

Deciding Reachability and Coverability in Lossy EOS

Francesco Di Cosmo¹, Soumodev Mal² and Tephilla Prince³

¹Free University of Bozen-Bolzano, Italy

²Chennai Mathematical Institute, India

³IIT Dharwad, India

Abstract

Elementary Object Systems (EOSs) are a representative of the nets-within-nets paradigm, where the places of a system Petri Net (PN) can also host object nets in place of standard black tokens. Recently, EOSs were put forward to naturally capture the notion of break-downs, under the let-it-crash approach, even when lacking any extra domain knowledge: break-downs can affect only the computational units modeled by object nets. The break-downs are represented by removing all tokens from the internal markings of the object nets, so as to enforce a deadlock. In this setting, one can check whether a property is robust, i.e., if it holds even in front of at most a given number of break-downs (possibly infinitely many). The approach of induced deadlocks is reminiscent of lossy Petri Nets, where some token (possibly all) may be non-deterministically lost. The partial loss of a marking can be seen as a more fine-grained variant of break-down, where the EOS components partially degrade. However, checking the decidability of reachability/coverability for EOSs with lossy steps has not been charted yet. In this paper, we fully chart the decidability status of reachability and coverability in front of various forms of lossiness: only at the object net level, only at the system net level, or at both. We do that for both EOSs and the fragment of conservative EOSs. Our results show that almost all the studied problems are undecidable. The only decidable cases regard conservative EOSs in front of any given positive number of losses and EOSs in front of arbitrarily many losses, both at the system and object net level. The latter result is especially interesting, since finitely many losses still result in undecidability.

Keywords

Nets-within-nets, Coverability, Reachability, Elementary Object Systems, Robustness

1. Introduction

Imperfections are natural in the context of message passing systems: imperfect communication channels may spontaneously lose, duplicate, or shuffle carried messages or even deliver new unwanted ones. When compared to their perfect counterpart, verification of imperfect systems is usually easier. For example, reachability in Communicating Finite Automata (CFA) over *perfect* (FIFO) channels is undecidable [1], but it is decidable when the channels are *lossy* [2] (messages may be non-deterministically lost and never delivered) or *unordered* [3] (the message sending and reception order may be different). The same principle generally holds even outside CFA. Specifically, two counter machines can encode Turing Machines [4, 5] and suffer from undecidable reachability. Instead, Mayr [6, 7] showed that if lossiness is applied to the counters (the natural numbers stored in the counters may non-deterministically decrease), then reachability becomes decidable; nevertheless, several other problems remain undecidable. Similar results

PNSE'24, International Workshop on Petri Nets and Software Engineering, 2024

✉ frdicosmo@unibz.it (F. Di Cosmo); soumodevmal@cmi.ac.in (S. Mal); tephilla.prince.18@iitdh.ac.in (T. Prince)

🆔 0000-0002-5692-5681 (F. Di Cosmo); 0000-0001-5054-5664 (S. Mal); 0000-0002-1045-3033 (T. Prince)

© 2024 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

are also available for Petri Nets (PNs): Bouajjani and Mayr [8] studied the impact of lossiness on the model checking problem of Vector Addition Systems with States (VASS, equivalent to Petri Nets) against fragments of the Modal μ -Calculus. In this case, while the EF (and EG) fragment of the UB language (which includes negation and labeled variants of the CTL next operator) is undecidable for VASS [9], it is decidable for lossy VASS.

Imperfections may be naturally interpreted as perturbations of the system configurations and, thus, verification of imperfect systems can be studied under the lens of robustness [10, 11], i.e., checking whether a property holds if at most k perturbations occur, for some given $k \leq |\mathbb{N}|$. Recently, Köhler-Bussmeier and Capra [11] put forward the nets-within-nets paradigm [12] to naturally specify robustness properties in a Multi Agent System (MAS) context. Specifically, in the PN setting, *black tokens* that mark a fixed set of places are moved around by a fixed set of transitions. In contrast, in nets-within-nets, the tokens could additionally be PN objects themselves. Thus, even without further expert information on the net design, these *object tokens* naturally model agents, which might be affected by perturbations. In [11], perturbations follow a drastic *let it crash* approach, causing agent break-downs. Technically, this is achieved by enforcing a deadlock in the object token. However, nets-within-nets with less disruptive perturbations may still be suitable to model perturbed MAS, where the agents may suffer imperfections even without completely breaking down.

In this paper, we lay the foundations for our ongoing project on formal verification of imperfect nets-within-nets. We focus on Elementary Object Systems (EOSs) [13], which are a simple nets-within-nets model, yet featuring most of the important ingredients. Moreover, they can be generalized to more sophisticated models, such as full-fledged Object Systems [12]. This makes EOSs an excellent candidate for our study. Our key contributions are:

1. We formally define three forms of lossiness in the EOS setting, corresponding to the nesting levels of the tokens.
2. We provide examples illustrating the relevance of these lossiness relations and formalize lossy reachability/coverability problems on them.
3. We completely chart the decidability status of these problems (see Tab. 1).

Standard reachability/coverability problems have been studied in [13], but only in a perfect setting, i.e., without lossiness. Preliminary results on EOS robustness were put forward in [11]. However, they do not address the problem of perturbations on reachability/coverability in a systematic way. To the best of our knowledge, ours is the first work that attempts a full formal classification of EOS reachability/coverability with lossy perturbations. As discussed above, the concept of lossiness is well established in the PN literature; the most relevant works are [8, 7, 14]. However, their approach uniformly interleaves each standard step with lossiness and, thus, does not properly address robustness, where also the number of lossy events is of interest. Moreover, when compared to PNs, EOSs offer more room for lossiness because of the nesting of tokens. This may significantly complicate the picture. Instead, we consider the full spectrum of lossiness, from one occurrence to infinitely many, at all nesting levels.

The paper outline is as follows. Preliminaries, including EOSs, are in Sec. 2. Lossy EOS relations and problems are in Sec. 3. In Sec. 4 and Sec. 5 we study theoretical aspects of lossy reachability/coverability problems. In Sec. 6, Sec. 7, and Sec. 8, we prove several decidability/undecidability results for reachability/coverability. We draw our conclusions in Sec. 9.

2. Preliminaries

2.1. Binary Relations

Let us fix some notation for binary relations. Given a set X , the *identity relation* id_X on X is the relation $\{(x, x) \in X^2 \mid x \in X\}$. Given a binary relation $<$ on X , we denote its *reflexive closure* $< \cup id_X$ by \leq and its *anti-reflexive part* $< \setminus id_X$ by \prec . We use the symbol $>$ to denote the relation such that, $x > y$ iff $y < x$. For example, if $<$ is transitive, then \leq and \geq are transitive and reflexive (i.e., quasi orders). The same applies to the symbols $<$ and $>$ and their closures. From now on, we use $<$ to denote arbitrary transitive relations, and $<$ (possibly with a subscript) to represent fixed transitive relations, e.g., the standard order of \mathbb{N} .

2.2. Multisets

A *multiset* \mathbf{m} on a set D is a mapping $\mathbf{m} : D \rightarrow \mathbb{N}$. The *support* of \mathbf{m} is the set $\text{Supp}(\mathbf{m}) = \{i \mid \mathbf{m}(i) > 0\}$. The multiset \mathbf{m} is finite if its $\text{Supp}(\mathbf{m})$ is finite. The family of all multisets over D is denoted by D^\oplus . We denote a finite multiset \mathbf{m} by enumerating the elements $d \in \text{Supp}(\mathbf{m})$ exactly $\mathbf{m}(d)$ times in between $\{\{$ and $\}\}$, where the ordering is irrelevant. For example, the finite multiset $\mathbf{m} : \{p, q\} \rightarrow \mathbb{N}$ such that $\mathbf{m}(p) = 1$ and $\mathbf{m}(q) = 2$ is denoted by $\{\{p, q, q\}\}$. The empty multiset $\{\{\}\}$ (with empty support) is also denoted by \emptyset . On the empty domain $D = \emptyset$ the only defined multiset is \emptyset ; to stress this out we denote this special case, i.e., the empty multiset over the empty domain, by ε . Given two multisets \mathbf{m}_1 and \mathbf{m}_2 on D , we define $\mathbf{m}_1 + \mathbf{m}_2$ and $\mathbf{m}_1 - \mathbf{m}_2$ on D as follows: $(\mathbf{m}_1 + \mathbf{m}_2)(d) = \mathbf{m}_1(d) + \mathbf{m}_2(d)$ and $(\mathbf{m}_1 - \mathbf{m}_2)(d) = \max(\mathbf{m}_1(d) - \mathbf{m}_2(d), 0)$. Similarly, for a finite set I of indices, $\sum_{i \in I} \{\{d_i\}\}$ denotes the multiset \mathbf{m} over $\bigcup_{i \in I} \{d_i\}$ such that $\mathbf{m}(d) = |\{i \in I \mid d_i = d\}|$ for each $d \in D$. With a slight abuse of notation, we omit the double brackets, i.e., $\sum_{i \in I} \{\{d_i\}\} = \sum_{i \in I} d_i$. If $I = \{1, \dots, n\}$, then $\sum_{i \in I} d_i = \sum_{i=1}^n d_i$. Finally, we write $\mathbf{m}_1 \sqsubseteq \mathbf{m}_2$ if, for each $d \in D$, we have $\mathbf{m}_1(d) \leq \mathbf{m}_2(d)$.

2.3. Petri Nets

We assume that the reader is familiar with standard PNs. Here we just fix the notation (see, e.g., [15]). We denote a PN N as a tuple $N = (P, T, F)$, where P is a finite place set, T is a finite transition set, and F is a flow function. Where useful, we equivalently interpret F via the functions $\text{pre}_N : T \rightarrow (P \rightarrow \mathbb{N})$ where $\text{pre}_N(t)(p) = F(p, t)$ and $\text{post}_N : T \rightarrow (P \rightarrow \mathbb{N})$ where $\text{post}_N(t)(p) = F(t, p)$. For example, a transition $t \in T$ is enabled on a marking μ (finite multiset of places) if, for each place $p \in P$, we have $\text{pre}_N(t)(p) \leq \mu(p)$. Its firing results in the marking μ' such that $\mu'(p) = \mu(p) - \text{pre}_N(t)(p) + \text{post}_N(t)(p)$, for each $p \in P$. We denote markings according to multiset notation. For example, the marking μ that places one token on place p and two on place q is denoted by $\{\{p, q, q\}\}$. The empty marking is denoted by \emptyset . We also work with the special *empty PN* $\blacksquare = (\emptyset, \emptyset, \emptyset)$, whose only marking is ε .

2.4. Elementary Object Systems

An EOS [13] is, intuitively, a PN (called system net) whose tokens carry an internal PN (called object net), taken from a finite set \mathcal{N} . Each place can host only one fixed type of internal PN. The EOS fires events, which synchronize a transition τ in the system net and multisets $\theta(N)$ of transitions in the object nets N consumed by τ .

Definition 1 (EOS). An EOS \mathfrak{E} is a tuple $\mathfrak{E} = \langle \hat{N}, \mathcal{N}, d, \Theta \rangle$ where:

1. $\hat{N} = \langle \hat{P}, \hat{T}, \hat{F} \rangle$ is a PN called *system net*; \hat{T} contains a special set $ID_{\hat{P}} = \{id_p \mid p \in \hat{P}\} \subseteq \hat{T}$ of *idle transitions* such that, for each distinct $p, q \in \hat{P}$, we have $\hat{F}(p, id_p) = \hat{F}(id_p, p) = 1$ and $\hat{F}(q, id_p) = \hat{F}(id_p, q) = 0$.
2. \mathcal{N} is a finite set of PNs, called *object PNs*, such that $\blacksquare \in \mathcal{N}$ and if $(P_1, T_1, F_1), (P_2, T_2, F_2) \in \mathcal{N} \cup \hat{N}$ ¹ then $P_1 \cap P_2 = \emptyset$ and $T_1 \cap T_2 = \emptyset$.
3. $d : \hat{P} \rightarrow \mathcal{N}$ is called the *typing function*.
4. Θ is a finite set of events where each event is a pair $(\hat{\tau}, \theta)$, where $\hat{\tau} \in \hat{T}$ and $\theta : \mathcal{N} \rightarrow \bigcup_{(P,T,F) \in \mathcal{N}} T^\oplus$, such that $\theta((P, T, F)) \in T^\oplus$ for each $(P, T, F) \in \mathcal{N}$ and, if $\hat{\tau} = id_p$, then $\theta(d(p)) \neq \emptyset$.

Since the nets in \mathcal{N} are disjoint, we can denote each event $\langle \hat{\tau}, \theta \rangle$, as a pair $\langle \hat{\tau}, M \rangle$ for a multiset M over $\bigcup_{(P,T,F) \in \mathcal{N}} T$ such that $M(t) = \theta(N)(t)$ where $N = (P, T, F) \in \mathcal{N}$ and $t \in T$. EOS tokens are nested, i.e., each token at a system place p carries a PN marking μ for the object net $d(p)$. EOS markings, also called nested markings, are multisets of nested tokens. With a slight abuse of notation, we denote markings omitting double curly brackets from multiset notation.

Definition 2 (Nested Markings). Let $\mathfrak{E} = \langle \hat{N}, \mathcal{N}, d, \Theta \rangle$ be an EOS. The set of *nested tokens* $\mathcal{T}(\mathfrak{E})$ of \mathfrak{E} is the set $\bigcup_{(P,T,F) \in \mathcal{N}} (d^{-1}(P, T, F) \times P^\oplus)$. The set of *nested markings* $\mathcal{M}(\mathfrak{E})$ of \mathfrak{E} is $\mathcal{T}(\mathfrak{E})^\oplus$. Given $\lambda, \rho \in \mathcal{M}(\mathfrak{E})$, we say that λ is a *sub-marking* of μ if $\lambda \sqsubseteq \mu$.

Note that λ is a sub-marking of μ iff there is some nested marking μ' such that $\mu = \lambda + \mu'$. EOSs inherit the graphical representation of PNs with the provision that we represent nested tokens via a dashed line from the system net place to an instance of the object net where the internal marking is represented in the standard PN way. However, if the nested token is $\langle p, \varepsilon \rangle$ for a system net place p of type \blacksquare , we represent it with a black-token \blacksquare on p . If a place p hosts $n > 2$ black-tokens, then we represent them by writing n on p . Each event $\langle \hat{\tau}, \theta \rangle$ is depicted by labeling $\hat{\tau}$ by $\langle \theta \rangle$ (possibly omitting double curly brackets). If there are several events involving $\hat{\tau}$, then $\hat{\tau}$ has several labels.

Example 1. Fig. 1 depicts the system net \hat{N} (the idle transitions are omitted) and object net drone of an EOS $\mathfrak{E} = \langle \hat{N}, \mathcal{N}, d, \Theta \rangle$ modeling a drone that (1) moves between a base and a field, (2) has two batteries, (3) consumes one charge-unit per battery per movement, and (4) charges its batteries by multiples of two charge-units when at base. Technically, $\mathcal{N} = \{\text{drone}, \blacksquare\}$ (even if \blacksquare is unused), $d(\text{base}) = d(\text{field}) = \text{drone}$, and Θ synchronizes `takeOff` and `land` (respectively `charge`) in \hat{N} with `move` (`charge1` and `charge2`) in drone. Formally, $\Theta = \{\langle \text{takeOff}, \{\{\text{move}\}\} \rangle, \langle \text{land}, \{\{\text{move}\}\} \rangle, \langle \text{charge}, \{\{\text{charge1}\}\} \rangle, \langle \text{charge}, \{\{\text{charge2}\}\} \rangle\}$.

¹This way, the system net and the object nets are pairwise distinct.

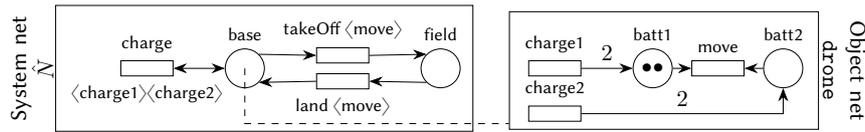


Figure 1: EOS in Example 1 with marking $\{\langle\langle\text{drone}, \{\{\text{batt1}, \text{batt1}\}\}\rangle\rangle\}$. The idle transitions are omitted.

The marking $\mu = \langle\text{drone}, \{\{\text{batt1}, \text{batt1}\}\}\rangle$ represents a single partially charged drone at base, with two charge units in the first battery.

When firing an event $\langle\tau, \theta\rangle$, nested tokens in the system net are consumed according to the preconditions of τ in the standard PN way. At the same time, for each object net N , the inner tokens are merged so as to obtain a PN marking $\mu(N)$ for N (possibly empty). Then, transitions in $\theta(N)$ are fired in the standard PN way obtaining markings $\mu'(N)$. Next, nested markings with empty inner markings are produced in the system net according to the postconditions of τ . Finally, the markings $\mu'(N)$ are non-deterministically distributed among the empty nested tokens, according to the typing function. To be fired, the event must be enabled at both the system and at the object net level. This is captured by the enabledness condition, which makes use of projection operators at the system (Π^1) and at the object net level (Π_N^2 for each $N \in \mathcal{N}$).

Definition 3 (Projection Operators). Let \mathfrak{E} be an EOS $\langle\hat{N}, \mathcal{N}, d, \Theta\rangle$. The *projection operators* Π^1 maps each nested marking $\mu = \sum_{i \in I} \langle\hat{p}_i, M_i\rangle$ for \mathfrak{E} to the PN marking $\sum_{i \in I} \hat{p}_i$ for \hat{N} . Given an object net $N \in \mathcal{N}$, the *projection operators* Π_N^2 maps each nested marking $\mu = \sum_{i \in I} \langle\hat{p}_i, M_i\rangle$ for \mathfrak{E} to the PN marking $\sum_{j \in J} M_j$ for N where $J = \{i \in I \mid d(\hat{p}_i) = N\}$.

To define the enabledness condition, we need the following notation. We set $\text{pre}_N(\theta(N)) = \sum_{i \in I} \text{pre}_N(t_i)$ where $(t_i)_{i \in I}$ is an enumeration of $\theta(N)$ counting multiplicities. We analogously set $\text{post}_N(\theta(N)) = \sum_{i \in I} \text{post}_N(t_i)$.

Definition 4 (Enabledness Condition). Let \mathfrak{E} be an EOS $\langle\hat{N}, \mathcal{N}, d, \Theta\rangle$. Given an event $e = \langle\hat{\tau}, \theta\rangle \in \Theta$ and two markings $\lambda, \rho \in \mathcal{M}(\mathfrak{E})$, the *enabledness condition* $\Phi(\langle\hat{\tau}, \theta\rangle, \lambda, \rho)$ holds iff

$$\begin{aligned} \Pi^1(\lambda) = \text{pre}_{\hat{N}}(\hat{\tau}) \wedge \Pi^1(\rho) = \text{post}_{\hat{N}}(\hat{\tau}) \wedge \forall N \in \mathcal{N}, \Pi_N^2(\lambda) \geq \text{pre}_N(\theta(N)) \wedge \\ \forall N \in \mathcal{N}, \Pi_N^2(\rho) = \Pi_N^2(\lambda) - \text{pre}_N(\theta(N)) + \text{post}_N(\theta(N)) \end{aligned}$$

The event e is *enabled with mode* (λ, ρ) on a marking μ iff $\Phi(e, \lambda, \rho)$ holds and $\lambda \sqsubseteq \mu$. Its firing results in the step $\mu \xrightarrow{\langle e, \lambda, \rho \rangle} \mu - \lambda + \rho$.

Example 2. In the setting of the EOS \mathfrak{E} and marking μ in Ex. 1 (Fig. 1), the event $\langle\text{charge}, \{\{\text{charge1}\}\}\rangle$ is enabled on $\mu = \langle\text{base}, \{\{\text{batt1}, \text{batt1}\}\}\rangle$ with mode (λ, ρ) where $\lambda = \mu$ and $\rho = \langle\text{base}, \{\{\text{batt1}, \text{batt1}, \text{batt1}, \text{batt1}\}\}\rangle$. Since $\lambda = \mu$, its firing results in the step $\mu \xrightarrow{\langle e, \lambda, \rho \rangle} \rho$. Instead, the event $\langle\text{charge}, \{\{\text{charge2}\}\}\rangle$ is enabled on μ with mode (λ, ρ') where $\rho' = \langle\text{base}, \{\{\text{batt1}, \text{batt1}, \text{batt2}, \text{batt2}\}\}\rangle$. Its firing results in the step $\mu \xrightarrow{\langle e, \lambda, \rho' \rangle} \rho'$. These are the only enabling modes for $\langle\text{charge}, \{\{\text{charge1}\}\}\rangle$ and $\langle\text{charge}, \{\{\text{charge2}\}\}\rangle$ on μ . No other event is enabled on μ , irrespective of the mode.

The reachability problem for EOSs is defined in the usual way, i.e., whether there is a run (sequence of event firings) from an initial marking μ_0 to a target marking μ_f . Also coverability definition is standard, but with respect to the order \leq_f (denoted by \leq in [13]; see Def. 6 below) that allows one to add both (1) tokens in the inner markings of available nested tokens (2) or nested tokens with some internal marking on the system net places. It is known that both these problems are undecidable (Th. 4.3 in [13]). However, coverability is decidable on the fragment of *conservative EOSs* (cEOSs; Th. 5.2 in [13]), where, for each system net transition t , if t consumes a nested token on a place of type N , then it also produces at least one token on a place of the same type N . Nevertheless, reachability remains undecidable (Th. 5.5 in [13]).

Definition 5 (cEOS). A cEOS is an EOS $\mathfrak{E} = \langle \hat{N}, \mathcal{N}, d, \Theta \rangle$ with $\hat{N} = \langle \hat{P}, \hat{T}, \hat{F} \rangle$ where, for all $\hat{t} \in \hat{T}$ and $\hat{p} \in \text{Supp}(\text{pre}_{\hat{N}}(\hat{t}))$, there exists $\hat{p}' \in \text{Supp}(\text{post}_{\hat{N}}(\hat{t}))$ such that $d(\hat{p}) = d(\hat{p}')$.

3. Problem

We study reachability and coverability of EOSs (cEOSs) affected by several forms of lossiness. First, we define three lossiness relations and show their relevance. Second, we formally define the problem we study in the following sections.

3.1. Lossy EOSs

We study EOSs (cEOSs) affected by lossiness, where nested markings may non-deterministically lose their tokens according to a quasi order, called lossiness relation. Lossiness can occur (1) at the object level, if lossiness removes only tokens from the inner markings of nested tokens, (2) at the system level, if lossiness removes whole nested tokens only, and (3) at both levels (the full EOS), if both whole nested tokens and/or regular tokens from the remaining nested tokens are removed. These levels are captured, respectively, by the lossiness quasi orders \leq_o (object-lossiness), \leq_s (system-lossiness), and \leq_f (full-lossiness) as defined next.

Definition 6. Given an EOS \mathfrak{E} and two nested markings μ and μ' for \mathfrak{E} , we have (1) $\mu \leq_s \mu'$ if $\mu \sqsubseteq \mu'$ or, equivalently, there is some μ'' such that $\mu' = \mu + \mu''$, (2) $\mu \leq_o \mu'$ if we can write $\mu = \sum_{i \in I} \langle \hat{p}_i, M_i \rangle$ and $\mu' = \sum_{i \in I} \langle \hat{p}_i, M'_i \rangle$ and, for each $i \in I$, $M_i \sqsubseteq M'_i$, and (3) $\mu \leq_f \mu'$ if there is some nested marking μ'' such that $\mu \leq_o \mu'' \leq_s \mu'$.

Example 3. Consider the marking $\mu = \langle \text{base}, \{\{\text{batt1}, \text{batt1}\}\} \rangle$ in Ex. 1. By removing 1 or 2 charge units we obtain the markings $\mu_1 = \langle \text{base}, \{\{\text{batt1}\}\} \rangle$ and $\mu_2 = \langle \text{base}, \emptyset \rangle$. By adding to μ a discharged token at place `field`, we obtain the $\mu' = \langle \text{base}, \{\{\text{batt1}, \text{batt1}\}\} + \langle \text{field}, \emptyset \rangle$. By removing the drone from μ , we obtain $\mu'' = \emptyset$. We have, among the others, $\mu' \geq_s \mu$, $\mu \geq_o \mu_1 \geq_o \mu_2$, $\mu' \geq_f \mu$, $\mu' \geq_f \mu_2$, and $\mu' \geq_f \mu''$.

The relation \leq_f coincides with \leq in [13]. Moreover, the order of \leq_o and \leq_s is irrelevant in \leq_f definition (Rem. 1 below). An EOS (cEOS) suffering from object-, system-, or full-lossiness is called, respectively, object-, system-, or full-lossy EOS (cEOS) or, simply lossy EOS (cEOS).

These lossiness relations are relevant to model non-deterministic phenomena not directly captured by the EOS. For example, in the context of Ex. 1, object-lossiness results in the non-deterministic loss of tokens at places `batt1` and `batt2`, which models the partial/total discharge

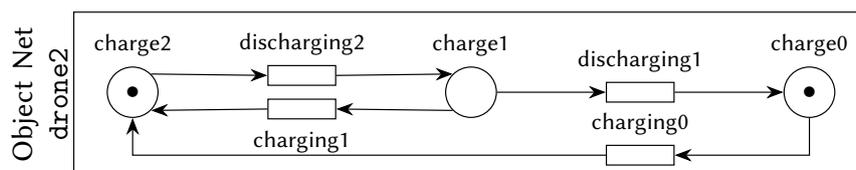


Figure 2: Object net `drone2` with one fully charged 2-bounded battery and a fully discharged one.

of drone batteries because of non-modeled drone movements within the base or the field, or because of other unexpected phenomena. In a slightly more complex EOS, with intermediate places capturing the flight from base to field and vice-versa, object-lossiness captures also the non-deterministic usage of extra charge-units because of contingencies like strong winds. Instead, system lossiness results in the loss of nested tokens modeling drones. This captures the loss of drones because of, e.g., break-downs, wrong flight paths, or seizure from higher priority processes (assuming the EOS is a module in a more complex system). Full-lossiness capture both aspects. Similar interpretations can be given each time the (nested) tokens represent resources, like charge-units or drones in Ex. 1. These scenarios are common in the literature (see, e.g., water- and fire-units in [12] and raw-resources in production plants in [16]).

Lossy EOSs are relevant also to capture partial/total internal break-downs. This happens, e.g., when the tokens model resource containers instead of resources themselves. For example, after modifying the drone object net into the object net `drone2` in Fig. 2,² each regular token represents a battery with bounded capacity. Its charge level is captured by its position in the object net. Consequently, object-lossiness represents the break-down of internal components, in this case the battery. The loss of all batteries results in drone deadlock (cf. [11]). This case is analogous to the application of system-lossiness discussed above, since drones can be seen as internal components of a higher level process (captured by the whole EOS): the loss/break-down of all drones results in system deadlock. More in general, this interpretation applies when the EOS uses conservative system and/or object nets.³ Also these scenario are common in the literature (see, e.g., the finite control of robots [13], the internal state of fire-fighters in [12], and customers and cars in [17]).

3.2. Lossy-reachability/coverability

We study the problem of ℓ -reachability/coverability, i.e., whether a target nested marking can be reached/covered from an initial one via a run suffering at most ℓ lossy steps, where $\ell \in \mathbb{N} \cup \{\omega\}$.⁴ These problems are relevant to study EOS robustness in front of losses/break-downs.

Definition 7. Given a transition system $TS = (V, \rightarrow)$, a transitive relation $<$ on V , and an $\ell \in \mathbb{N} \cup \{\omega\}$, a $(<, \ell)$ -run in TS is a run whose steps are labeled either by \rightarrow , called *standard steps*, or by $>$, called *<-lossy* or *lossy steps*, and at most ℓ steps are lossy. The set of $(<, \ell)$ -runs

²Also the events have to be modified accordingly, i.e., for each $n \in \{0, 1, 2\}$, by synchronizing `takeOff` and `landing` with `discharge n` object net transitions, and `charge` with the `charge n` object net transitions.

³A PN is conservative when each transition consumes and produces the same number of tokens.

⁴Recall that ω is the first limit ordinal, whose cardinality is $|\mathbb{N}|$, i.e., the same as \mathbb{N} .

from μ_0 is denoted by $Runs_\ell(<, \mu_0)$. A $(<, \ell)$ -run is called ℓ' -strong if it contains exactly ℓ' lossy steps.

The definition also applies to reflexive or anti-reflexive transitive relations, i.e., we can also talk about (\leq, ℓ) -runs and $(\not\leq, \ell)$ -runs. We denote a labeled step from μ to μ' by $\mu \rightsquigarrow \mu'$. To stress that the step is labeled by \rightarrow or by $<$, we denote it by $\mu \rightarrow \mu'$ or by $\mu > \mu'$, respectively. Whenever we have a lossy run σ from μ to μ' , we write $\mu \rightsquigarrow^\sigma \mu'$. The $(<, \ell)$ -reachability/coverability problems ask whether a target can be reached/covered under $<$ from an initial configuration with at most ℓ $<$ -lossy steps.

Definition 8 ($(<, \ell)$ -reachability/coverability for EOSs (cEOSs)). Let $\ell \in \mathbb{N} \cup \{\omega\}$.

Input: An EOS (cEOS) E , an initial marking μ_0 and a target marking μ_1 for E .

Output of reachability: Is there a run $\sigma \in Runs_\ell(<, \mu_0)$ such that $\mu_0 \rightsquigarrow^\sigma \mu_1$?

Output of coverability: Is there a run $\sigma \in Runs_\ell(<, \mu_0)$ such that $\mu_0 \rightsquigarrow^\sigma \mu \geq \mu_1$ for some μ ?

We call these problems *lossy-problems*. A *<-lossy-problem* is a lossy-problem under $<$. The *degree* of a $(<, \ell)$ -reachability/coverability problem is ℓ . If $\ell = 0$ we obtain standard reachability/coverability, i.e., over perfect runs. Our objective is to fully chart the decidability status of the lossy-problems for \leq_f , \leq_s , and \leq_o . Previous results for EOSs are available only for $\ell = 0$ and the relation \leq_f . Consequently, they do not inform us on the status of the other (proper) lossy-problems, whose study still requires a careful and in-depth analysis.

4. Coincident Problems

The well known notion of compatibility from WSTS has a strong impact on ℓ -reachability problems. In fact, we now show that all these problems, for $\ell \geq 1$ (including ω) and any quasi order \leq , collapse to $(\leq, 0)$ -coverability if and only if the lossiness relation is compatible.

Lemma 1. *Each yes-instance of (\leq, ℓ) -reachability is also a yes instance of (\leq, ℓ) -coverability.*

Proof. Immediate consequence of reflexivity of the quasi order \leq . □

Lemma 2. *Each yes-instance of (\leq, ℓ) -coverability is also a yes instance of $(\leq, \ell + 1)$ -reachability, if ℓ is finite, and a yes-instance of (\leq, ω) -reachability, if $\ell = \omega$.*

Proof. If μ_1 is coverable from μ_0 , then there is a (\leq, ℓ) -run σ from μ_0 to μ and $\mu \geq \mu_1$. Take the run σ' as the run σ followed by the lossy step $\mu \geq \mu_1$. Thus, σ' reaches μ_1 from μ_0 . Moreover, if ℓ is finite, then σ' is a $(\leq, \ell + 1)$ -run and, otherwise, σ' is a (\leq, ω) -run. □

Corollary 1. *(\leq, ω) -reachability and (\leq, ω) -coverability coincide.*

Note that Cor. 1 is consistent with other lossy PN models (see, e.g., [8]).

Lemma 3. *Each yes-instance of (\leq, ℓ) -reachability or (\leq, ℓ) -coverability is also a yes instance of (\leq, ω) -reachability or (\leq, ω) -coverability, respectively.*

Proof. Immediate consequence of the fact that each (\leq, ℓ) -run is also a (\leq, ω) -run. □

Summarising, for each quasi order \leq , the \leq -lossy-problems form a hierarchy ordered according to inclusion of the yes-instance sets. For each $i \in \mathbb{N}$, the i -th hierarchy level for \leq is the $(\leq, i/2)$ -reachability problem, if i is even, and the $(\leq, (i-1)/2)$ -coverability problem, if i is odd. We say that the hierarchy *collapses* if all the (\leq, ℓ) -reachability and (\leq, ℓ) -coverability problems with $\ell \geq 1$ coincide with (standard) $(\leq, 0)$ -coverability or, equivalently, if the yes-instances of (\leq, ω) -reachability are also yes-instances of $(\leq, 0)$ -coverability. The next lemma states that this latter property is equivalent to compatibility, that is, if $\mu_1 \geq \mu_2 \rightarrow \mu_3$, then there is a μ_4 such that $\mu_1 \rightarrow^* \mu_4 \geq \mu_3$.

Lemma 4. \leq is compatible iff each yes-instance of (\leq, ω) -reachability is also a yes-instance of $(\leq, 0)$ -coverability.

Proof. Assume that \leq is compatible. If v_1 is reachable from v_0 via an ω -run σ , then, without loss of generality, we can assume that σ is finite and, thus, it is a ℓ -run for some finite ℓ . By compatibility, we can push, one by one, the finitely many lossy steps in σ at the end of the run, obtaining an ℓ -run σ' (possibly with different length) where all lossy steps occur at the end, i.e., σ' is of the form $v_0 \rightarrow^* w_1 \geq w_2 \cdots \geq w_\ell \geq v_1$. By transitivity of \geq , there is also a run σ'' of the form $v_0 \rightarrow^* w_1 \geq v_1$, which witness that v_1 is \leq -coverable from v_0 . Vice-versa, assume that each yes-instance of (\leq, ω) -reachability is also a yes-instance of $(\leq, 0)$ -coverability. If $v_0 \geq v_1 \rightarrow v_2$, then v_2 is $(\leq, 1)$ -reachable from v_0 , as well as (\leq, ω) -reachable. Thus, v_2 is also coverable from v_0 , i.e., $v_0 \rightarrow^* v_1 \geq v_2$. \square

Corollary 2. The hierarchy of lossy-problems induced by \leq collapses iff \leq is compatible.

Note that Cor. 2 can be generalized to other lossy models, since its proof does not take advantage of the technical details of lossy EOSs, but relies only on a compatible quasi order. This fact has some immediate consequence on the lossy-problems we are studying. In fact, it is known that \leq_f is strong compatible on cEOSs, that is, if $\mu_1 \geq_f \mu_2 \xrightarrow{\langle e, \lambda, \rho \rangle} \mu_3$, then there is some μ_4 such that $\mu_1 \xrightarrow{\langle e, \lambda, \rho \rangle} \mu_4$ (Lemma 5.1 in [13]). Thus, the hierarchy for full-lossy cEOSs collapses. Note that \leq_f is not compatible over EOSs. This helps to prove the undecidability of reachability and coverability over them (cf. Th.4.3 in [13]). Since $(\leq_f, 0)$ -coverability for cEOSs is decidable (Th. 5.2 in [13]), we obtain the following theorem.

Theorem 1. For $\ell \geq 1$, (\leq_f, ℓ) -reachability and (\leq_f, ℓ) -coverability for cEOSs are decidable.

Instead \leq_s is compatible for both EOSs and cEOSs, as shown next.

Lemma 5. If (λ, ρ) enables the event e on μ and $\mu' \geq_s \mu$, then (λ, ρ) enables the event e also on μ' .

Proof. If (λ, ρ) enables the event e on μ , then the enabledness formula $\Phi(e, \lambda, \rho)$ holds and $\lambda \leq_s \mu$. Since $\mu \leq_s \mu'$, by transitivity of \leq_s , we have that $\lambda \leq_s \mu'$. Thus, (λ, ρ) enables e on μ' . \square

Lemma 6. \leq_s is strong compatible on EOSs.

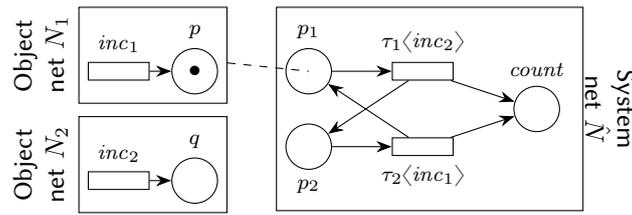


Figure 4: The lossiness-counter gadget \mathcal{G} in Def. 9 (where $d(p_1) = N_1, d(p_2) = N_2, d(count) = \blacksquare$) with initial marking $\mu_0 = \{\langle p_1, \{\{p\}\} \rangle\}$.

5. Distinct Problems

Unfortunately, compatibility does not hold for \leq_o and \leq_f on EOSs. This is the main fact allowing the simulation of inhibitory nets via EOSs in [13]. Thus, the hierarchies induced by \leq_o and \leq_f do not collapse. In fact, we now show that all the problems in the hierarchy are distinct. We make use of a gadget with a dedicated place that counts the lossy steps.

Definition 9. The *lossiness-counter gadget* \mathcal{G} is the EOS $(\hat{N}, \mathcal{N}, d, \Theta)$ depicted in Fig. 4 where

1. $\hat{N} = (\hat{P}, \hat{T}, \hat{F})$ where $\hat{P} = \{p_1, p_2, count\}$, $\hat{T} = \{\tau_1, \tau_2\}$, $\hat{F}(x) = 1$ if $x \in \{(p_1, \tau_1), (\tau_1, count), (\tau_1, p_2), (p_2, \tau_2), (\tau_2, count), (\tau_2, p_1)\}$ and $\hat{F}(x) = 0$ otherwise,
2. $\mathcal{N} = \{N_1, N_2\}$ where $N_1 = (\{p\}, \{inc_1\}, F_1)$, $F_1(\{(inc_1, p)\}) = 1$, $F_1(\{(p, inc_1)\}) = 0$, $N_2 = (\{q\}, \{inc_2\}, F_2)$, $F_2(\{(inc_2, q)\}) = 1$, and $F_2(\{(q, inc_2)\}) = 0$,
3. $d(p_1) = N_1, d(p_2) = N_2$, and $d(count) = \blacksquare$, and
4. $\Theta = \{\langle \tau_1, inc_2 \rangle, \langle \tau_2, inc_1 \rangle\}$.

In what follows, we work with a fixed initial nested marking $\mu_0 = \langle p_1, \{\{p\}\} \rangle$ of \mathcal{G} .

5.1. Distinct Problems for Object-lossy EOSs

We first study the case of object-lossiness.

Lemma 8. *Let $<$ be a transitive relation. Given $\ell \in \mathbb{N} \cup \{\omega\}$, we have that μ is (\leq, ℓ) -reachable from μ_0 in \mathcal{G} iff μ is (\preceq, ℓ) -reachable from μ_0 .*

Proof. Each \preceq -lossy step $\mu_1 \succ \mu_2$ can be interpreted as a \leq -lossy step $\mu_1 \geq \mu_2$. Thus, each (\preceq, ℓ) -run can be interpreted as a (\leq, ℓ) -run. Similarly, each (\leq, ℓ) -run that does not contain reflexive lossy steps of the form $\mu \geq \mu$ can be interpreted as a (\preceq, ℓ) -run. Moreover, each maximal finite or infinite sub-run $\mu \geq \mu \geq \dots \mu \geq \dots$ of an arbitrary (\leq, ℓ) -run σ can be substituted by a single occurrence of μ , obtaining a (\preceq, ℓ) -run without reflexive lossy steps. \square

Lemma 9. *For each $\ell \geq 0$, the lossiness-counter gadget \mathcal{G} exhibits a single maximal ℓ -strong (\preceq_o, ℓ) -run from its initial nested-marking μ_0 . This run has the form*

$$\mu_0 \succeq_o \mu'_0 \rightarrow \mu_1 \succeq_o \mu'_1 \rightarrow \mu_2 \succeq_o \dots \mu_{\ell-1} \succeq_o \mu'_{\ell-1} \rightarrow \mu_\ell$$

where, for each $i \leq \ell$, we have $\mu_i = \langle x_i, \{\{y_i\}\} \rangle + i \langle \text{count}, \varepsilon \rangle$ and $\mu'_i = \langle x_{i+1}, \emptyset \rangle + i \langle \text{count}, \varepsilon \rangle$ where, for each $j \in \mathbb{N}$, $x_j = p_1$ and $y_j = p$ if j is even and $x_j = p_2$ and $y_j = q$ if j is odd.

Proof. By induction on ℓ . If $\ell = 0$, then, since μ_0 is a deadlock for \rightarrow , the run μ_0 of length 0 is the only (\preceq_o, ℓ) -run. Moreover, 0 is even and $\mu_0 = \langle p_1, \{\{p\}\} \rangle + 0 \langle \text{count}, \varepsilon \rangle$.

If the inductive hypothesis holds for an arbitrary even ℓ , then $\mu_\ell = \langle p_1, \{\{p\}\} \rangle + \ell \langle \text{count}, \varepsilon \rangle$ is a deadlock for \rightarrow . The only way to continue the run is by one \succeq_o step. However, the only token that can be lost under \leq_o is the token inside $\langle p_1, \{\{p\}\} \rangle$, thus, the only \leq_o -successor of μ_ℓ is $\mu'_\ell = \langle p_1, \emptyset \rangle + \ell \langle \text{count}, \varepsilon \rangle$. On μ'_ℓ there is no available \succeq_o step and the only enabled event is (τ_1, inc_2) with mode (λ, ρ) where $\lambda = \langle p_1, \emptyset \rangle$ and $\rho = \langle p_2, \{\{q\}\} \rangle + \langle \text{count}, \varepsilon \rangle$. Its firing reaches $\mu_{\ell+1} = \langle p_2, \{\{q\}\} \rangle + (\ell + 1) \langle \text{count}, \varepsilon \rangle$. This configuration is a deadlock for \rightarrow and the so obtained $(\preceq_o, \ell + 1)$ -run from μ_0 to $\mu_{\ell+1}$ already contains $\ell + 1$ \preceq_o -steps. Thus, this run is maximal among the $(\preceq_o, \ell + 1)$ -runs.

If the inductive hypothesis holds for an arbitrary odd ℓ , the same argument applies with the provision that p_1 and p_2 , p and q , as well as (τ_1, inc_2) and (τ_2, inc_1) , have to be swapped. \square

Corollary 3. *For each finite ℓ , the set of nested markings which are (\leq_o, ℓ) -reachable from μ_0 is $\{\mu_i \mid i \in \{0, \dots, \ell\}\} \cup \{\mu'_i \mid i \in \{0, \dots, \ell - 1\}\}$ where μ_i and μ'_i are defined as in Lemma 9.*

Consequently, using the same notation as in Lem. 9, for each $\ell \in \mathbb{N}$, we have that μ'_ℓ is (\leq_o, ℓ) -coverable but not (\leq_o, ℓ) -reachable and $\mu_{\ell+1}$ is $(\leq_o, \ell + 1)$ -reachable but not (\leq_o, ℓ) -coverable. Thus, the sequence of yes-instance sets of (\leq_o, ℓ) -reachability/coverability problems (recall Lem. 1 and Lem. 2) is a sequence of proper subsets. Consequently, for each finite ℓ , all (\leq_o, ℓ) -reachability/coverability problems are pairwise distinct. Moreover, while (\leq_o, ω) -reachability coincides with (\leq_o, ω) -coverability by Lem. 4, also (\leq_o, ω) -reachability is distinct from (\leq_o, ℓ) -reachability/coverability for each ℓ .⁶

Corollary 4. *For each $\ell_1 < \ell_2 \leq \omega$, we have that (\leq_o, ℓ_1) -reachability, (\leq_o, ℓ_1) -coverability, (\leq_o, ℓ_2) -reachability, and (\leq_o, ℓ_2) -coverability are pair-wise distinct problems.*

5.2. Distinct Problems for Full-lossy EOSs

A fact analogous to Cor. 4 applies also to \leq_f . In fact, even if \mathcal{G} exhibits more complex maximal (\preceq_f, ℓ) -runs, we still need ℓ lossy steps to put ℓ tokens on *count*. We first show that if a marking is reachable/coverable using \leq_f -lossy steps, then it is also covered under \leq_s by some marking reachable via a run as in Lem. 9.

Lemma 10. *If $\sigma \in \text{Runs}_\ell(\leq_f, \mu_0)$ and $\mu_0 \rightsquigarrow^\sigma \mu$, then there is some $\lambda \succeq_s \mu$ and there is an ℓ' -strong run $\sigma' \in \text{Runs}_{\ell'}(\preceq_o, \mu_0)$ such that $\mu_0 \rightsquigarrow^{\sigma'} \lambda$, for some $\ell' \leq \ell$.*

Proof. Since $\leq_f = \leq_o \circ \leq_s$, we can expand each \leq_f -lossy step in two subsequent \leq_o - and \leq_s -lossy steps. By compatibility of \leq_s on EOSs (Lem. 7) and the fact that $\leq_s \circ \leq_o = \leq_o \circ \leq_s$ (Rem. 1), we can push all the \leq_s -lossy steps at the end of the run, obtaining a run $\sigma' \in \text{Runs}_\ell(\leq_o, \mu_0)$ such that $\mu_0 \rightsquigarrow^{\sigma'} \lambda \succeq_s \dots \succeq_s \mu$, for some marking λ . By transitivity of \leq_s , we have $\lambda \succeq_s \mu$.

⁶Otherwise, it would coincide also with $(\leq_o, \ell + 1)$ -reachability/coverability. Thus, (\leq_o, ℓ) -reachability/coverability and $(\leq_o, \ell + 1)$ -reachability/coverability would coincide, which is a contradiction.

Moreover, we can drop the \leq_o -lossy steps in σ' of the form $\pi \geq_o \pi$ for some π , obtaining an ℓ' -strong run $\sigma'' \in \text{Runs}_{\ell'}(\leq_o, \mu_0)$ for some $\ell' \leq \ell$. \square

We now show that also the sequence of yes-instance sets of \leq_f -lossy reachability/coverability problems (recall Lem. 1 and Lem. 2) is a sequence of proper subsets. We do that by using the same markings μ_ℓ and μ'_ℓ defined in Lem. 9.

Lemma 11. *Using the notation in Lem. 9, for each finite $\ell \in \mathbb{N}$, we have that $\mu_{\ell+1}$ is $(\leq_f, \ell + 1)$ -reachable, but not (\leq_f, ℓ) -coverable.*

Proof. By Lem. 9 we know that $\mu_{\ell+1}$ is $(\leq_o, \ell + 1)$ -reachable and, thus, also $(\leq_f, \ell + 1)$ -reachable. We now show that $\mu_{\ell+1}$ is not (\leq_f, ℓ) -coverable. Assume by contradiction that $\mu_{\ell+1}$ is (\leq_f, ℓ) -coverable. Then, there is some run $\sigma \in \text{Runs}_\ell(\leq_f, \mu_0)$ and a nested marking λ such that $\mu_0 \rightsquigarrow^\sigma \lambda \geq_f \mu_{\ell+1}$. Since $\mu_{\ell+1} \geq_s (\ell + 1)\langle \text{count}, \varepsilon \rangle$, also $\lambda \geq_f (\ell + 1)\langle \text{count}, \varepsilon \rangle$. Thus, $\lambda \geq_s (\ell + 1)\langle \text{count}, \varepsilon \rangle$.⁷ By applying Lem. 10 on the run σ , there is some nested marking $\lambda' \geq_s \lambda$ and, for some $\ell' \leq \ell$, a ℓ' -strong run $\sigma' \in \text{Runs}_{\ell'}(\leq_o, \mu_0)$ such that $\mu_0 \rightsquigarrow^{\sigma'} \lambda'$. Thus, by Lem. 9, we have that either $\lambda' = \mu_{\ell'}$ or $\lambda' = \mu'_{\ell'-1}$ and, hence, λ' places at most ℓ' black tokens on *count*. However, $\lambda' \geq_s \lambda \geq_s (\ell + 1)\langle \text{count}, \varepsilon \rangle$, i.e., λ' puts at least $\ell + 1$ tokens on *count* even if $\ell' \leq \ell < \ell + 1$, which is a contradiction. \square

Lemma 12. *Using the notation in Lem. 9, for each finite $\ell \in \mathbb{N}$, we have that $\mu'_{\ell+1}$ is (\leq_f, ℓ) -coverable, but not (\leq_f, ℓ) -reachable.*

Proof. By Lem. 9 we know that $\mu'_{\ell+1}$ is (\leq_o, ℓ) -coverable and, thus, also (\leq_f, ℓ) -coverable. We now show that $\mu'_{\ell+1}$ is not (\leq_f, ℓ) -reachable. Assume by contradiction that $\mu'_{\ell+1}$ is (\leq_f, ℓ) -reachable. Then, there is some run $\sigma \in \text{Runs}_\ell(\leq_f, \mu_0)$ such that $\mu_0 \rightsquigarrow^\sigma \mu'_{\ell+1}$. By applying Lem. 10 on the run σ , there is some nested marking $\lambda \geq_s \mu'_{\ell+1}$ and, for some $\ell' \leq \ell$, a ℓ' -strong run $\sigma' \in \text{Runs}_{\ell'}(\leq_o, \mu_0)$ such that $\mu_0 \rightsquigarrow^{\sigma'} \lambda$. Since $\lambda \geq_s \mu'_{\ell+1} \geq_s (\ell + 1)\langle \text{count}, \varepsilon \rangle$, we have that λ puts at least $\ell + 1$ tokens on *count*. However, by Lem. 9, we have that either $\lambda = \mu_{\ell'}$ or $\lambda = \mu'_{\ell'-1}$. Hence, λ puts at most $\ell' \leq \ell < \ell + 1$ tokens on *count*, which is a contradiction. \square

Corollary 5. *For each $\ell_1 < \ell_2 \leq \omega$, we have that (\leq_f, ℓ_1) -reachability, (\leq_f, ℓ_1) -coverability, (\leq_f, ℓ_2) -reachability, and (\leq_f, ℓ_2) -coverability are pair-wise distinct problems.*

6. Undecidability for object- and system-lossy cEOSs

We now study the decidability status of $(\leq_o, 0)$ -coverability and $(\leq_s, 0)$ -coverability for cEOSs. In [18], reachability for cEOSs is proved undecidable via a reduction from reachability of 2CMs. We provide a variant of that reduction that reduces reachability of 2CMs to \leq_o -coverability and to \leq_s -coverability of cEOSs. This proves that both coverability problems are undecidable. By Th. 3 and Th. 2, all ℓ -reachability problems for object-lossy and system-lossy cEOSs are, thus, undecidable. Since cEOSs are a special case of EOSs, undecidability applies also to object-lossy and system-lossy EOSs.

⁷In fact, $\lambda \geq_f (\ell + 1)\langle \text{count}, \varepsilon \rangle$ implies that there is some π such that $\lambda \geq_s \pi \geq_o (\ell + 1)\langle \text{count}, \varepsilon \rangle$; since the latter marking only places black tokens at the system net level, π is obtained by adding zero tokens at the object net level, i.e., $\pi = (\ell + 1)\langle \text{count}, \varepsilon \rangle$. Hence $\lambda \geq_s (\ell + 1)\langle \text{count}, \varepsilon \rangle$.

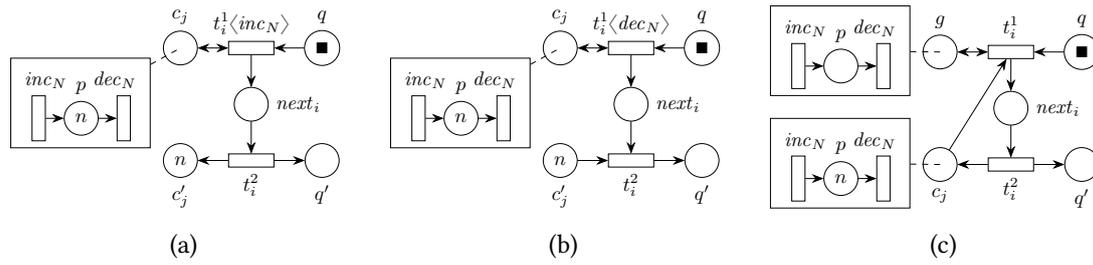


Figure 5: Part of $\mathfrak{E}_{\mathcal{K}}$ capturing an (a) increment, (b) decrement, or (c) zero-check instruction $i \in \delta$.

6.1. Reduction to Reachability

We show a reduction from reachability of a target configuration (q_f, n_1, n_2) of any 2CM $\mathcal{K} = (Q, \delta, q_0)$ with increment, decrement, and zero-check instructions to reachability of a cEOS $\mathfrak{E}_{\mathcal{K}}$.

Definition 10. Given a 2CM $\mathcal{K} = (Q, \delta, q_0)$, we define the EOS $\mathfrak{E}_{\mathcal{K}} = (\hat{N}, \mathcal{N}, d, \Theta)$ where

1. $\hat{N} = (\hat{P}, \hat{T}, \hat{F})$ is such that:
 - \hat{P} contains Q , a place g , a place $next_i$ for each instruction $i \in \delta$, and two places c_j and c_j' for each counter cnt_j ;
 - \hat{T} contains two transitions t_i^1 and t_i^2 for each instruction $i \in \delta$;
 - For each increment (resp., decrement, zero-check) instruction $i \in \delta$, \hat{F} captures the pre- and post-conditions depicted in Fig. 5a (Fig. 5b, Fig. 5c).
2. \mathcal{N} contains the net \blacksquare and a single net $N = (P, T, F)$ where $P = \{p\}$, $T = \{inc_N, dec_N\}$, $F(inc_N, p) = F(p, dec_N) = 1$, and $F(p, inc_N) = F(dec_N, p) = 0$.
3. $d(c_1) = d(c_2) = d(g) = N$ and $d(x) = \blacksquare$ for each other place $x \in \hat{P} \setminus \{c_1, c_2, g\}$.
4. The synchronization structure Θ contains the events
 - $e_i^1 = (t_i^1, \{\{inc_N\}\})$ and $e_i^2 = (t_i^2, \emptyset)$ for each increment instruction i .
 - $e_i^1 = (t_i^1, \{\{dec_N\}\})$ and $e_i^2 = (t_i^2, \emptyset)$ for each decrement instruction i .
 - $e_i^1 = (t_i^1, \emptyset)$ and $e_i^2 = (t_i^2, \emptyset)$ for each zero-check instruction i .

Note that $\mathfrak{E}_{\mathcal{K}}$ is a cEOS. It weakly simulates the increment and decrement instructions of \mathcal{K} and performs zero-guesses in place of zero-check instructions. These guesses may be wrong but leave behind them two irreversible evidences: tokens in the internal marking of the object net at g and non-matching numbers of tokens on the place c_j and in the object net on c_j' , for some $j \in \{0, 1\}$. The former can be detected by $(\leq_s, 0)$ -coverability; the latter can be detected by $(\leq_o, 0)$ -coverability. We make these notions precise.

Definition 11. We say that a nested marking μ for $\mathfrak{E}_{\mathcal{K}}$ is *legal* if it places exactly one object-token N at c_0, c_1 , and g , exactly one \blacksquare on exactly one place $q \in Q$, and no token on any place in $\hat{P} \setminus \{c_0', c_1', q\}$. If μ is legal, we denote

1. by $[\mu]_j'$ the number of black tokens placed at c_j' ,
2. by $[\mu]_j$ the number of black-tokens in the object-token at c_j ,
3. by $[\mu]_g$ the number of black-tokens in the object-token at g , and
4. by $[\mu]_Q$ the place $q \in Q$ marked by μ with a black-token.

Moreover, we say that μ :

1. is *broken at (counter) j* if $[\mu]_j \neq [\mu]'_j$, *sub-broken at j* if $[\mu]_j < [\mu]'_j$, *broken at g* if $[\mu]_g \neq 0$,
2. is *broken* if it is broken at 0, at 1, or at g ,
3. *encodes* the 2CM configuration $c = (q, n_0, n_1)$, denoted by $\mu = \langle c \rangle$, if μ is non-broken and $c = ([\mu]_Q, [\mu]_0, [\mu]_1)$.

The following lemma is a direct consequence of the pre- and post-conditions of the transitions in $\mathfrak{E}_{\mathcal{K}}$, the shape of its synchronization structure Θ , the EOS semantics, and Def. 10. Its proof consists in simple, yet space-consuming algebraic checks and, thus, it is omitted.

Lemma 13. *Let $i \in \{(q, +, j, q'), (q, -, j, q'), (q, =, j, q')\} \cap \delta$ and μ a legal nested marking for*

$\mathfrak{E}_{\mathcal{K}}$. *If e_i^1 is enabled on μ , there is some μ' and μ'' such that $\mu \xrightarrow{e_i^1} \mu' \xrightarrow{e_i^2} \mu''$ and*

1. $[\mu]_Q = q$, μ'' is legal, $[\mu'']_Q = q'$, $[\mu'']_{1-j} = [\mu]_{1-j}$, and $[\mu'']'_{1-j} = [\mu]'_{1-j}$.
2. if $i = (q, +, j, q')$, then $[\mu'']_j = [\mu]_j + 1$, $[\mu'']'_j = [\mu]'_j + 1$ and $[\mu'']_g = [\mu]_g$.
3. if $i = (q, -, j, q')$, then $[\mu'']_j = [\mu]_j - 1$, $[\mu'']'_j = [\mu]'_j - 1$ and $[\mu'']_g = [\mu]_g$.
4. if $i = (q, =, j, q')$, then $[\mu'']_j = 0$, $[\mu'']'_j = [\mu]'_j$, and $[\mu'']_g = [\mu]_g + [\mu]_j$.

Corollary 6. *Let $\mu \xrightarrow{e_i^1} \mu' \xrightarrow{e_i^2} \mu''$ and μ legal. If μ is sub-broken at 0 or at 1, then μ'' is sub-broken at 0 or at 1, respectively. If μ is broken at g , then μ'' is broken at g .*

Corollary 7. *Let $\mu \xrightarrow{e_i^1} \mu' \xrightarrow{e_i^2} \mu''$, and μ legal and non-broken at 0 or at 1. If μ'' is broken at 0 or at 1, respectively, then μ'' is sub-broken at 0 or at 1.*

Corollary 8. *Let $\mu \xrightarrow{e_i^1} \mu' \xrightarrow{e_i^2} \mu''$, μ legal, and $i \in \{(q, +, j, q'), (q, -, j, q')\} \cap \delta$. If μ is non-broken, then μ'' is non-broken.*

Cor. 8 indicates that the simulation of increment or decrement instructions cannot lead, on their own, to broken markings. We now show that this is not the case for some simulation of zero-checks, called *wrong zero-guesses*.

Definition 12. *A wrong zero-guess on counter j is a run $\mu \xrightarrow{e_i^1} \mu' \xrightarrow{e_i^2} \mu''$ where i is a zero-check instruction on counter j , μ is legal, and $[\mu]_j > 0$.*

Lemma 14. *If $\sigma : \mu \xrightarrow{e_i^1} \mu' \xrightarrow{e_i^2} \mu''$ and μ is legal and non-broken, then μ'' is broken at j for some $j \in \{0, 1\}$ iff σ is a wrong zero-guess on counter j .*

Proof. By Cor. 8, since μ is not broken at j but μ'' is, i is a zero-check instruction on counter j . If $[\mu]_j = [\mu]'_j = 0$, then $[\mu'']_j = 0 = [\mu]'_j = [\mu'']'_j$ and μ'' is not broken at j , contradiction. Thus, $[\mu]_j = [\mu]'_j > 0$ and, thus, σ is a wrong zero-guess. Vice-versa, if σ is a wrong-zero guess on counter j , then, by Lemma 13, μ'' is broken at j . \square

The next lemma is proved analogously.

Lemma 15. *If $\sigma : \mu \xrightarrow{e_i^1} \mu' \xrightarrow{e_i^2} \mu''$ and μ is legal and non-broken, then μ is broken at g iff σ is a wrong zero-guess.*

Corollary 9. *If $\sigma : \langle (q_0, 0, 0) \rangle \rightarrow^* \mu$ is a run of even length, then the following are equivalent: (1) μ is sub-broken at 0 or 1, (2) μ is broken at g , (3) μ is broken, and (4) σ has a wrong zero-guess.*

Clearly, for each run $c_0 \xrightarrow{i_0} c_1 \rightarrow i_1 \dots$ in \mathcal{K} there is a run $\langle c_0 \rangle \xrightarrow{e_{i_0}^1} \mu_0 \xrightarrow{e_{i_0}^2} \langle c_1 \rangle \xrightarrow{e_{i_1}^1} \mu_1 \xrightarrow{e_{i_1}^2} \dots$. If the target configuration (q_f, n_0, n_1) is reachable from $(q_0, 0, 0)$ in \mathcal{K} , then $\mu_f = \langle (q_f, n_0, n_1) \rangle$ is reachable from $\mu_0 = \langle (q_0, 0, 0) \rangle$ in $\mathfrak{E}_{\mathcal{K}}$. Vice-versa, if μ_f is reachable from μ_0 in $\mathfrak{E}_{\mathcal{K}}$ via a run σ , then, μ_f is legal, non-broken, $[\mu_f]_Q = q_f$, and σ has even length. Thus, σ does not have any wrong zero-guess and can be simulated by a corresponding run in \mathcal{K} from $(q_0, 0, 0)$.

Theorem 4. *(q_f, n_0, n_1) is reachable from $(q_0, 0, 0)$ in \mathcal{K} iff $\mu_f = \langle (q_f, n_0, n_1) \rangle$ is reachable from $\mu_0 = \langle (q_0, 0, 0) \rangle$ in $\mathfrak{E}_{\mathcal{K}}$.*

Note that the EOS in Def. 10 is a cEOS. Thus, since 2CM reachability is undecidable, our result confirms that reachability for cEOSs is undecidable. However, we can conclude more.

6.2. From Reachability to $(\leq_o, 0)$ -coverability

We now show that our construction yields undecidability also for $(\leq_o, 0)$ -coverability. This is based on the fact that the nested marking μ_f in Th. 4 is reachable if and only if it is $(\leq_o, 0)$ -coverable. In fact, if $\mu_0 \rightarrow^* \mu \geq_o \mu_f$, then, for each $j \in \{0, 1\}$,

1. μ and μ_f mark in the same way all system net places of type \blacksquare .
2. $[\mu]_Q$ is well-defined, $[\mu]_Q = [\mu_f]_Q = q_f$, μ is reachable only via runs of even length, and μ is legal.
3. for each $j \in \{0, 1\}$, $[\mu]'_j = [\mu_f]'_j = [\mu_f]_j \leq [\mu]_j$;
4. if μ is broken at j , then by Cor. 6 and Cor. 7 it is sub-broken at j and, thus, $[\mu]_j < [\mu]'_j \leq [\mu]_j$, contradiction; thus, μ is not broken at j and is not broken at 0 and, consequently, not broken at g .
5. $[\mu]_j = [\mu]'_j = [\mu_f]'_j = [\mu_f]_j$ and $[\mu]_g = 0 = [\mu_f]_g$.

Summarising μ and μ_f coincide. Thus, μ_f is $(\leq_o, 0)$ -coverable from μ_0 iff it is reachable. By Th. 4, we obtain the following result.

Theorem 5. *$(\leq_o, 0)$ -coverability for cEOSs is undecidable.*

By Th. 3, by undecidability of reachability for cEOSs (Th.5.5 in [13]), and by the fact that each cEOS is also an EOS, we obtain the following result.

Corollary 10. *For each $\ell \in \mathbb{N} \cup \{\omega\}$, we have that (\leq_o, ℓ) -reachability and (\leq_o, ℓ) -coverability for cEOSs and for EOSs are undecidable.*

6.3. From Reachability to $(\leq_s, 0)$ -coverability

We now show that similar statements apply also for $(\leq_s, 0)$ -coverability. In fact, if $\mu_0 \rightarrow^* \mu \geq_s \mu_f$, then, for each $j \in \{0, 1\}$,

1. μ places at least one black-token on $[\mu_f]_Q = q_f$, thus μ is reachable only via runs of even length, and μ is legal.

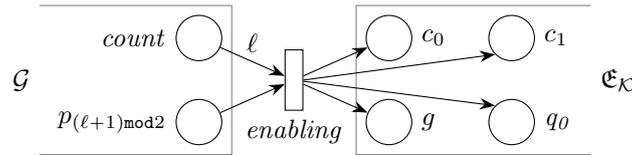


Figure 6: The places, transitions, and conditions we add on top of \mathcal{M} .

2. possibly with the exception of c'_0 and c'_1 , μ and μ_f coincide on all system net places, including all object-tokens on them.
3. $[\mu]_j = [\mu_f]_j = [\mu_f]'_j \leq [\mu]'_j$ and $[\mu]_g = [\mu_f]_g$, thus μ is not broken and $[\mu]'_j = [\mu]_j = [\mu_f]_j$.

Summarising μ and μ_f coincide. Thus, μ_f is $(\leq_s, 0)$ -coverable from μ_0 iff it is reachable. By Th. 4, we obtain the following result.

Theorem 6. $(\leq_s, 0)$ -coverability for cEOSs is undecidable.

By Th. 2, by undecidability of reachability for cEOSs (Th.5.5 in [13]), and the fact that each cEOS is also an EOS, we obtain the following corollaries.

Corollary 11. For each $\ell \in \mathbb{N} \cup \{\omega\}$, we have that (\leq_s, ℓ) -reachability and (\leq_s, ℓ) -coverability for cEOSs and for EOSs are undecidable.

7. Undecidability for full-lossy EOSs

We now show that all (\leq_f, ℓ) -reachability problems for EOSs with finite $\ell \geq 1$ are undecidable. This is achieved via a reduction of reachability of 2CMs to (\leq_f, ℓ) -reachability with any given ℓ .

Fix the value of $\ell \geq 1$. Given an arbitrary 2CM $\mathcal{K} = (Q, \delta, q_0)$, a target configuration (q_f, n_1, n_2) of \mathcal{K} , and its simulating EOS $\mathfrak{E}_{\mathcal{K}} = (\hat{N}_{\mathcal{K}}, \mathcal{N}_{\mathcal{K}}, d_{\mathcal{K}}, \Theta_{\mathcal{K}})$ as in Sec. 6, we merge $\mathfrak{E}_{\mathcal{K}}$ with the lossiness-counter gadget $\mathcal{G} = (\hat{N}_{\mathcal{G}}, \mathcal{N}_{\mathcal{G}}, d_{\mathcal{G}}, \Theta_{\mathcal{G}})$ from Sec. 4. Specifically, for $\hat{N}_{\mathcal{K}} = (\hat{P}_{\mathcal{K}}, \hat{T}_{\mathcal{K}}, \hat{F}_{\mathcal{K}})$ and $\hat{N}_{\mathcal{G}} = (\hat{P}_{\mathcal{G}}, \hat{T}_{\mathcal{G}}, \hat{F}_{\mathcal{G}})$, we start with the EOS $\mathcal{M} = (\hat{M}, \mathcal{N}_{\mathcal{K}} \cup \mathcal{N}_{\mathcal{G}}, d_{\mathcal{K}} \cup d_{\mathcal{G}}, \Theta_{\mathcal{K}} \cup \Theta_{\mathcal{G}})$ where $\hat{M} = (\hat{P}_{\mathcal{K}} \cup \hat{P}_{\mathcal{G}}, \hat{T}_{\mathcal{K}} \cup \hat{T}_{\mathcal{G}}, \hat{F}_{\mathcal{K}} \cup \hat{F}_{\mathcal{G}})$.

We now add to \mathcal{M} a system net transition *enabling* together with a dedicated event $e = \langle \textit{enabling}, \emptyset \rangle$ (see Fig. 6). This transition consumes (1) ℓ tokens from *count*, (2) one from p_1 if ℓ is even, (3) one from p_2 if ℓ is odd. The transition *enabling* produces (1) one in q_0 , and (2) one object-token with empty internal marking in each of g , c_0 , and c_1 . We call the so obtained EOS \mathcal{F} . The initial marking μ_0 of \mathcal{F} is $\langle p_1, \{\{p\}\} \rangle$, as for \mathcal{G} .

Note that, along the runs of \mathcal{F} , the event $e = \langle \textit{enabling}, \emptyset \rangle$ fires at most once. Before firing e , there is no token on $\mathfrak{E}_{\mathcal{K}}$ while, after firing e , there is no token on \mathcal{G} . Let σ be a (\leq_f, ℓ) -run of \mathcal{F} . If e is not fired along σ , then σ does not reach the target marking $\mu_f = \langle (q_f, n_1, n_2) \rangle$ (which puts some token on $\mathfrak{E}_{\mathcal{K}}$). Otherwise, σ can be split into two runs σ_1 and σ_2 , such that $\sigma = \sigma_1 \xrightarrow{e} \sigma_2$. Since e consumes ℓ tokens from *count* and one from either p_1 or p_2 in \mathcal{G} , the last marking μ_1 in σ_1 has to put at least ℓ tokens on *count* and at least one token on either p_1 or p_2 , respectively.

If σ_1 is not ℓ -strong, then it is ℓ' -strong for some $\ell' < \ell$. By Lem. 10, there is some ℓ'' -strong run $\sigma \in \text{Runs}_{\ell''}(\leq_o, \mu_0)$ for some $\ell'' \leq \ell'$ such that $\mu_0 \rightsquigarrow^{\sigma_1} \lambda \geq_s \mu_1$ for some marking λ . Thus, by Lem. 9, λ puts on *count* at most $\ell'' < \ell$ tokens. Since $\lambda \geq_s \mu_1$, so does μ_1 , which is a contradiction. Thus, σ_1 is ℓ -strong.

Consequently, since σ is a (\leq_f, ℓ) -run, σ_2 must be a (perfect) $(\leq_f, 0)$ -run of $\mathfrak{E}_{\mathcal{K}}$. Also, because of the post-conditions of *enabling*, the first marking in σ_2 is $\langle (q_0, 0, 0) \rangle$. Thus, the 2CM \mathcal{K} reaches the target (q_f, n_1, n_2) from $(q_0, 0, 0)$ iff $\mathfrak{E}_{\mathcal{K}}$ has a (perfect) $(\leq_f, 0)$ -run from $\langle (q_0, 0, 0) \rangle$ to $\mu_f = \langle (q_f, n_1, n_2) \rangle$ iff \mathcal{F} exhibits an ℓ -strong (\leq_f, ℓ) -run from μ_0 to μ_f iff \mathcal{F} exhibits (\leq_f, ℓ) -run from μ_0 to μ_f . Since reachability of 2CM is undecidable, we get the next theorem.

Theorem 7. *For each finite $\ell \in \mathbb{N}$, (\leq_f, ℓ) -reachability for EOSs is undecidable.*

One can adapt this construction so as to concatenate the lossy-counter gadget \mathcal{G} with any EOS \mathfrak{E} (in place of $\mathfrak{E}_{\mathcal{K}}$ for any 2CM \mathcal{K}). If the initial marking of \mathfrak{E} contains several nested tokens, a chain of enabling events like e may be necessary to initialize it. As above, in order to fire them, the (\leq_f, ℓ) -runs of the concatenated EOS \mathcal{F} have to preliminary fire all their lossy steps. Moreover, after firing e , the intended initial marking of \mathfrak{E} is initialized and the (\leq_f, ℓ) -runs of \mathcal{F} can continue only by simulating \mathfrak{E} without any further lossy step. Thus, \mathfrak{E} reaches/covers a target marking μ_f iff \mathcal{F} (\leq_f, ℓ) -reaches/covers the same target μ_f from $\mu_0 = \langle p_1, \{\{p\}\} \rangle$. Since (perfect) $(\leq_f, 0)$ -reachability is known to be undecidable for EOSs (Th. 4.3 in [13]), one get again Th. 7. Moreover, also (perfect) $(\leq_f, 0)$ -coverability is known to be undecidable (again, Th. 4.3 in [13]). Thus, we get undecidability also for (\leq_f, ℓ) -coverability for each finite ℓ .

Theorem 8. *For each finite $\ell \in \mathbb{N}$, (\leq_f, ℓ) -coverability for EOSs is undecidable.*

8. Decidability of (\leq_f, ω) -reachability for EOSs

We now show that (\leq_f, ω) -reachability is decidable for EOSs. We use a modified semantics for full-lossy EOSs which merges standard and lossy steps.

Definition 13. *A merged-EOS (mEOS) is an EOS interpreted under the semantics induced by the step relation \rightsquigarrow where $\rightsquigarrow = \geq_f \cup \rightarrow$.*

Since EOSs and mEOSs are syntactically the same, given an EOS \mathfrak{E} , we denote by \mathfrak{E}_S the EOS \mathfrak{E} interpreted under the standard \rightarrow step relation (S stands for standard), and by \mathfrak{E}_M the EOS \mathfrak{E} interpreted under the \rightsquigarrow step relation (M stands for merged). Similarly, we denote the set of predecessors of μ in \mathfrak{E}_S by $\text{Pred}_S(\mu)$ and in \mathfrak{E}_M by $\text{Pred}_M(\mu)$. The benefit of mEOSs is that \leq_f becomes trivially compatible. This solves the major source of undecidability behind the undecidability result of \leq_f -reachability for EOSs.

Lemma 16. *\leq_f is compatible for mEOSs.*

Proof. By definition of \rightsquigarrow , if $\mu_1 \geq_f \mu_2 \rightsquigarrow \mu_3$, then either (1) $\mu_1 \geq_f \mu_2 \geq_f \mu_3$ and, by transitivity of \geq_f , also $\mu_1 \rightsquigarrow^* \mu_1 \geq_f \mu_3$, or (2) $\mu_1 \geq_f \mu_2 \rightarrow \mu_3$ and, by definition of \rightsquigarrow and reflexivity of \leq_f , also $\mu_1 \rightsquigarrow \mu_3 \geq_f \mu_3$. \square

Consequently, since \leq_f is a well-quasi order (see Th.5.2 in [13]), mEOSs with \leq_f are well-structured transition systems (WSTS; see [19]). Clearly, \leq_f is decidable. Moreover, mEOSs have the effective pred-basis property. This is because, $\uparrow \text{Pred}_M(\uparrow \mu) = \uparrow \{\mu\} \cup \uparrow \text{Pred}_S(\uparrow \mu)$ and \mathfrak{E}_S has the effective pred-basis property [13], where $\uparrow X$ denotes the upward-closure of X .⁸

Lemma 17. *For an EOS \mathfrak{E} and a nested marking μ , we have $\uparrow \text{Pred}_M(\uparrow \mu) = \uparrow \mu \cup \uparrow \text{Pred}_S(\uparrow \mu)$.*

Proof. If $\mu_1 \in \uparrow \text{Pred}_M(\uparrow \mu)$, then there are μ' and μ'' such that $\mu_1 \geq_f \mu' \rightsquigarrow \mu'' \geq_f \mu$. If $\mu' \geq_f \mu''$, then, by transitivity of \leq_f , we have that $\mu_1 \in \uparrow \mu$. If $\mu' \rightarrow \mu''$, then $\mu' \in \text{Pred}_S(\uparrow \mu)$ and, hence, $\mu_1 \in \uparrow \text{Pred}_S(\uparrow \mu)$. Vice-versa, since $\rightarrow \subseteq \rightsquigarrow$, we have $\text{Pred}_S(\mu) \subseteq \text{Pred}_M(\mu)$ and, thus, $\uparrow \text{Pred}_S(\uparrow \mu) \subseteq \uparrow \text{Pred}_M(\uparrow \mu)$. Moreover, since $\mu' \geq_f \mu$ implies $\mu' \in \text{Pred}_M(\mu)$, we have $\uparrow \{\mu'\} \subseteq \uparrow \text{Pred}_M(\uparrow \mu)$. \square

We can then apply the theory of WSTS and obtain decidability of coverability for mEOSs.

Lemma 18. *$(\leq_f, 0)$ -coverability for mEOSs is decidable.*

Since each (\leq_f, ω) -run in \mathfrak{E}_S is a $(\leq_f, 0)$ -run in \mathfrak{E}_M and vice-versa, we have that $(\leq_f, 0)$ -coverability for \mathfrak{E}_M coincides with (\leq_f, ω) -coverability for \mathfrak{E}_S , which, in turn, coincides with (\leq_f, ω) -reachability for \mathfrak{E}_S (Cor. 4). We thus obtain the following theorem.

Theorem 9. *(\leq_f, ω) -reachability is decidable for EOSs.*

Concerning the complexity of (\leq_f, ω) -reachability for EOSs, this problem extends (\leq_f, ω) -reachability for cEOSs, which is equivalent to $(\leq_f, 0)$ -coverability for cEOSs. By noting that these can encode PN coverability, we obtain a lower bound for (\leq_f, ω) -reachability for EOSs.

Theorem 10. *(\leq_f, ω) -reachability is EXPSPACE-hard for EOSs.*

9. Conclusions

We have completely charted the decidability status of all lossy-reachability problems for three lossiness relations: full-lossiness \leq_f , object-lossiness \leq_o , and system-lossiness \leq_s . The decidability landscape is summarized in Tab. 1. For cEOSs, proper lossy-reachability coincides with standard coverability under the respective lossiness quasi order. All problems for object- and system-lossy EOSs and cEOSs are undecidable. This is enabled by the fact that the orders \leq_o and \leq_s are not well-quasi orders (cf. [20]). For full-lossy EOSs, all (\leq_f, ℓ) -reachability problems are undecidable even if they do not coincide with standard coverability under \leq_f . The most interesting result is the decidability of (\leq_f, ω) -reachability for EOSs. This result follows from the fact that each quasi order \leq induces a WSTS when interpreted over (\leq, ω) -runs (cf. [7]). This problem is at least as hard as \leq_f -coverability for cEOSs, which, in turn, extends PN coverability. This yields an EXPSPACE lower-bound. The precise complexity of (\leq_f, ω) -reachability for

⁸I.e., if X is a set of markings, $\uparrow X$ is the set of markings μ such that $\mu \geq_f x$ for some $x \in X$; also, $\uparrow \mu$ denotes $\uparrow \{\mu\}$.

	Problem	\leq_f	\leq_o	\leq_s
cEOS	0-reach.	undec. (Th 5.5 [13])	undec. (Th 5.5 [13])	undec. (Th 5.5 [13])
	cover.	dec. (Th 5.2 [13])	undec. [2CM]	undec. [2CM]
	ℓ -reach./cover. for $\ell \in \mathbb{N}_0$	dec. [comp.]	undec. [comp.]	undec. [comp.]
	ω -reach./cover	dec. [comp.]	undec. [comp.]	undec. [comp.]
EOS	0-reach	undec. (Th 4.3 [13])	undec. (Th 4.3 [13])	undec. (Th 4.3 [13])
	cover.	undec. (Th 4.3 [13])	undec. [cEOS]	undec. [cEOS]
	ℓ -reach./cover. for $\ell \in \mathbb{N}_0$	undec. [\mathcal{G}]	undec. [cEOS]	undec. [comp.]
	ω -reach./cover	dec. [WSTS]	undec. [cEOS]	undec. [comp.]

Table 1

Decidability status of lossy problems for full-, object-, and system-lossy EOS and cEOS. \mathbb{N}_0 denotes $\mathbb{N} \setminus \{0\}$. References are put next to already known results. The labels next to our results indicate the techniques used to obtain them: [comp.] - compatibility; [2CM] - 2CM reachability; [cEOS] - generalization of cEOS results; [\mathcal{G}] - lossiness-counter gadget \mathcal{G} merging; [WSTS] - WSTS theory.

EOSs and, to the best of our knowledge, of \leq_f -coverability for cEOSs, remains uncharted. We aim to fill this gap in future works.

Decidability of (\leq_f, ω) -reachability enables, in principle, the analysis of EOS models, e.g., of business processes where resources may be lost both at the system and object level. However, where undecidability applies, we may still perform verification by employing partial procedures, e.g., by resorting to bounded model checking approaches. Recently, in [16], a Maude encoding of EOSs was proposed and reachability searches on a bounded EOS were performed. However, to the best of our knowledge, there is no tool that natively addresses (bounded) model checking of lossy EOSs. Such a tool should also be able to express formulas about the number and distribution of the lossy steps in the runs. This feature is reminiscent of program definitions in the temporal operators of Dynamic Propositional Logic (PDL) [21, 22]. Interestingly, recent Answer Set Programming tools [23], such as Telingo [24], support PDL constraints, which may be used to perform bounded model checking. A recent prototype [25] is discussed in [26]. The development of such a tool would enable us to practically verify the robustness of EOS models.

Another direction for further studies is the analysis of *reset* versions \leq_f , \leq_o , and \leq_s , where, for each place, at each step either no or all tokens on the place are lost. The *reset object-lossy* case is especially interesting in light of [11], where break-downs are modeled precisely by induced deadlocks of object-tokens via the loss of all tokens (reset) in the object. However, the technique we used to show undecidability of (ℓ, \leq_f) -reachability for finite ℓ and EOSs can be seamlessly applied to reset object- and full lossiness. Moreover, since these *reset* relations are not compatible (even for cEOSs) nor well-quasi orders, we expect that all of these problems (even ω -reachability for cEOSs) are undecidable. Further interesting variants of lossy relations include different types of imperfections such as spontaneous token moves, duplications, and insertions.

When compared to lossy EOSs, lossy PNs exhibit a much more restricted landscape. This is mainly due to the fact that the absence of nesting results in only one lossiness relation (which removes regular tokens from the PN). Moreover, by compatibility, its hierarchy collapses to two decidable problems, i.e., standard reachability (non-elementary) and standard coverability

(EXSPACE-complete). This picture is distinct even from that of full-lossy cEOSs, where, even if $(\leq_f, 0)$ -coverability is decidable, standard reachability is undecidable. Moreover, it is likely that $(\leq_f, 0)$ -coverability for cEOSs is harder than coverability for PNs. It would also be interesting to compare EOSs with (likely) more expressive models of computation, such as lossy counter machines (LCM) [7]. To that end, several additional verification problems are relevant, e.g., those studied in [14] for LCM. However, the number of lossy step parameters was not considered there and, thus, a preliminary study on LCM robustness seems necessary. Nevertheless, assuming that problems with ω lossy steps are always easier than those with a finite number, it is likely that all undecidability results in [14] also hold for their robustness version. This leaves little room for decidability results on the robustness of LCM.

References

- [1] D. Brand, P. Zafiropulo, On communicating finite-state machines, *J. ACM* 30 (1983) 323–342. URL: <https://doi.org/10.1145/322374.322380>.
- [2] P. Chambart, P. Schnoebelen, Mixing lossy and perfect fifo channels, in: *International Conference on Concurrency Theory*, volume 5201 of *LNCS*, 2008, pp. 340–355. URL: https://doi.org/10.1007/978-3-540-85361-9_28.
- [3] C. Aiswarya, On network topologies and the decidability of reachability problem, in: *International Conference on Networked Systems*, volume 12129 of *LNCS*, 2020, pp. 3–10. URL: https://doi.org/10.1007/978-3-030-67087-0_1.
- [4] M. L. Minsky, *Computation: finite and infinite machines*, Prentice-Hall, Inc., USA, 1967.
- [5] N. Dershowitz, Let’s be honest, *Commun. ACM* 64 (2021) 37–41. URL: <https://doi.org/10.1145/3431281>.
- [6] R. Mayr, *Lossy Counter Machines*, Technical Report, 1998. URL: <https://mediatum.ub.tum.de/1094480>.
- [7] R. Mayr, Undecidable problems in unreliable computations, *Theoretical Computer Science* 297 (2003) 337–354. doi:10.1016/S0304-3975(02)00646-1.
- [8] A. Bouajjani, R. Mayr, Model checking lossy vector addition systems, in: *Annual Symposium on Theoretical Aspects of Computer Science*, volume 1563 of *LNCS*, 1999, pp. 323–333. URL: https://doi.org/10.1007/3-540-49116-3_30.
- [9] J. Esparza, Decidability of model checking for infinite-state concurrent systems, *Acta Informatica* 34 (1997) 85–107. URL: <https://doi.org/10.1007/s002360050074>.
- [10] M. Baldoni, C. Baroglio, R. Micalizio, Fragility and robustness in multiagent systems, in: *International Workshop on Engineering Multi-Agent Systems*, volume 12589 of *LNCS*, 2020, pp. 61–77. URL: https://doi.org/10.1007/978-3-030-66534-0_4.
- [11] M. Köhler-Bussmeier, L. Capra, Robustness: A natural definition based on nets-within-nets, in: *International Workshop on Petri Nets and Software Engineering*, volume 3430 of *CEUR Workshop Proceedings*, 2023, pp. 70–87. URL: <https://ceur-ws.org/Vol-3430/paper5.pdf>.
- [12] R. Valk, Object Petri nets: Using the nets-within-nets paradigm, in: *Advances in Petri Nets*, volume 3098 of *LNCS*, 2003, pp. 819–848. URL: https://doi.org/10.1007/978-3-540-27755-2_23.

- [13] M. Köhler-Bußmeier, A survey of decidability results for elementary object systems, *Fundamenta Informaticae* 130 (2014) 99–123. URL: <https://doi.org/10.3233/FI-2014-983>.
- [14] P. Schnoebelen, Lossy counter machines decidability cheat sheet, in: *International Workshop on Reachability Problems*, volume 6227 of *LNCS*, 2010, pp. 51–75. URL: https://doi.org/10.1007/978-3-642-15349-5_4.
- [15] T. Murata, Petri nets: Properties, analysis and applications, *IEEE* 77 (1989) 541–580. doi:10.1109/5.24143.
- [16] L. Capra, M. Köhler-Bussmeier, Modelling adaptive systems with nets-within-nets in maude, in: *International Conference on Evaluation of Novel Approaches to Software Engineering*, 2023, pp. 487–496. URL: <https://doi.org/10.5220/0011860000003464>.
- [17] I. A. Lomazova, Nested Petri nets - a formalism for specification and verification of multi-agent distributed systems, *Fundamenta Informaticae* 43 (2000) 195–214. doi:10.3233/FI-2000-43123410.
- [18] M. Köhler, Reachable markings of object Petri nets, *Fundamenta Informaticae* 79 (2007) 401–413.
- [19] A. Finkel, P. Schnoebelen, Well-structured transition systems everywhere!, *Theoretical Computer Science* 256 (2001) 63–92. URL: <https://www.sciencedirect.com/science/article/pii/S030439750000102X>, iSS.
- [20] S. Lasota, Decidability border for Petri nets with data: WQO dichotomy conjecture, in: *Application and Theory of Petri Nets and Concurrency*, volume 9698 of *LNCS*, 2016, pp. 20–36. URL: https://doi.org/10.1007/978-3-319-39086-4_3.
- [21] M. J. Fischer, R. E. Ladner, Propositional dynamic logic of regular programs, *J. Comput. Syst. Sci.* 18 (1979) 194–211. URL: [https://doi.org/10.1016/0022-0000\(79\)90046-1](https://doi.org/10.1016/0022-0000(79)90046-1).
- [22] P. Balbiani, E. Lorini, Ockhamist propositional dynamic logic: A natural link between PDL and CTL, in: *Logic, Language, Information, and Computation Proceedings*, volume 8071 of *LNCS*, 2013, pp. 251–265. URL: https://doi.org/10.1007/978-3-642-39992-3_22.
- [23] M. Gebser, R. Kaminski, B. Kaufmann, T. Schaub, Multi-shot ASP solving with clingo, *Theory Pract. Log. Program.* 19 (2019) 27–82. URL: <https://doi.org/10.1017/S1471068418000054>.
- [24] P. Cabalar, Temporal ASP: from logical foundations to practical use with telingo, in: *Reasoning Web. Declarative Artificial Intelligence*, volume 13100 of *LNCS*, 2021, pp. 94–114. URL: https://doi.org/10.1007/978-3-030-95481-9_5.
- [25] T. Prince, F. Di Cosmo, Nets within nets telingo analyser, 2024. doi:10.5281/zenodo.11401876.
- [26] F. Di Cosmo, T. Prince, Bounded verification of petri nets and EOSs using Telingo: an experience report, in: *CILC’24: 39th Italian Conference on Computational Logic*, June 26-28, 2024, Rome, Italy, 2024. To appear.