots, m_4')$. Since M is bounded-strongly-cyclic, this run must eventually visit q_0. Thus there exist $m_1'', \dots, m_4'' \in \mathbb{N}$ s.t. there is an infinite space-bounded run of M that starts at $(q_0, m_1'', \dots, m_4'')$. Thus M is a positive instance of $\exists\alpha$BSC-CM$_b^\omega$.

It follows from Lemma 4.2 that $\exists(\alpha, q)$LCM$^\omega$ is undecidable. ■

It is interesting to note that the undecidability result does not depend on the lossiness relation. It holds for any lossiness relation. Another variant of this problem is the following:

$\exists\alpha$LCM$^\omega$

Instance: A strongly-cyclic, input-bounded LCM M with five counters and initial state q_0.

Question: Do there exist $m_1, \ldots, m_5 \in \mathbb{N}$ s.t. there is an infinite run that starts at $(q_0, m_1, \ldots, m_5)$?

Theorem 4.4 *$\exists\alpha LCM^\omega$ is undecidable for every lossiness relation.*

Proof Directly from Theorem 4.3, because the LCM is strongly-cyclic. ■

Another variant is the following.

$\exists n$LCM$^\omega$

Instance: An input-bounded nondeterministic lossy counter machine M with five counters and initial state q_0.

Question: Does there exist a number $n \in \mathbb{N}$ s.t. there is an infinite run that starts at $(q_0, n, 0, 0, 0, 0)$?

Theorem 4.5 *$\exists nLCM^\omega$ is undecidable for every lossiness relation.*

Proof We reduce $\exists\alpha$LCM$^\omega$ to $\exists n$LCM$^\omega$. Let M with initial state q_0 be an instance of $\exists\alpha$LCM$^\omega$. We construct a new LCM M' with initial state q_0' as follows. It starts at $(q_0', n, 0, 0, 0, 0)$. and can (nondeterministically) go to any new configuration $(q_0, m_1, m_2, m_3, m_4, m_5)$ with $m_1+m_2+m_3+m_4+m_5 \leq n$. Then it behaves just like M. It follows that M is a positive instance of $\exists\alpha$LCM$^\omega$ iff M' is a positive instance of $\exists n$LCM$^\omega$. ■

5 Applications

We show how lossy counter machines can be used to prove the undecidability of several problems.

5.1 Lossy FIFO-Channel Systems

In [3] it was shown that it is undecidable if there exists an initial configuration of a lossy FIFO-channel system s.t. it has an infinite run. The lossiness relation in [3] was classic lossiness, i.e. the contents of a FIFO-channel can change to any substring at any time. The results in Section 4 subsume this result, since lossy counter machines are weaker than lossy FIFO-channel systems. A lossy FIFO-channel system can simulate a LCM (with additional deadlocks) in the following way: Every lossy FIFO-channel contains a string in X^* (for some symbol X) and is used as a lossy counter. The only problem is the test for zero. We test the emptiness of a channel by adding a special symbol Y and removing it in the very next step. If it can be done then the channel is empty (or has become empty by lossiness). If this cannot be done, then the channel was not empty or the symbol Y was lost. In this case we get a deadlock. These additional deadlocks do not affect the existence of infinite runs, and thus the results of Section 4 carry over.

5.2 Model Checking Lossy BPP

Basic Parallel Processes (BPP) [7] correspond to communication-free nets, the subclass of labeled Petri nets where every transition has exactly one place in its preset. The branching-time temporal logics EF, EG and EG_ω are defined as extensions of Hennessy-Milner Logic by the operators EF, EG and EG_ω, respectively. $s \models EF\varphi$ iff there exists an s' s.t. $s \xrightarrow{*} s'$ and $s' \models \varphi$. $s_0 \models EG_\omega\varphi$ iff there exists an infinite run $s_0 \to s_1 \to s_2 \to \ldots$ s.t. $\forall i.\, s_i \models \varphi$. EG is similar, except that it also includes finite runs that end in a deadlock. Model checking Petri nets with EF is undecidable, but model checking BPP with EF is $PSPACE$-complete [11]. Model checking BPP with EG is undecidable [10]. It is different for lossy systems: By induction on the nesting-depth of the operators EF, EG and EG_ω, and constructions similar to the ones in Lemma 3.1 and Theorem 3.3, it can be shown that model checking classic LCM with the logics EF, EG and EG_ω is decidable. Thus it is also decidable for classical lossy Petri nets and classical lossy BPP (see also [5]). However, model checking lossy BPP with nested EF and EG operators is still undecidable for every subclassic lossiness relation.

Theorem 5.1 *Model checking lossy BPP (with any subclassic lossiness relation) with formulae of the form $EFEG_\omega\Phi$, where Φ is a Hennessy-Milner Logic formula, is undecidable.*

Proof Esparza and Kiehn showed in [10] that for every counter machine M (with all counters initially 0) a BPP P and a Hennessy-Milner Logic formula φ can be constructed s.t. M does not halt iff $P \models EG_\omega\varphi$. The construction carries over to subclassic LCM and subclassic lossy BPP. The control-states of the counter machine are modeled by special places of the BPP. In every infinite run that satisfies φ exactly one of these places is marked at any time. We reduce $\exists n\text{LCM}^\omega$ to the model checking problem. Let M be a subclassic LCM. Let P be the corresponding BPP as in [10] and let φ be the corresponding Hennessy-Milner Logic formula as in [10]. We use the same subclassic lossiness relation on M and on P. P stores the contents of the first counter in a place Y. Thus $P\|Y^n$ corresponds to the configuration of M with n in the first counter (and 0 in the others). We define a new initial state X and transitions $X \stackrel{a}{\to} X\|Y$ and $X \stackrel{b}{\to} P$, where a and b do not occur in P. Let $\Phi := \varphi \wedge \neg\langle b\rangle true$. Then M is a positive instance of $\exists n$LCM$^\omega$ iff $X \models EFEG_\omega\Phi$. The result follows from Theorem 4.5. ■

This result is quite surprising, since lossy BPP is an extremely weak model for concurrent systems. The same model checking problem was shown to be undecidable for classical lossy FIFO-channel systems by Abdulla and Jonsson in [3]. Theorem 5.1 subsumes this result, since classical lossy BPP are a weaker model than classical lossy FIFO-channel systems.

Corollary 5.2 *Model checking lossy Petri nets with CTL is undecidable for every subclassic lossiness relation.*

Proof Directly from Theorem 5.1, because BPP is a subclass of Petri nets and $EFEG_\omega\varphi$ can be expressed in CTL. ■

Remark 5.3 *For Petri nets and BPP, the meaning of Hennessy-Milner Logic formulae can be expressed by boolean combinations of constraints of the form $p \geq k$, which mean that there are at least k tokens on place p. Thus Theorem 5.1 and Corollary 5.2 also hold if boolean combinations of such constraints are used instead of Hennessy-Milner Logic formulae.*

5.3 Reset/Transfer Petri Nets

It was shown in [8] that termination is decidable for 'Reset Post G-nets', an extension of Petri nets that subsumes reset nets and transfer nets. For normal Petri nets termination is *EXPSPACE*-complete [12]. Now we consider structural termination, i.e. the problem if the net terminates for every initial marking. The negation of this problem is the question if there exists an initial marking s.t. there is an infinite run. Structural termination is decidable in polynomial time for normal Petri nets. (Just check if there is a positive linear combination of effects of transitions.) However, we show that structural termination is undecidable for reset nets and transfer nets.

Theorem 5.4 *Structural termination is undecidable for lossy reset nets and transfer nets for every subclassic lossiness relation.*

Proof We want to simulate a lossy counter machine by a lossy reset net. Let $\xrightarrow{l}$ be the subclassic lossiness relation for the reset net. Since the control-states of the counter machine will be simulated by special places of the reset net we consider a new lossiness relation $\xrightarrow{l'}$ on the counter machine that does the same as $\xrightarrow{l}$ on the reset net. Then we use the lossiness relation $\xrightarrow{l'} \cup \xrightarrow{rl}$ (with $\xrightarrow{rl}$ from Definition 2.2) as the lossiness relation for counter machines. We reduce $\exists(\alpha, q)\text{LCM}^\omega$ (with $\xrightarrow{l'} \cup \xrightarrow{rl}$) to the structural termination problem for lossy reset nets (with $\xrightarrow{l}$). For every LCM M we construct a reset net N in the following way. Let there be places c_1, c_2, c_3, c_4, c_5 that hold the contents of the counters and a place q for every state $q \in Q$ of the finite control of M. For every instruction of M of the form $(q: \ c_i := c_i + 1;\ \textsf{goto}\ q')$ with $(1 \le i \le 5)$ there is a transition that takes one token from q, puts one token on c_i, puts one token on q' and resets all places except $q', c_1, \dots, c_5$. For every instruction of M of the form $(q: \ \textsf{If}\ c_i = 0\ \textsf{then goto}\ q'\ \textsf{else}\ c_i := c_i - 1; \textsf{goto}\ q'')$ with $(1 \le i \le 5)$ there are two transitions: The first transition takes a token from q, puts a token on q' and resets c_i and all places except $q'', c_1, \dots, c_5$. The second transition takes one token from q and one from c_i, puts one token on q'' and resets all places except $q'', c_1, \dots, c_5$. A run of this net is a faithful simulation of the lossy counter machine M, because the lossiness relation of M includes $\xrightarrow{rl}$. (Instead of testing for zero we can reset a place/counter to zero.)

$\Rightarrow$ If M is a positive instance of $\exists(\alpha, q)\text{LCM}^\omega$ then there are $q \in Q$ and $m_i \in \mathbb{N}$ for $1 \le i \le 5$ s.t. an infinite run of M starts at $(q, m_1, \dots, m_5)$. Thus an infinite run of N starts at the marking that has m_i tokens on place c_i (for $1 \le i \le 5$), one token on q and zero tokens on any other place q'. Thus N is not structurally terminating.

$\Leftarrow$ If N is not structurally terminating then there exists an initial marking Σ with an infinite run. The first transition of this run takes a token from exactly one of the places q that correspond to the states of the finite control of M. Let $m_i := \Sigma(c_i)$ for $1 \le i \le 5$. Then M has an infinite run that starts at $(q, m_1, \dots, m_5)$ and thus M is a positive instance of $\exists(\alpha, q)\text{LCM}^\omega$.

The proof for transfer nets is similar. Instead of resetting places to zero, the tokens are moved to a special dead place. Theorem 4.3 yields the result. ■

	Petri nets	reset/transfer nets
Termination	*EXPSPACE*-complete	decidable
Structural termination	$\in \mathcal{P}$	undecidable

Now we consider structural boundedness and structural place-boundedness. This is the problem if the whole net or a certain place p is bounded for every initial marking. For normal Petri nets this is decidable. Just check if there is a linear combination of the effects of transitions that is > 0 on some place/on place p. This does not hold for reset nets and transfer nets.

Theorem 5.5 *Structural boundedness and structural place-boundedness is undecidable for lossy reset nets and transfer nets for every subclassic lossiness relation.*

Proof Like in Theorem 5.4 we consider the subclassic lossiness relation $\xrightarrow{l}$ for the reset net, and $\xrightarrow{l'} \cup \xrightarrow{rl}$ for the LCM. For every LCM M we construct the reset net N as in Theorem 5.4. Then we add a new place P and arcs from every transition to P. Let the new net be N'.

$\Rightarrow$ If M is a positive instance of $\exists(\alpha, q)\mathrm{LCM}^\omega$ (with $\xrightarrow{l'} \cup \xrightarrow{rl}$) then there is an initial marking of N' s.t. there is an infinite run in which P is unbounded. (Choose a run where nothing is lost from P. This is possible, because $id \subseteq \xrightarrow{l}$.) Thus P is not structurally bounded and thus N' is not structurally bounded.

$\Leftarrow$ If N' is not structurally bounded (on place P or any other place), then there exists an initial marking of N' s.t. there is an infinite run. Since N' is a faithful simulation of M there is also an initial configuration of M with an infinite run. Thus M is a positive instance of $\exists(\alpha, q)\mathrm{LCM}^\omega$.

The result follows from Theorem 4.3. The proof for transfer nets is similar. ■

5.4 Parametric Problems

We consider verification problems for systems whose definition includes a parameter $n \in \mathbb{N}$. Intuitively, n can be seen as the size of the system. Examples are

- Systems of n communicating finite-state processes.
- Systems of communicating pushdown automata with n-bounded stack.
- Systems of (a fixed number of) processes who communicate through (lossy) buffers or queues of size n.

Let $P(n)$ be such a system with parameter n. For every fixed n, $P(n)$ is a finite-state system and thus every verification problem is decidable for it. So the problem $P(n) \models \Phi$ is decidable for any temporal logic formula Φ. The parametric verification problem is if a property holds independently of the parameter n, i.e. for any size. Formally, the question is if for given P and Φ we have $\forall n \in \mathbb{N}. P(n) \models \Phi$ (or $\neg\exists n \in \mathbb{N}. P(n) \models \neg\Phi$). Many of these parametric problems are undecidable by the following meta-theorem.

Theorem 5.6 *A parametric verification problem is undecidable if it satisfies the following conditions:*

1. *It can encode an n-space-bounded nondeterministic lossy counter machine (for some lossiness relation) in such a way that $P(n)$ corresponds to the initial configuration with n in the first counter and 0 in the others.*

2. *It can check for the existence of an infinite run.*

Proof By a reduction of $\exists n\mathrm{LCM}^\omega$ and Theorem 4.5. ■

The technique of Theorem 5.6 is used in [9] to show the undecidability of the model checking problem for linear-time temporal logic (LTL) and broadcast communication protocols. These are systems of n communicating processes where a 'broadcast' by one process can affect all other $n-1$ processes. Such a broadcast can be used to set a simulated counter to zero. However, there is no test for zero. One reduces $\exists n\mathrm{LCM}^\omega$ with lossiness relation $\xrightarrow{rl}$ to the model checking problem.

6 Conclusion

While the addition of lossiness to systems makes some verification problems decidable, this extends not very far. Some only slightly more complex verification problems are still undecidable even for lossy systems (see especially Subsection 5.2).

Lossy counter machines can be used as a general tool to show the undecidability of many verification problems for lossy and non-lossy systems. We suspect that many more problems can be shown to be undecidable with the help of lossy counter machines, especially in the area of parametric problems (see Subsection 5.4).

Acknowledgment: Many thanks to Javier Esparza for fruitful discussions.

References

[1] P. Abdulla, A. Bouajjani, and B. Jonsson. On-the-fly Analysis of Systems with Unbounded, Lossy Fifo Channels. In *10th Intern. Conf. on Computer Aided Verification (CAV'98)*. LNCS 1427, 1998.

[2] P. Abdulla and B. Jonsson. Verifying Programs with Unreliable Channels. In *LICS'93*. IEEE, 1993.

[3] P. Abdulla and B. Jonsson. Undecidable verification problems for programs with unreliable channels. *Information and Computation*, 130(1):71–90, 1996.

[4] A. Bouajjani and P. Habermehl. Symbolic reachability analysis of FIFO-channel systems with nonregular sets of configurations. In *Proc. of ICALP'97*, volume 1256 of *LNCS*, 1997.

[5] A. Bouajjani and R. Mayr. Model checking lossy vector addition systems. In *Proc. of STACS'99*, LNCS. Springer Verlag, 1999. to appear.

[6] G. Cécé, A. Finkel, and S.P. Iyer. Unreliable Channels Are Easier to Verify Than Perfect Channels. *Information and Computation*, 124(1):20–31, 1996.

[7] S. Christensen. *Decidability and Decomposition in Process Algebras*. PhD thesis, Edinburgh University, 1993.

[8] C. Dufourd, A. Finkel, and Ph. Schnoebelen. Reset nets between decidability and undecidability. In *Proc. of ICALP'98*, volume 1443 of *LNCS*. Springer Verlag, 1998.

[9] J. Esparza, A. Finkel, and R. Mayr. On broadcast protocols. Submitted to LICS'99, 1999.

[10] J. Esparza and A. Kiehn. On the model checking problem for branching time logics and Basic Parallel Processes. In *CAV'95*, volume 939 of *LNCS*, pages 353–366. Springer Verlag, 1995.

[11] R. Mayr. Weak bisimulation and model checking for Basic Parallel Processes. In *Foundations of Software Technology and Theoretical Computer Science (FST&TCS'96)*, volume 1180 of *LNCS*. Springer Verlag, 1996.

[12] H. Yen. A unified approach for deciding the existence of certain Petri net paths. *Information and Computation*, 96(1):119–137, 1992.