mathematics

MDPI

Article

Language-Based Opacity Verification in Partially Observed Petri Nets through Linear Constraints

Ikram Saadaoui [1,2,†], Abdeljalil Labed [2,†], Zhiwu Li [1,*,†], Ahmed M. El-Sherbeeny [3,†] and Huiran Du [4,†]

[1] Institute of Systems Engineering, Macau University of Science and Technology, Taipa, Macau SAR 999078, China; ikram.saadaoui@medtech.tn
[2] Mediterranean Institute of Technology, South Mediterranean University, Tunis 99628, Tunisia; abdeljalil.labed@medtech.tn
[3] Industrial Engineering Department, College of Engineering, King Saud University, P.O. Box 800, Riyadh 11421, Saudi Arabia; aelsherbeeny@ksu.edu.sa
[4] Hitachi Building Technology (Guangzhou) Co., Ltd., Guangzhou 510700, China; duhuiran@hitachi-helc.com
* Correspondence: zwli@must.edu.mo
† These authorscontributed equally to this work.

Abstract: Information security is an important area of concern in modern computer-integrated systems. It involves implementing preventative measures to protect confidential data from potential vulnerabilities, such as unauthorized access, secret disclosure, modification, or destruction. Considering such threats, we investigate a particular confidentiality property called opacity, which specifies a system's ability to cover its 'secret' data from being interfered with by outside observers, termed as intruders. This paper discusses language-based opacity formulation and verification in the context of discrete event systems represented by partially observed Petri nets. In this context, we identify two opacity properties, called consistency and non-secrecy; then, we exploit the mathematical characterization of a net system, to separately check each property, by specifying two feasibility problems. The proposed method is carried out for two distinct settings of a system. The first setting is centralized, where an intruder is granted complete information about the system structure but a partial observation of its behavior. The second setting is decentralized, where a group of intruders cooperates to reveal the secret language, by using a coordinator. Finally, experimental findings are given, to demonstrate the proficiency of the proposed approach.

Keywords: discrete event system; opacity; Petri net; sensor configuration; linear constraints; integer linear programming

MSC: 93E99

Citation: Saadaoui, I.; Labed, A.; Li, Z.; El-Sherbeeny, A.M.; Du, H. Language-Based Opacity Verification in Partially Observed Petri Nets through Linear Constraints. *Mathematics* **2023**, *11*, 3880. https://doi.org/10.3390/math11183880

Academic Editor: Vasyl Martsenyuk

Received: 7 August 2023
Revised: 31 August 2023
Accepted: 8 September 2023
Published: 11 September 2023

1. Introduction

The growing number of modern cyber-physical systems involving critical information, such as defense, health care, banking, and communication systems, emphasizes the need to establish measures to ensure their security against hostile actions. These systems are particularly subject to unauthorized access, because the risk of information leakage to unauthorized users may reveal sensitive details about their behavior.

Both the industry and the research communities have recently stressed the importance of security properties. These properties are primarily categorized into three groups: confidentiality, availability, and integrity, which guide policies for information security within systems and organizations. Motivated by security concerns about discrete event systems (DESs), our study concentrates on a specific confidentiality property, called opacity. This property describes the capability of preventing a hostile observer, referred to as an intruder, from inferring whether or not a system's secret behavior has occurred. Numerous aspects of opacity have recently been investigated in the computer security community.

Due to their competency for system modeling, Petri nets under partial observations have been widely used in cyber-physical systems. In the area of DESs, there has been a deluge of studies on fault diagnosis, deadlock control [1], and supervisory control [2] based on Petri nets. One significant feature of a Petri net (PN) is its mathematical formalism that allows the use of integer linear programming (ILP), which can alleviate the state explosion problem, to some extent. For this reason, we use a Petri net as a modeling framework in this study.

Opacity is considered as a general information flow property that covers a wide range of applications in DESs. Bryans et al. [3] proved that certain information flow properties, such as anonymity and non-interference [4] problems, can be converted into opacity, by using appropriate observation mappings. In addition, Lin et al. [5] showed that observability, detectability, and diagnosability [6] can be conceived as opacity problems.

Opacity in DESs is categorized into two main groups, based on the secret's representation: state-based opacity [7], where a secret is specified as a subset of the state space, and language-based opacity (LBO) [8], where a secret is specified as a subset of firable transition sequences.

This study considers the language-based opacity verification problem in DESs represented by partially observed Petri nets (POPNs), where a secret is defined as a finite sub-language of a PN system, and an intruder is allowed to possess complete knowledge of the system's structure, but can only identify the firing of visible transitions and (or) the tokens variation in visible places. Language-based opacity was initially introduced to DESs modeled with FSA in [5,9], and it was extended recently to bounded labeled Petri net systems (LPNs) in [10]. In [5], Lin et al. specified two classes of opacity, called strong and weak opacity. A strongly (resp. weakly) opaque language is a language where strings from a different language camouflage all (resp. some) of its strings. Specifically, given a regular system language and a state-based projection information mapping, the authors in [5] proposed algorithms with exponential complexity, to verify both weak and strong opacity. In the current study, language-based opacity is attributed to strong opacity. In [10], Tong et al. studied a special case of LBO, called strict language opacity, in a bounded PN system. This work was based on the assumption that a secret is defined as a subset of firable sequences of transitions and an intruder who is interested in observable transitions only. To check the strict language opacity property, Tong et al. built a finite automaton, called a verifier, which synchronized the PN system and the secret language, based on observable transitions. This approach was time-consuming, because it required off-line construction of the verifier, which suffered an exponential space complexity. In [11], Cong et al. also established a necessary and sufficient condition for current state opacity (CSO) verification in LPN through integer linear programming. However, their method cannot be directly used for LBO verification, unless we convert the CSO problem into an LBO problem, using the transformation method given in [12]. Furthermore, the research in [11] only considered secret markings defined by a set of generalized mutual exclusion constraints (GMECs), which restricts the scope of secret selection.

Another interesting work is found in [13], which investigated the verification problem of LBO in LPNs, where the unobservable subnet was considered to be acyclic, and the secret was specified across sequences of events rather than transitions. The study in [13] was closely related to this particular research, in which the authors solved an integer linear programming problem (ILPP), to establish a condition that was both sufficient and necessary for LBO verification. The technique proposed in this study is more general, as it applies to a greater range of PN systems under partial observations (i.e., PNs equipped with place and transition sensors).

Nowadays, intruders tend to be highly skilled and intelligent. They are often part of an organized group providing illegal specialized services, such as credit card fraud, theft of intellectual property, or counterfeiting documents. This paper investigates the LBO verification problem in a decentralized setting, where local intruders collaborate by using a

coordinator. Each local intruder has complete information about the net structure but only partial information about its evolution.

Decentralized opacity can be reduced to many security properties, such as decentralized anonymity, secrecy, and non-interference. Specifically, consider a multi-user system that allows many users to share the resources of a single computer. In our framework, these users can be viewed as intruders who should not determine whether or not the decentralized security property is met. Furthermore, many properties employed in discrete event systems for supervisory control can be restated as a particular instance of decentralized opacity. Paoli et al. [14] showed that co-opacity is an extension of co-observability [15] used for supervisory control [16], and that it can be used to check the existence of a group of decentralized local supervisors (intruders in our framework, with or without a coordinator) that control a system. Tripakis and Rudie [17] developed a decidable condition called "at least one can tell", which ensures that decentralized agents can make correct decisions about the behavior of a system. This condition is relevant to the decentralized opacity problem, because it provides a way to verify whether or not a decentralized system is opaque.

This research aims to compose the LBO verification problem in the context of a DES represented by a POPN, taking into account observations from place and transition sensors. Then, based on the proposed formulation, new approaches to verifying LBO are set up. In light of the preceding, the findings of this study contribute the following to the existing literature:

- We define the concept of secret observation sequences, by considering a case where a secret involves numerous transition sequences yielding the same observation. This concept decreases the effort required to check language opacity in a POPN, by avoiding redundant explorations.
- We identify two language-based opacity properties, called consistency and non-secrecy properties; we separately check each property; then, we provide necessary and sufficient conditions to check language-based opacity, by defining and solving an ILPP.
- Finally, we extend the above results from a centralized framework to a decentralized framework with a coordinator, where more than one intruder observes the system.

The following is a breakdown of the paper's structure. Section 2 provides an overview of Petri nets and partially observed Petri nets. The theory underlying the partition of unobservable transitions, as well as the concept of discernible transitions, is presented in Section 3. The LBO verification problem is formulated in a POPN system in Section 4. An ILP problem is established in Section 5, to solve the LBO verification problem. In Section 7, we discuss opacity in a decentralized case, where a group of local intruders unites to reveal a secret language via a coordinator. To demonstrate the proposed method, an example is given in Section 6. Finally, Section 8 brings the study to a close and points the way forward for further research.

2. Preliminaries

2.1. Petri Nets

A Petri net structure is a weighted bipartite graph $N = (P, T, Pre, Post)$, where $P = \{p_1, p_2, \ldots, p_m\}$ is a set of m places and $T = \{t_1, t_2, \ldots, t_n\}$ is a set of n transitions with $P \cup T \neq \emptyset$ and $P \cap T = \emptyset$. $Pre : P \times T \to \mathbb{N}$ and $Post : P \times T \to \mathbb{N}$ ($\mathbb{N}$ denotes the set of non-negative integer numbers) are the pre- and post-incidence functions designating arcs from places to transitions and from transitions to places, respectively, in a net, and they are represented as matrices in $\mathbb{N}^{m \times n}$. The incidence matrix of a PN is defined by $C = Post - Pre$.

A marking is a mapping $M : P \to \mathbb{N}$ that attributes to each place a non-negative integer number of tokens. A PN system with initial marking M_0 is denoted by a couple (N, M_0).

A transition t_i is enabled at a marking M, denoted by $M[t_i\rangle$, if $M \geq Pre(\cdot, i)$, where $Pre(\cdot, i)$ is the ith column of matrix Pre. Firing an enabled transition yields a marking M' with

$$M' = M + C(\cdot, t), \tag{1}$$

where $C(\cdot, t)$ is a column vector that denotes the token change generated by firing t. Given a transition sequence $\sigma \in T^*$, $|\sigma|$ denotes the length of σ. Let $\pi : T^* \to \mathbb{N}^n$ be a function that allocates a Parikh vector $y \in \mathbb{N}^n$ to a transition sequence $\sigma \in T^*$, called the firing vector of σ. The notations $M[\sigma\rangle$ and $M[\sigma\rangle M'$ indicate, respectively, that σ is enabled at M and the firing of σ at M yields M', following the equation:

$$M' = M + C \cdot \pi(\sigma). \tag{2}$$

Equation (2) is called the state equation of (N, M_0). We denote by $L(N, M_0) = \{\sigma \in T^* | M_0[\sigma\rangle\}$ the set of all firable transition sequences in (N, M_0), and by $R(N, M_0)$ the reachability set of (N, M_0). A PN is said to be acyclic if there is no directed circuit and bounded if there exists a positive constant $k \in \mathbb{N}$, such that, for all $M \in R(N, M_0)$ and $p \in P$, $M(p) \leq k$, where $M(p)$ is the tokens number in place p.

2.2. Partially Observed Petri Nets

A partially observed Petri net is a PN system equipped with place sensors that display the number of tokens in some places, known as observable places, and (or) transition sensors that display the labels of observable transitions, when fired. In this sense, POPNs can be seen as a generalization of LPNs, since any LPN can be represented as a POPN. However, not all POPNs can be represented as LPNs. This is because POPNs allow for the presence of place sensors, which LPNs do not.

The set of observable places is denoted by $P_o \subseteq P$. We re-index observable places in P_o from 1 to m_o, such that $P_o = \{p_{o_1}, p_{o_2}, \ldots, p_{o_{m_o}}\}$ with $o_i \in \{1, 2, \ldots, m\}$ and $p_{o_i} \in P$ for $i \in \{1, 2, \ldots, m_o\}$. A partially observed Petri net is a quintuple-tuple $Q = (N, M_0, E, V, \delta)$, where (N, M_0) is a PN system with m places and n transitions, E is an alphabet (a set of labels), and $V = (v_{ij}) \in \{0, 1\}^{m_o \times m}$ is a place sensor configuration matrix, where $v_{ij} = 1$ if $j = o_i$ and $v_{ij} = 0$, otherwise. A labeling function, $\delta : T \to E \cup \{\varepsilon\}$, represents the transition sensor configuration, which allocates a label from E or the empty string ε to a transition $t \in T$. Based on these allocations, we divide the set of transitions T into two disjoint sets T_o and T_u, satisfying $T = T_o \cup T_u$ and $T_o \cap T_u = \emptyset$, where $T_o = \{t \in T | \delta(t) \in E\}$ is the set of observable transitions and $T_u = \{t \in T | \delta(t) = \varepsilon\}$ is the set of unobservable transitions. Thus, a label $\delta(t)$ is displayed only when we fire a transition $t \in T_o$. We denote by $\hat{M} = V \cdot M$ the marking measurement of M using a place sensor configuration matrix V. Arguably, matrix V characterizes a marking M projection on P_o.

Definition 1. *Let (N, M_0) be a PN system with $N = (P, T, Pre, Post)$. An evolution of N starting from M_0 is denoted as a transition-marking sequence.*

$$M_0 t_1 M_1 t_2 M_2 \ldots t_h M_h,$$

satisfying $M_0[t_1\rangle M_1[t_2\rangle M_2 \ldots [t_h\rangle M_h$, where $t_i \in T$, $M_i \in R(N, M_0)$ for $i \in \{1, \ldots, h\}, h \geq 1$.

Definition 2. *Given a POPN $Q = (N, M_0, E, V, \delta)$, the collected measure associated with $M[t\rangle M'$ is given as follows:*

$$\rho(M, t) = \begin{cases} \varepsilon_p, & \text{if } (\hat{M} = \hat{M}') \wedge (\delta(t) = \varepsilon) \\ \hat{M}\delta(t)\hat{M}', & \text{otherwise,} \end{cases}$$

where $\hat{M} = V \cdot M$, $\hat{M}' = V \cdot M'$ and ε_p denotes an empty observation.

We denote by $\rho(M,\sigma)$ the extension of the operation ρ to transition sequences. Specifically, let $Mt_1M_1t_2M_2\dots M_{h-1}t_hM_h$ be the evolution produced by firing $\sigma = t_1t_2\dots t_h \in T^*$ at M. We define the associated measurement sequence by concatenating the successive collected measures, as seen below:

$$\begin{aligned}\rho(M,\sigma) &= \rho(M,t_1)\rho(M_1,t_2)\dots\rho(M_{h-1},t_h)\\ &= \hat{M}e_1\hat{M}_1\hat{M}_1e_2\dots\hat{M}_{ho-1}\hat{M}_{ho-1}e_{ho}\hat{M}_{ho},\end{aligned}$$

where $e_i \in E\cup\{\varepsilon\}$ and $ho \le h$. Given a measurement sequence $\hat{M}_0e_1\dots\hat{M}_i\hat{M}_i\dots e_{ho}\hat{M}_{ho}$, $1 \le i \le ho-1$, the operation Λ fuses each two adjacent $\hat{M}_i$ as follows:

$$\Lambda(\hat{M}_0e_1\dots\hat{M}_i\hat{M}_i\dots e_{ho}\hat{M}_{ho}) = \hat{M}_0e_1\dots\hat{M}_i\dots e_{ho}\hat{M}_{ho}.$$

Definition 3. *Let $Q = (N, M_0, E, V, \delta)$ be a POPN, M be a marking from $R(N,M_0)$, and $\sigma = t_1t_2\dots t_h \in T^*$ be a transition sequence, such that $M[\sigma\rangle$. The observation sequence generated when σ fires at M is*

$$\begin{aligned}w(M,\sigma) &= \Lambda(\rho(M,\sigma))\\ &= \Lambda(\hat{M}e_1\hat{M}_1\hat{M}_1e_2\dots\hat{M}_{ho-1}\hat{M}_{ho-1}e_{ho}\hat{M}_{ho})\\ &= \hat{M}e_1\hat{M}_1e_2\dots\hat{M}_{ho-1}e_{ho}\hat{M}_{ho}.\end{aligned}$$

Marking measurements are displayed by sensors whenever observable events fire or when token variations occur in observable places. After obtaining $\rho(M,\sigma)$, each two adjacent measures $\hat{M}_i\hat{M}_i$ are merged into one measure, using the operation Λ. For simplification purposes, if no confusion is caused, we use the symbol w to designate an observation sequence in a POPN.

Example 1. *Consider a POPN in Figure 1 with $M_0 = [1\,0\,0\,0\,0\,0]^T$. The place sensor configuration is derived from $P_o = \{p_1,\ p_5\}$ as follows:*

$$V = \begin{array}{c} \\ p_{o_1} \\ p_{o_2} \end{array} \begin{array}{c} \begin{array}{cccccc} p_1 & p_2 & p_3 & p_4 & p_5 & p_6 \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \end{array}.$$

We have $T_o = \{t_1, t_4\}$ and $T_u = \{t_2, t_3, t_5\}$. Given $E = \{a, b\}$, we define the event sensor configuration using the labeling function δ as $\delta(t_1) = a$, $\delta(t_4) = b$, and $\delta(t_2) = \delta(t_3) = \delta(t_5) = \varepsilon$. Let $\sigma = t_1t_3t_4$ be a transition sequence that is enabled at M_0. By Definition 1, $[1\,0\,0\,0\,0\,0]^Tt_1[0\,1\,1\,0\,1\,0]^Tt_3[0\,1\,0\,0\,1\,1]^Tt_4[0\,0\,0\,1\,1\,1]^T$ is the evolution generated by firing σ at M_0. We have $\rho([0\,1\,1\,0\,1\,0]^T, t_3) = \varepsilon_p$; by Definition 3, the associated observation sequence is $\Lambda(\rho(M_0,\sigma)) = [1\,0]^Ta[0\,1]^Tb[0\,1]^T$.

Let $Q = (N, M_0, E, V, \delta)$ be a POPN and M be a marking from $R(N,M_0)$. We define by

$$\mathcal{O}(Q,M) = \{w \mid \exists\sigma \in T^*,\ M[\sigma\rangle,\ \Lambda(\rho(M,\sigma)) = w\}$$

the set of feasible observation sequences generated by Q from M, and by

$$\mathcal{S}(w) = \{\sigma \in T^* \mid M_0[\sigma\rangle,\ \Lambda(\rho(M_0,\sigma)) = w\}$$

the set of transition sequences consistent with w.

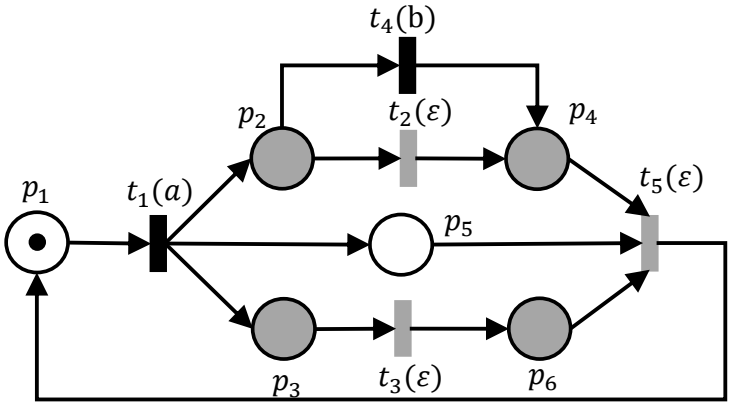

Figure 1. A POPN, where unobservable transitions and places are depicted by gray bars and circles, respectively.

3. Quasi-Unobservable and Discernible Transitions

Let Q be a POPN with a sensor configuration (V, δ) and $C_o \in \mathbb{N}^{m_o \times n}$ be the restriction of C to P_o. Based on C_o, the set of unobservable transitions T_u can be partitioned into two disjoint sets $\tilde{T}_q$ and $\tilde{T}_u$, such that $T_u = \tilde{T}_q \cup \tilde{T}_u$ and $\tilde{T}_q \cap \tilde{T}_u = \emptyset$, where $\tilde{T}_q = \{t \in T_u | \ C_o(\cdot, t) \neq \vec{\mathbf{0}}_{m_o}$ ($\vec{\mathbf{0}}_{m_o}$ is an m_o-dimensional column vector with all entries being 0) $\}$ is a set of quasi-observable transitions and $\tilde{T}_u = \{t \in T_u | \ C_o(\cdot, t) = \vec{\mathbf{0}}_{m_o}\}$ is a set of truly unobservable transitions. We define $T_d = T_o \cup \tilde{T}_q$ as the set of discernible transitions. For $e \in E \cup \{\varepsilon\}$, $T(e) = \{t \in T | \delta(t) = e\}$ and $T_d(e) = \{t \in T_d | \delta(t) = e\}$, respectively, represent the subsets of transitions in T and T_d, having the same label, e. We denote by $\Gamma_e = \{\vec{v}_t | \vec{v}_t = C_o(\cdot, t), t \in T(e)\}$ the set of column vectors in matrix C_o associated with transitions in $T(e)$.

Example 2. *Consider the POPN with the sensor configuration (V, δ) specified in Example 1. Matrix C_o is given as follows:*

$$C_o = \begin{array}{c} \\ p_1 \\ p_5 \end{array} \begin{array}{c} \begin{array}{ccccc} t_1(a) & t_2(\varepsilon) & t_3(\varepsilon) & t_4(b) & t_5(\varepsilon) \end{array} \\ \begin{bmatrix} -1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & -1 \end{bmatrix} \end{array}.$$

It is evident that $\tilde{T}_u = \{t_2, t_3\}$, $\tilde{T}_q = \{t_5\}$, and $T_d = \{t_1, t_4, t_5\}$.

Definition 4. *Let $Q = (N, M_0, E, V, \delta)$ be a POPN and $\tilde{T}_u$ be the set of its truly unobservable transitions. A truly unobservable subnet of Q is defined as $\tilde{Q} = (\tilde{N}, E, M_0, V, \delta)$, where $\tilde{N} = (P, \tilde{T}_u, \tilde{Pre}_u, \tilde{Post}_u)$, with $\tilde{Pre}_u$ and $\tilde{Post}_u$ being, respectively, the restrictions of Pre and Post on $\tilde{T}_u$. A discernible subnet of Q is defined as $Q_d = (N_d, M_0, E, V, \delta)$, where $N_d = (P, T_d, Pre_d, Post_d)$, with Pre_d and $Post_d$ being, respectively, the restrictions of Pre and Post on T_d. $\tilde{C}_u$ and C_d, respectively, denote the incidence matrices of $\tilde{Q}$ and Q_d. Let $|\tilde{T}_u| = \tilde{n}_u$ and $|T_d| = n_d$.*

4. Language-Based Opacity

We assume that an intruder is only interested in a limited number of secret sequences, i.e., the secret is a finite sub-language of the net system.

Definition 5. *Let $Q = (N, M_0, E, V, \delta)$ be a POPN and $L_s \subseteq L(N, M_0)$ be a finite secret language. Q is said to be language-opaque, with regard to L_s, if, for all $\sigma \in L_s$, there exists a sequence $\sigma' \in L(N, M_0) \setminus L_s$, such that $\Lambda(\rho(M_0, \sigma)) = \Lambda(\rho(M_0, \sigma'))$.*

Definition 6. *Given a POPN $Q = (N, M_0, E, V, \delta)$ and a finite secret language $L_s \subseteq L(N, M_0)$, the set of secret observation sequences associated with L_s is defined as*

$$\mathcal{O}(L_s) = \{w | \exists \sigma \in L_s,\ \Lambda(\rho(M_0, \sigma)) = w\}.$$

Let $w \in \mathcal{O}(L_s)$ be a secret observation sequence. The set of secret transition sequences consistent with w is given by

$$\mathcal{L}_s(w) = \{\sigma \in L_s | M_0[\sigma\rangle,\ \Lambda(\rho(M_0, \sigma)) = w\}.$$

Note that each $w \in \mathcal{O}(L_s)$ can be associated with at least one transition sequence from L_s, i.e., $|\mathcal{L}_s(w)| \geq 1$. Knowing that L_s is finite, then $\mathcal{O}(L_s)$ is also finite, with $|\mathcal{O}(L_s)| \leq |L_s|$.

Now we reformulate the definition of LBO in a POPN, by exploiting the concept of secret observation sequences.

Definition 7. *Let $Q = (N, M_0, E, V, \delta)$ be a POPN and $L_s \subseteq L(N, M_0)$ be a finite secret language. An observation sequence $w \in \mathcal{O}(L_s)$ is language-opaque, with regard to L_s, if and only if, at least, there is a sequence $\sigma \in L(N, M_0) \setminus L_s$, such that $\Lambda(\rho(M_0, \sigma)) = w$, i.e., $\mathcal{S}(w) \nsubseteq L_s$ holds.*

Based on Definition 7, a POPN system is language-opaque, with regard to a secret language L_s, if for any $w \in \mathcal{O}(L_s)$ there exists a transition sequence σ satisfying the following two properties:

- Consistency property: $\sigma \in \mathcal{S}(w)$;
- Non-secrecy property: $\sigma \notin L_s$.

Proposition 1. *Let $Q = (N, M_0, E, V, \delta)$ be a POPN and $L_s \subseteq L(N, M_0)$ be a secret language. Q is language-opaque, with regard to L_s, if and only if, for any $w \in \mathcal{O}(L_s)$, w is language-opaque, with regard to L_s.*

Proof. (If) Suppose that for any $w \in \mathcal{O}(L_s)$, w is LBO, with respect to L_s. Thus, for any $w \in \mathcal{O}(L_s)$ (i.e., $\sigma \in L_s$), there is at least a sequence $\sigma' \in L(N, M_0) \setminus L_s$, such that $\Lambda(\rho(M_0, \sigma')) = \Lambda(\rho(M_0, \sigma)) = w$. Based on Definition 5, Q is language-opaque, with regard to L_s.

(Only if) By contradiction, assume that Q is language-opaque, with respect to L_s, and that there exists an observation sequence $w \in \mathcal{O}(L_s)$ such that w is not language-opaque, with regard to L_s, i.e., $\mathcal{S}(w) \subseteq L_s$. In other words, for $\sigma \in \mathcal{S}(w) \subseteq L_s$, there does not exist $\sigma' \in L(N, M_0) \setminus L_s$, such that $\Lambda(\rho(M_0, \sigma')) = \Lambda(\rho(M_0, \sigma)) = w$. Thus, following Definition 5, Q is not LBO, which opposes the assumption. □

If a secret observation sequence w has several consistent transition sequences from L_s, i.e., $|\mathcal{L}_s(w)| > 1$, it is more efficient to verify the opacity of w instead of investigating LBO separately for each $\sigma \in \mathcal{L}_s(w)$. Using this proposition, we put our focus on a portion of the reachability space that allows checking the secret language occurrence $\mathcal{O}(L_s)$, rather than checking the entire PN system language.

Example 3. *Consider the POPN in Figure 1. Let $L_s = \{t_1t_3t_4, t_1t_3t_2t_5, t_1t_2t_3t_5\}$ be a secret language. Its associated secret observation sequences are given by $\mathcal{O}(L_s) = \{w_1, w_2\}$, where $w_1 = [1\ 0]^T a [0\ 1]^T b [0\ 1]^T$, $w_2 = [1\ 0]^T a [0\ 1]^T [1\ 0]^T$, $\mathcal{L}_s(w_1) = \{t_1t_3t_4\}$, and $\mathcal{L}_s(w_2) = \{t_1t_3t_2t_5, t_1t_2t_3t_5\}$. We have $\mathcal{S}(w_1) = \{t_1t_3t_4, t_1t_4t_3\}$ and $\mathcal{S}(w_2) = \{t_1t_2t_3t_5, t_1t_3t_2t_5\}$. We observe that $\mathcal{S}(w_1) \nsubseteq L_s$, implying that w_1 is LBO, with respect to L_s. However, we can see that $\mathcal{S}(w_2) \subseteq L_s$. Hence, based on Definition 7, w_2 is not LBO, which, based on Proposition 1, implies the non-language opacity of Q.*

5. Mathematical Characterization of LBO

In this section, we suggest a linear algebraic characterization, to check LBO in a POPN Q. The following assumption supports this characterization: The truly unobservable subnet $\tilde{Q}$ and the discernible subnet Q_d are acyclic.

Unfortunately, the majority of studies dealing with the opacity problem in PN systems [10,11,13], including the proposed approach, have a limitation, due to the assumption that the unobservable (observable) part of the Petri net is structurally acyclic, which is a strong requirement that limits their application to more general systems. This assumption ensures that the state equation is necessary and sufficient to capture the set of reachable markings by firing unobservable transitions. In this case, the mathematical characterization of a PN system using the state equation becomes a double-edge sword that reduces the exhaustive computation load on the one hand, but affects negatively the system generality on the other hand.

As far as we know, the only research that addresses the verification of language-based opacity that does not make the assumption of the unobservable subnet being acyclic is our earlier study, which we introduced in [18], where a verification algorithm employing a depth-first search approach verifies the presence of a non-secret transition sequence that is consistent with a secret observation sequence.

5.1. Consistency Assurance

With an observation sequence $w \in \mathcal{O}(Q, M_0)$ being observed, an intruder with full knowledge of the system structure tries to establish an estimation of the event sequences consistent with w. In the following proposition, we prove that each $\sigma \in \mathcal{S}(w)$ can be represented by a linear system.

Proposition 2. *Let $Q = (N, M_0, E, V, \delta)$ be a POPN and $w = \hat{M}_0 e_1 \hat{M}_1 e_2 \dots \hat{M}_{ho-1} e_{ho} \hat{M}_{ho} \in \mathcal{O}(Q, M_0)$ be an observation sequence. A transition sequence σ' is consistent with w if $\sigma' = \sigma'_{u_1}\sigma'_{e_1}\sigma'_{u_2}\sigma'_{e_2}\dots\sigma'_{u_{ho}}\sigma'_{e_{ho}}\sigma'_{u_{ho+1}}$ and its firing vectors $\vec{\sigma}'_{u_1}, \vec{\sigma}'_{u_2}, \dots, \vec{\sigma}'_{u_{ho+1}}, \vec{\sigma}'_{e_1}, \vec{\sigma}'_{e_2}, \dots, \vec{\sigma}'_{e_{ho}}$ satisfy the following constraint set $\mathfrak{C}(M_0, C_d, \tilde{C}_u, w) =$*

$$
\begin{cases}
M_0 + \tilde{C}_u \sum\limits_{i=1}^{ho+1} \vec{\sigma}'_{u_i} + C_d \sum\limits_{j=1}^{ho} \vec{\sigma}'_{e_j} \geq \vec{\boldsymbol{0}} & (a) \\
\left.\begin{array}{l} M_0 + \tilde{C}_u \sum\limits_{i=1}^{l} \vec{\sigma}'_{u_i} + C_d \sum\limits_{j=1}^{l-1} \vec{\sigma}'_{e_j} \geq Pre_d \cdot \vec{\sigma}'_{e_l} \\ l \in \{1, \dots, ho\} \end{array}\right\} & (b) \\
\left.\begin{array}{l} V \cdot \tilde{C}_u \cdot \vec{\sigma}'_{u_i} = \vec{\boldsymbol{0}} \\ V \cdot C_d \cdot \vec{\sigma}'_{ej} = \hat{M}_j - \hat{M}_{(j-1)} \end{array}\right\} & (c) \\
\left.\begin{array}{l} \sum\limits_{t \in T_d(e_j)} \vec{\sigma}'_{e_j}(t) = 1 \\ \sum\limits_{t \notin T_d(e_j)} \vec{\sigma}'_{e_j}(t) = 0 \end{array}\right\} & (d) \\
\left.\begin{array}{l} \vec{\sigma}'_{u_i} \in \mathbb{N}^{\tilde{n}_u},\ i \in \{1, \dots, ho+1\} \\ \vec{\sigma}'_{e_j} \in \mathbb{N}^{n_d},\ j \in \{1, \dots, ho\} \end{array}\right\} & (e),
\end{cases}
\tag{3}
$$

where

- $|\sigma'_{u_i}| \geq 0, i \in \{1, \dots, ho+1\}$;
- $|\sigma'_{e_j}| = 1, j \in \{1, \dots, ho\}$;
- *Constraints (3.a) represent the state equation;*
- *Constraints (3.b) represent the enabling condition of discernible transitions;*
- *Constraints (3.c) represent the token variation generated by firing σ'_{u_i} and σ'_{e_j};*
- *Constraints (3.d) define a mutual exclusion condition, to prevent firing more than one discernible transition.*

Proof. (Only if) Suppose that σ' is consistent with w, then $\sigma' = \sigma'_{u_1}\sigma'_{e_1}\sigma'_{u_2}\sigma'_{e_2}\ldots\sigma'_{u_{ho}}\sigma'_{e_{ho}}\sigma'_{u_{ho+1}}$, where $\sigma'_{e_j} \in T_d(e_j)$ with $|\sigma'_{e_j}| = 1$, $j \in \{1,\ldots,ho\}$, and $\sigma'_{u_i} \in \tilde{T}^*_u$, $i \in \{1,\ldots,ho+1\}$. Evidently, the associated firing vectors satisfy Constraints (3.a), (3.d) and (3.e).

When fired at M_0, σ' generates a trajectory:

$$M_0[\sigma'_{u_1}\sigma'_{e_1}\rangle M_1 \ldots M_{ho-1}[\sigma'_{u_{ho}}\sigma'_{e_{ho}}\rangle M_{ho}[\sigma'_{u_{ho+1}}\rangle M_{ho+1}.$$

If a discernible transition fires σ'_{e_j}, we have $\rho(M_{i-1}, \sigma'_{u_i}\sigma'_{e_j}) = \hat{M}_{i-1}e_j\hat{M}_i$. The firing of $\vec{\sigma}'_{u_i}$ at M_{i-1} enables $\vec{\sigma}'_{e_j}$ but does not produce any token variations in observable places (i.e., $V \cdot C_u \cdot \vec{\sigma}'_{u_i} = \vec{\mathbf{0}}$), while the firing of $\vec{\sigma}'_{e_j}$ generates a token variation given by $\hat{M}_i - \hat{M}_{(i-1)}$. Therefore, the enabling Constraints (3.b) and the token variation Constraints (3.c) hold.

(If) Assume that there exists σ', such that $\sigma' = \sigma'_{u_1}\sigma'_{e_1}\sigma'_{u_2}\sigma'_{e_2}\ldots\sigma'_{u_{ho}}\sigma'_{e_{ho}}\sigma'_{u_{ho+1}}$, satisfying the Constraints Set (3). By Constraints (3.a) and (3.b), σ' is enabled at M_0 and generates the following trajectory when fired at M_0:

$$M_0[\sigma'_{u_1}\sigma'_{e_1}\rangle M_1 \ldots M_{ho-1}[\sigma'_{u_{ho}}\sigma'_{e_{ho}}\rangle M_{ho}[\sigma'_{u_{ho+1}}\rangle M_{ho+1}.$$

Constraints (3.c) and (3.d) demonstrate the congruity between the transition sequences, the associated labels, and the marking measurements, which suggests that $\Lambda(\rho(M_0, \sigma')) = \hat{M}_0e_1\hat{M}_1e_2\ldots\hat{M}_{ho-1}e_{ho}\hat{M}_{ho}$. Hence, $\sigma \in \mathcal{S}(w)$. □

5.2. Non-Secrecy Assurance

To ensure the language opacity of a POPN, we simply prove that for each observation sequence w, there exists one transition sequence $\sigma' \in \mathcal{S}(w)$, such that $\sigma' \notin \mathcal{L}_s(w)$. In other words, for all $\sigma \in \mathcal{L}_s(w)$, we prove that $\sigma' \neq \sigma$. Let $z_i \in \mathbb{Z}^{\tilde{n}_u}$ and $q_j \in \mathbb{Z}^{n_d}$ be two vectors satisfying $z_i = \vec{\sigma}'_{u_i} - \vec{\sigma}_{u_i}, i \in \{1,\ldots,ho+1\}$ and $q_j = \vec{\sigma}'_{e_j} - \vec{\sigma}_{e_j}, j \in \{1,\ldots,ho\}$. An entry z_{ik} of vector z_i can be characterized as follows:

$$\begin{cases} z_{ik} = 0, & \text{if } \vec{\sigma}'_{u_i} = \vec{\sigma}_{u_i} \\ |z_{ik}| \geqslant 1, & \text{otherwise.} \end{cases} \tag{4}$$

Similarly, an entry q_{jk} of vector q_j can be characterized by

$$\begin{cases} q_{jk} = 0, & \text{if } \vec{\sigma}'_{e_j} = \vec{\sigma}_{e_j} \\ |q_{jk}| \geqslant 1, & \text{otherwise.} \end{cases} \tag{5}$$

To verify that $\sigma \neq \sigma'$, we have to prove that there is at least one entry z_{ik} of z_i (q_{jk} of q_j), such that $|z_{ik}| \geqslant 1$ ($|q_{jk}| \geqslant 1$).

Proposition 3. *Let $Q = (N, M_0, E, V, \delta)$ be a POPN, $w = \hat{M}_0e_1\hat{M}_1e_2\ldots\hat{M}_{ho-1}e_{ho}\hat{M}_{ho} \in \mathcal{O}(Q, M_0)$ be an observation sequence, and $\sigma = \sigma_{u_1}\sigma_{e_1}\sigma_{u_2}\sigma_{e_2}\ldots\sigma_{u_{ho}}\sigma_{e_{ho}}\sigma_{u_{ho+1}}$ be a transition sequence, such that $\sigma \in \mathcal{S}(w)$. A transition sequence $\sigma' = \sigma'_{u_1}\sigma'_{e_1}\sigma'_{u_2}\sigma'_{e_2}\ldots\sigma'_{u_{ho}}\sigma'_{e_{ho}}\sigma'_{u_{ho+1}} \in \mathcal{S}(w) \setminus \{\sigma\}$ if and only if $\vec{\sigma}'_{u_1}, \vec{\sigma}'_{u_2}, \ldots, \vec{\sigma}'_{u_{ho+1}}, \vec{\sigma}'_{e_1}, \vec{\sigma}'_{e_2}, \ldots, \vec{\sigma}'_{e_{ho}}$ satisfy $\mathfrak{C}(M_0, C_d, \tilde{C}_u, w)$, together with the following constraint set:*

$$
\begin{cases}
\left.\begin{array}{l}
\vec{z}_i = \vec{\sigma}'_{u_i} - \vec{\sigma}_{u_i} \\
\sum\limits_{i=1}^{ho+1} \sum\limits_{k=1}^{\tilde{n}_u} |z_{ik}| \geq 1 \cdot y_1 \\
\vec{z}_i = (z_{ik})_{k\in[1,\ldots,\tilde{n}_u]}
\end{array}\right\} & (a) \\
\left.\begin{array}{l}
\vec{q}_j = \vec{\sigma}'_{e_j} - \vec{\sigma}_{e_j} \\
\sum\limits_{j=1}^{ho} \sum\limits_{k=1}^{n_d} |q_{jk}| \geq 1 \cdot y_2 \\
\vec{q}_j = (q_{jk})_{k\in[1,\ldots,n_d]}
\end{array}\right\} & (b) \\
\left.\begin{array}{l}
M_{i-1} = M_0 + \tilde{C}_u \sum\limits_{k=1}^{i-1} \vec{\sigma}'_{u_k} + C_d \sum\limits_{k=1}^{i-1} \vec{\sigma}'_{e_k} \\
M_{i-1} + \tilde{C}_u (\sum\limits_{k=1}^{r-1} \vec{x}_{ik}) \geq P\tilde{re}_u \cdot \vec{x}_{ir} \\
\sum\limits_{k=1}^{L_i} \vec{x}_{ik} \leq \vec{\sigma}'_{u_i} \\
v_i = \sum\limits_{k=1}^{L_i} \vec{x}_{ik}(t_{ik}) \\
\sum\limits_{i=1}^{ho+1} v_i \leq \sum\limits_{i=1}^{ho+1} L_i - y_3 \\
i = 1,\ldots,ho+1, j \in \{1,\ldots,ho\}, \\
r = 1,\ldots,L_i, l = 1,\ldots,L_i - 1, \\
\vec{x}_{ik} \in \{0,1\}^{\tilde{n}_u}, v_i \in \mathbb{N}
\end{array}\right\} & (c) \\
\left.\begin{array}{l}
y_1 + y_2 + y_3 \geq 1 \\
y_1, y_2, y_3 \in \{0,1\}
\end{array}\right\} & (d),
\end{cases}
\tag{6}
$$

where:

- $\sigma_{u_i} = (t_{ik})_{k\in[1,\ldots,L_i]}$, *with* $L_i = |\sigma_{u_i}|$ *if* $|\sigma_{u_i}| \geq 1$ *and* $L_i = 1$ *otherwise, for* $i \in \{1,\ldots,ho+1\}$;
- *Const. (6.a) checks the difference between* $\vec{\sigma}'_{u_i}$ *and* $\vec{\sigma}_{u_i}$;
- *Const. (6.b) expresses the difference between* $\vec{\sigma}'_{e_j}$ *and* $\vec{\sigma}_{e_j}$;
- *Const. (6.c) indicates whether* $\vec{\sigma}_{u_i}$ *and* $\vec{\sigma}'_{u_i}$ *have the same firing order or not;*
- *Const. (6.d) uses three binary variables,* y_1, y_2, *and* y_3, *to ensure that at least one of Constraints (6.a), (6.b) or (6.c) is satisfied.*

Proof. (If) If there exists $\sigma' = \sigma'_{u_1}\sigma'_{e_1}\sigma'_{u_2}\sigma'_{e_2}\ldots\sigma'_{u_{ho}}\sigma'_{e_{ho}}\sigma'_{u_{ho+1}}$, whose firing vectors $\vec{\sigma}'_{u_1}$, $\vec{\sigma}'_{u_2}$, $\ldots$, $\vec{\sigma}'_{u_{ho+1}}$, $\vec{\sigma}'_{e_1}$, $\vec{\sigma}'_{e_2}$, $\ldots$, $\vec{\sigma}'_{e_{ho}}$ satisfy the Constraint Set $\mathfrak{C}(M_0, C_d, \tilde{C}_u, w)$, then by Proposition 2, $\sigma' \in \mathcal{S}(w)$. In Constraint Set (6.a), we compute the difference between the firing vectors of σ' and the associated firing vectors of σ, using $\vec{z}_i$ and $\vec{q}_j$. Then, we check for each $\vec{z}_i, i \in \{1,\ldots,ho+1\}(\vec{q}_j, j \in \{1,\ldots,ho\})$ if there exists at least an entry $z_{ik}(q_{jk})$, such that $|z_{ik}| \geq 1(|q_{jk}| \geq 1)$, to ensure that σ' is different from σ for at least one firing vector. In fact, σ'_{e_j} and σ_{e_j} are both composed of one transition (i.e., $|\sigma_{e_j}| = |\sigma'_{e_j}| = 1$), so it is easy to check whether $\sigma_{e_j} = \sigma'_{e_j}$, using the firing vectors only. However, in the case of $\vec{\sigma}'_{u_i} = \vec{\sigma}_{u_i}$, only the transition firing order can distinguish σ'_{u_i} from σ_{u_i}. In this situation, for $L_i \geq 1$, we split $\vec{\sigma}'_{u_i}$ into L_i sub-firing vectors $\vec{x}_{ik}$, using the enabling condition $M_{i-1} + \tilde{C}_u \sum\limits_{k=1}^{r-1} \vec{x}_{ik} \geq P\tilde{re}_u \cdot \vec{x}_{ir}$, such that $\vec{\sigma}'_{u_i} = \sum\limits_{k=1}^{L_i} \vec{x}_{ik}$. The value $v_i = \sum\limits_{k=1}^{L_i} \vec{x}_{ik}(t_{ik})$ corresponds to the number of unobservable transitions with the same firing order in σ'_{u_i} and σ_{u_i}. Therefore, if $v_i \leq L_i - y_3$ holds, where y_3 is a binary decision variable, then σ'_{u_i} is different from σ_{u_i} for at least one transition. Given the binary decision variables y_1, y_2, and y_3, Constraints (6.d) indicate that the satisfaction of either (6.a), (6.b) or (6.c) correspond to the satisfaction of the

linear constraint $y_1 + y_2 + y_3 \geq 1$. Thus, σ and σ' are different for at least one transition. Accordingly, $\sigma' \in \mathcal{S}(w) \setminus \{\sigma\}$ holds.

(Only if) Given a transition sequence $\sigma \in \mathcal{S}(w)$, assume that there exists $\sigma' \in \mathcal{S}(w) \setminus \{\sigma\}$. By Proposition 2, $\sigma' = \sigma'_{u_1}\sigma'_{e_1}\sigma'_{u_2}\sigma'_{e_2}\ldots\sigma'_{u_{ho}}\sigma'_{e_{ho}}\sigma'_{u_{ho+1}}$, such that its firing vectors satisfy $\mathfrak{C}(M_0, C_d, \tilde{C}_u, w)$. In addition, we have $\sigma \neq \sigma'$, which possibly implies the existence of a particular step i, such that $\vec{\sigma}'_{u_i} \neq \vec{\sigma}_{u_i}$ or a step j, such that $\vec{\sigma}'_{e_j} \neq \vec{\sigma}_{e_j}$. Thus, Constraints (6.a) and (6.b) hold. If not, there exist two sub-sequences σ'_{u_i} and σ_{u_i} with identical firing vectors but different transition firing orders, and then Constraint (6.c) holds. Finally, the satisfaction of either (6.a), (6.b) or (6.c) implies the satisfaction of Constraint (6.d). □

Evidently, with the use an absolute value function in (6.a) and (6.b), the problem becomes non-linear. In this case, it is difficult to apply standard ILP solvers to solve it. However, it is possible to linearize the absolute value of an integer variable (i.e., $|z_{ik}|$), by replacing z_{ik} and $|z_{ik}|$, using two integer variables z^+_{ik} and z^-_{ik} as follows:

$$\begin{cases} z_{ik} = z^+_{ik} - z^-_{ik} & \text{(a)} \\ |z_{ik}| = z^+_{ik} + z^-_{ik} & \text{(b)} \\ z^+_{ik} \leq U \cdot a_{ik} & \text{(c)} \\ z^-_{ik} \leq U \cdot (1 - a_{ik}) & \text{(d).} \\ a_{ik} \in \{0,1\}, z^+_{ik}, z^-_{ik} \in \mathbb{N} & \end{cases} \tag{7}$$

U is an integer satisfying $U \geq max\{B(p)|p \in P\}$, where $B(p)$ denotes place p upper bound [19]. Constraints (7.c) and (7.d) guarantee that either z^+_{ik} or z^-_{ik} (or both, if $z_{ik} = 0$) will be 0. Thus, $z_{ik} = z^+_{ik}$ when $z_{ik} \geq 0$, and $z_{ik} = z^-_{ik}$ when $z_{ik} \leq 0$. We can exploit the notation in (7) to remove $|z_{ik}|$ in (6.a) and $|q_{jk}|$ in (6.b), to relax (6) into a set of linear constraints $\mathfrak{MS}(M_0, C_d, \tilde{C}_u, \sigma) =$

$$
\left\{\begin{array}{l}
\left.\begin{array}{ll}
\vec{z}_i^{+} - \vec{z}_i^{-} = \vec{\sigma}'_{u_i} - \vec{\sigma}_{u_i} & (a_1)\\
\sum\limits_{i=1}^{ho+1}\sum\limits_{k=1}^{\tilde{n}_u} z_{ik}^{+} + z_{ik}^{-} \geq 1 \cdot y_1 & (a_2)\\
\left.\begin{array}{l} z_{ik}^{+} \leq U \cdot a_{ik} \\ z_{ik}^{-} \leq U \cdot (1 - a_{ik}) \end{array}\right\} & (a_3)\\
\vec{z}_i^{+} = (z_{ik}^{+})_{k\in[1,\ldots,\tilde{n}_u]},\\
\vec{z}_i^{-} = (z_{ik}^{-})_{k\in[1,\ldots,\tilde{n}_u]},
\end{array}\right\} \quad \text{(a)}\\
\left.\begin{array}{ll}
\vec{q}_j^{+} - \vec{q}_j^{-} = \vec{\sigma}'_{e_j} - \vec{\sigma}_{e_j} & (b_1)\\
\sum\limits_{i=1}^{ho}\sum\limits_{k=1}^{n_d} q_{jk}^{+} + q_{jk}^{-} \geq 1 \cdot y_2 & (b_2)\\
\left.\begin{array}{l} q_{jk}^{+} \leq U \cdot b_{jk} \\ q_{jk}^{-} \leq U \cdot (1 - b_{jk}) \end{array}\right\} & (b_3)\\
\vec{q}_j^{+} = (q_{jk}^{+})_{k\in[1,\ldots,n_d]},\\
\vec{q}_j^{+} = (q_{jk}^{-})_{k\in[1,\ldots,n_d]}
\end{array}\right\} \quad \text{(b)}\\
\left.\begin{array}{ll}
M_{i-1} = M_0 + \tilde{C}_u \sum\limits_{k=1}^{i-1} \vec{\sigma}'_{u_k} + C_d \sum\limits_{k=1}^{i-1} \vec{\sigma}'_{e_k} & (c_1)\\
M_{i-1} + \tilde{C}_u (\sum\limits_{k=1}^{r-1} \vec{x}_{ik}) \geq P\tilde{r}e_u \cdot \vec{x}_{ir} & (c_2)\\
\sum\limits_{k=1}^{L_i} \vec{x}_{ik} \leq \vec{\sigma}'_{u_i} & (c_3)\\
v_i = \sum\limits_{k=1}^{L_i} \vec{x}_{ik}(t_{ik}) & (c_4)\\
\sum\limits_{i=1}^{ho+1} v_i \leq \sum\limits_{i=1}^{ho+1} L_i - y_3 & (c_5)\\
i = \{1,\ldots,ho+1\}, j \in \{1,\ldots,ho\}\\
r = \{1,\ldots,L_i\}, l = \{1\ldots,L_i - 1\},\\
\vec{x}_{ik} \in \{0,1\}^{\tilde{n}_u}, v_i \in \mathbb{N}
\end{array}\right\} \quad \text{(c)}\\
\left.\begin{array}{l}
y_1 + y_2 + y_3 \geq 1\\
y_1, y_2, y_3 \in \{0,1\}
\end{array}\right\} \quad \text{(d).}
\end{array}\right. \tag{8}
$$

Theorem 1. *Let $Q = (N, M_0, E, V, \delta)$ be a POPN and $L_s \subseteq L(N, M_0)$ be a secret language. Q is language-opaque, with regard to L_s, if, for any $w \in \mathcal{O}(L_s)$, the following system of linear equations and inequalities (combining (3) and (6)), denoted by $\mathcal{G}(w)$,*

$$
\mathcal{G}(w) = \left\{\begin{array}{l}
\mathfrak{C}(M_0, C_d, \tilde{C}_u, w),\\
\mathfrak{NS}(M_0, C_d, \tilde{C}_u, \sigma),\ \sigma \in L_s
\end{array}\right. \tag{9}
$$

admits at least one feasible solution.

Proof. Let $w \in \mathcal{O}(Q, M_0)$ be an observation sequence. Assume that there exists σ', such that $\sigma' = \sigma'_{u_1}\sigma'_{e_1}\sigma'_{u_2}\sigma'_{e_2}\ldots\sigma'_{u_{ho}}\sigma'_{e_{ho}}\sigma'_{u_{ho+1}}$ satisfying the Constraints Set $\mathcal{G}(w)$. We have the firing vectors of σ' satisfy the constraint set $\mathfrak{C}(M_0, C_d, \tilde{C}_u, w)$. According to Proposition 2, $\sigma' \in \mathcal{S}(w)$. We also have the firing vectors of σ' satisfy $\mathfrak{NS}(M_0, C_d, \tilde{C}_u, \sigma) for\ \sigma \in L_s$. Based on Proposition 3, $\sigma' \in \mathcal{S}(w) \setminus \{\sigma\}$ for each $\sigma \in L_s$: that is to say, $\sigma' \in \mathcal{S}(w) \setminus L_s$. According to Defintion 7, w is LBO, with regard to L_s. □

Note that the Constraint set $\mathcal{G}(w)$ is usually referred to as a "feasibility problem". One way to solve a feasibility problem is to convert it into an optimization problem with a dummy objective function, and then solve it using ILP solvers. This objective function could be a linear combination of the subset of decision variables. When we aim to maximize it, we will discover a feasible solution, provided one exists. Conversely, if we aim to minimize it, we will attain a different feasible solution, typically located on the opposite side of the feasible region. One option for the objective function would be

$$min\ \vec{\mathbf{1}}_{\tilde{n}_u}^T \cdot \sum_{i=1}^{ho+1} \vec{\sigma}'_{u_i}.$$

5.3. Complexity Analysis and Comparison

Theoretically, an ILP problem is NP-hard, and the computational overhead of its resolution depends on its variables and constraints numbers. In Table 1, we analyze the number of variables and constraints of ILPP (9). Columns 1, 2, and 3 denote the lists of variables, their types, and their sizes, respectively. Column 4 indicates the range of subscripts associated with each variable, and Column 5 denotes the total number of variables in scalar form. From the presented results, it is obvious that the numbers of variables in the worst case is

$$\tilde{n}_u \cdot (ho+1) \cdot (1+3 \cdot \eta + L_i \cdot \eta) + n_d \cdot ho \cdot (1+3 \cdot \eta) + (ho+1) \cdot \eta \cdot (m+1) + 4 \cdot \eta,$$

where $\eta = |\mathcal{L}_s(w)|$.

Table 1. Integer variables of ILPP $\mathcal{G}(w)$.

Variable	Type	Size	Range	Total
$\vec{\sigma}_{u_i}$	Integer	$\tilde{n}_u \times 1$	$i \in \{1, \ldots, ho+1\}$	$\tilde{n}_u \cdot (ho+1)$
$\vec{\sigma}_{e_j}$	Integer	$n_d \times 1$	$i \in \{1, \ldots, ho+1\}$	$n_d \cdot ho$
z_{ik}^+	Integer	1×1	$i \in \{1, \ldots, ho+1\} k \in \{1, \ldots, \tilde{n}_u\}$	$\tilde{n}_u \cdot (ho+1) \cdot \eta$
z_{ik}^-	Integer	1×1	$i \in \{1, \ldots, ho+1\} k \in \{1, \ldots, \tilde{n}_u\}$	$\tilde{n}_u \cdot (ho+1) \cdot \eta$
q_{jk}^+	Integer	1×1	$i \in \{1, \ldots, ho+1\} k \in \{1, \ldots, n_d\}$	$n_d \cdot ho \cdot \eta$
q_{jk}^-	Integer	1×1	$i \in \{1, \ldots, ho+1\} k \in \{1, \ldots, n_d\}$	$n_d \cdot ho \cdot \eta$
a_{ik}	Binary	1×1	$i \in \{1, \ldots, ho+1\} k \in \{1, \ldots, \tilde{n}_u\}$	$\tilde{n}_u \cdot (ho+1) \cdot \eta$
b_{jk}	Binary	$j \times 1$	$i \in \{1, \ldots, ho+1\} k \in \{1, \ldots, n_d\}$	$n_d \cdot ho \cdot \eta$
U	Integer	1×1	1	$1 \cdot \eta$
M_{i-1}	Integer	$m \times 1$	$i \in \{1, \ldots, ho+1\}$	$m \cdot (ho+1) \cdot \eta$
$\vec{x}_{ik}$	Binary	$\tilde{n}_u \times 1$	$i \in \{1, \ldots, ho+1\} k \in \{1, \ldots, L_i\}$	$\tilde{n}_u \cdot (ho+1) \cdot L_i \cdot \eta$
v_i	Integer	1×1	$i \in \{1, \ldots, ho+1\}$	$(ho+1) \cdot \eta$
y_1	Binary	1×1	1	$1 \cdot \eta$
y_2	Binary	1×1	1	$1 \cdot \eta$
y_3	Binary	1×1	1	$1 \cdot \eta$

In Table 2, the first column denotes the constraint sets involved in ILPP (9), the second column represents the sub-constraints of each constraint set, the third column shows the extent in lines of each sub-constraint, and the fourth column denotes the range of their associated indexes. Finally, Column 5 denotes the total number of constraints contained in each sub-constraint set. According to the results reported in Table 2, we can deduce the number of constraints in ILPP (9), which is given by $(ho+1) \cdot (4 \cdot \tilde{n}_u \cdot \eta + m_o + m \cdot L_i \cdot \eta + 2 \cdot \eta + m) + ho \cdot (m_o + 3 \cdot n_d \cdot \eta + 2) + 4 \cdot \eta + 1$. As a result, the variables and constraints numbers are polynomial in the secret observation sequence length.

Table 2. Constraints of ILPP $\mathcal{G}(w)$.

Constraint Set	Sub-Constraints	Extent	Range	Total
f	1	1	1	1
$\mathfrak{C}(M_0, C_d, \tilde{C}_u, w)$	3.a	m	1	m
	3.b	m	$l \in \{1 \dots ho\}$	$m \cdot ho$
	3.c	m_o	$i \in \{1 \dots ho+1\}$ $j \in \{1 \dots ho\}$	$m_o \cdot (ho+1)$ $m_o \cdot ho$
	3.d	2	$j \in \{1 \dots ho\}$	$2 \cdot ho$
$\mathfrak{NS}(M_0, C_d, \tilde{C}_u, \sigma)\ \sigma \in \mathcal{L}_s(w)$	8.a1	$\tilde{n}_u$	$i \in \{1 \dots ho+1\}$	$\tilde{n}_u \cdot (ho+1) \cdot \eta$
	8.b1	n_d	$j \in \{1 \dots ho\}$	$n_d \cdot ho \cdot \eta$
	8.a2	1	1	$1 \cdot \eta$
	8.b2	1	1	$1 \cdot \eta$
	8.a3	2	$i \in \{1 \dots ho+1\}$ $k \in \{1 \dots \tilde{n}_u\}$	$2 \cdot \tilde{n}_u \cdot (ho+1) \cdot \eta$
	8.b3	2	$j \in \{1 \dots ho\}$ $k \in \{1 \dots n_d\}$	$2 \cdot n_d \cdot ho \cdot \eta$
	8.c1	m	$i \in \{1 \dots ho+1\}$	$\tilde{n}_u \cdot (ho+1) \cdot \eta$
	8.c2	m	$i \in \{1 \dots ho+1\}$ $r \in \{1 \dots L_i\}$	$m \cdot L_i(ho+1) \cdot \eta$
	8.c3	1	$i \in \{1 \dots ho+1\}$	$(ho+1) \cdot \eta$
	8.c4	1	$i \in \{1 \dots ho+1\}$	$(ho+1) \cdot \eta$
	8.c5	1	1	$1 \cdot \eta$
	8.d	1	1	$1 \cdot \eta$

In Table 3, the proposed approach is compared to previous works on LBO verification presented in [5,10,13]. The second column presents the application framework. The third column indicates whether the verification approach is based on an off-line graph computation or not. The fourth column denotes the secret nature. Finally, the fifth column reports the complexity of each approach.

Table 3. LBO verification approaches.

	Framework	Graph	Secret Nature	Complexity
[5]	Automaton	No	Labels	Exponential
[10]	LPN	Yes	Observable transitions	Exponential
[13]	LPN	No	Labels	NP-hard
Proposed approach	POPN	No	Transitions	NP-hard

6. Experimental Results: A Manufacturing System

In this section, we exhibit the above results, using the example of an automated manufacturing system from [20], as shown in Figure 2. It contains 46 places and 39 transitions, including two inputs (I1 and I2), two outputs (O1 and O2), four machines (M1–M4), one buffer with finite capacity (B), and four robots (R1–R4). In [20], the operation of this system is discussed in detail. Assume that I1 and I2 each include a sensor that displays the number of items handled by the corresponding lines L1 and L2. Furthermore, sensors are installed on the machines (M1–M4), the two robots (R1 and R2), and the buffer (B), to indicate when an item is added. This PN might be considered as a POPN.

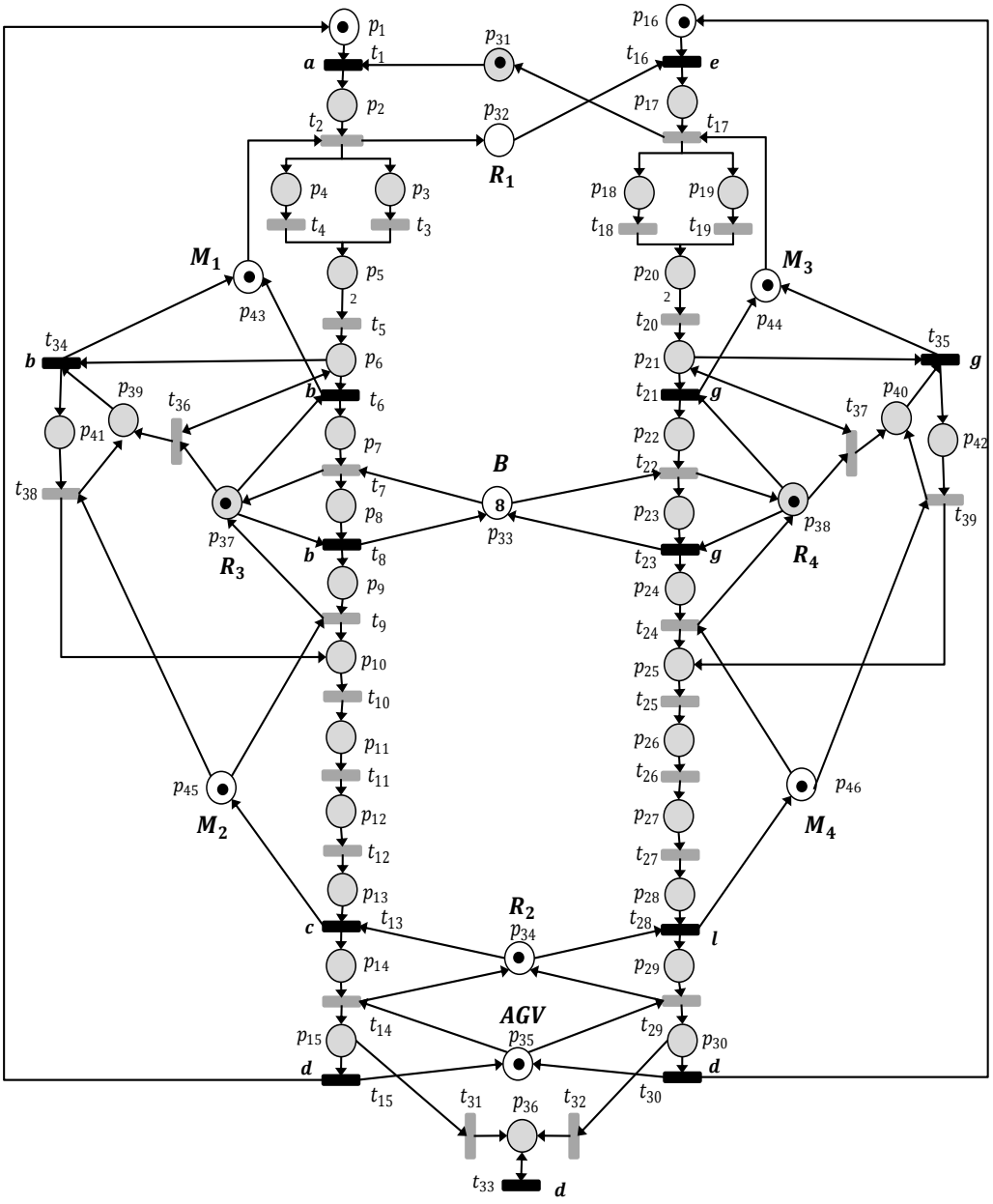

Figure 2. An automated manufacturing system's Petri net model.

We have:
$T_d = \{t_1, t_2, t_6, t_7, t_8, t_9, t_{13}, t_{14}, t_{15}, t_{16}, t_{17}, t_{21}, t_{22}, t_{23}, t_{24}, t_{28}, t_{29}, t_{30}, t_{34}, t_{35}, t_{38}, t_{39}\}$.
$\tilde{T}_u = \{t_3, t_4, t_5, t_{10}, t_{11}, t_{12}, t_{18}, t_{19}, t_{20}, t_{25}, t_{26}, t_{27}, t_{31}, t_{32}, t_{33}, t_{36}, t_{37}\}$.

Let $L_s = \{\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5, \sigma_6, \sigma_7, \sigma_8, \sigma_9, \sigma_{10}, \sigma_{11}, \sigma_{12}\}$ be a secret language and $\mathcal{O}(L_s) = \{w_1, w_2, w_3\}$ be its associated secret observation sequences, as specified in Tables 4 and 5, respectively. The results reported in this paper were obtained by solving (9) using CPLEX, which is a commercial Python library for linear programming optimization.

For $\mathcal{G}(w_1)$, there exists a sequence $\sigma'_1 = \sigma'_{u_1}\sigma'_{e_1}\sigma'_{u_2}\sigma'_{e_2}\sigma'_{u_3}\sigma'_{e_3}\sigma'_{u_4} = t_1t_2t_3t_4t_5t_6$ enabled at M_0, such that $\sigma'_1 \notin L_s$ and $\Lambda(\rho(M_0, \sigma'_1)) = w_1$, whose firing vectors $\vec{\sigma}'_{u_1} = \vec{\mathbf{0}}$, $\vec{\sigma}'_{e_1} = b_1, \vec{\sigma}'_{u_2} = \vec{\mathbf{0}}, \vec{\sigma}'_{e_2} = b_2, \vec{\sigma}'_{u_3} = a_1 + a_2 + a_3, \vec{\sigma}'_{e_3} = b_3, \vec{\sigma}'_{u_4} = \vec{\mathbf{0}}$ satisfy (9), where a_i and b_i are called standard basis vectors for $\mathbb{N}^{\tilde{n}_u}$ and $\mathbb{N}^{n_d}$, respectively, having all entries zero, except that the i^{th} entry equals 1. Notice that the solver takes into account the firing order of transitions, since σ' has the same firing vector as the secret sequence $t_1t_2t_4t_3t_5t_6$.

Table 4. Secret transition sequences.

σ_i	Value	$\mathcal{L}_s(\sigma)$
σ_1	$t_1t_2t_4t_3t_5t_6$	w_1
σ_2	$t_1t_2t_{16}$	w_2
σ_3	$t_1t_2t_{16}t_3, t_1t_2t_3t_4t_5t_{36}t_{16}$	w_2
σ_4	$t_1t_2t_3t_4t_5t_{16}, t_1t_2t_3t_4t_{16}t_5$	w_2
σ_5	$t_1t_2t_3t_4t_5t_{36}t_{34}$	w_1
σ_6	$t_1t_2t_3t_4t_5t_{16}t_{36}$	w_2
σ_7	$t_1t_2t_3t_4t_{16}$	w_2
σ_8	$t_1t_2t_3t_4t_{16}t_5t_{36}$	w_2
σ_9	$t_1t_2t_{16}t_3t_4t_5$	w_2
σ_{10}	$t_1t_2t_{16}t_3t_4$	w_2
σ_{11}	$t_1t_2t_{16}t_3t_4t_5t_{36}$	w_2
σ_{12}	$t_1t_2t_3t_4t_5t_6t_7t_8t_9t_{10}t_{11}t_{12}t_{13}$	w_3

Table 5. Secret observation sequences.

w_i	Value
w_1	$[1108111111]^T a[0108111111]^T[0118110111]^T b[0118111111]^T$
w_2	$[1108111111]^T a[0108111111]^T[0118110111]^T e[0008110111]^T$
w_3	$[1108111111]^T a[0108111111]^T[0118110111]^T b[0118111111]^T$ $[0117111111]^T b[0118111111]^T[0118111101]^T c[0118011111]^T$

For $\mathcal{G}(w_2)$, there exists a sequence $\sigma'_2 = \sigma'_{u_1}\sigma'_{e_1}\sigma'_{u_2}\sigma'_{e_2}\sigma'_{u_3}\sigma'_{e_3}\sigma'_{u_4} = t_1t_2t_4t_{16}$ enabled at M_0, such that $\sigma'_2 \notin L_s$ and $\Lambda(\rho(M_0,\sigma'_2)) = w_2$, whose firing vectors $\vec{\sigma}'_{u_1} = \vec{\mathbf{0}}$, $\vec{\sigma}'_{e_1} = b_1$, $\vec{\sigma}'_{u_2} = \vec{\mathbf{0}}$, $\vec{\sigma}'_{e_2} = b_2$, $\vec{\sigma}'_{u_3} = a_2$, $\vec{\sigma}'_{e_3} = b_{10}$, and $\vec{\sigma}'_{u_4} = \vec{\mathbf{0}}$ satisfy (9).

For $\mathcal{G}(w_3)$, however, the ILP solver does not find any feasible solution, which implies the non-language opacity of the considered POPN.

7. LBO in Decentralized Setting

Modern networking technologies have rendered distributed systems increasingly more broadly used. These systems are composed of multiple networked components that communicate and coordinate actions, to achieve a common goal and appear to end-users as a single coherent entity. Figure 3 illustrates the privacy risks in an IoT-based smart house, where a set of intruders, distributed at different sites, are collecting sensing data and trying to deduce private information.

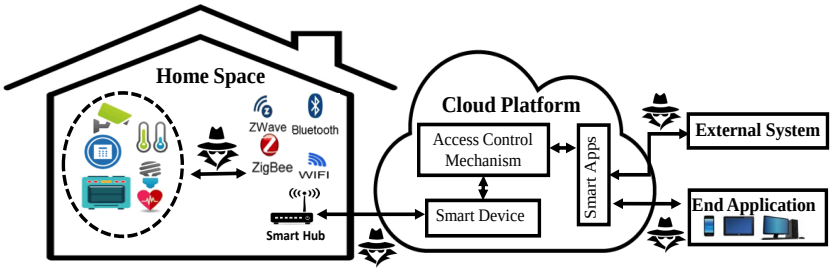

Figure 3. Privacy risks of Iot-based smart houses.

7.1. Decentralized Opacity with Coordinator

At this point, language-based opacity has been discussed in a centralized framework, where only a single intruder is trying to infer the system's secret behavior. In this section, we consider decentralized settings, as shown in Figure 4, where a POPN is watched by a set of intruders $\mathcal{J} = \{1, 2, \ldots, n\}$ who collaborate to verify, under a given observation sequence, if their language estimation is entirely contained in a secret. Formally, the system Q, with regard to the j^{th} intruder, can be described as follows:

$Q_j = (N, M_0, E_j, V_j, \delta_j)$ is a POPN system, where $N = (P, T, Pre, Post)$ and $E_j \subseteq E$ is the set of observable labels by the j^{th} local intruder. As in the previous section, it holds that $P = P_{o,j} \cup P_{u,j}$ and $P_{o,j} \subseteq P_o (P_{u,j} = P \setminus P_{o,j})$ comprise the set of locally observable (unobservable) places of intruder j. In addition, $T = T_{o,j} \cup T_{u,j}$ and $T_{o,j} \subseteq T_o (T_{u,j} = T \setminus T_{o,j})$ comprise the set of locally observable (silent) transitions of intruder j. Moreover, we denote by $T_{d,j} \subseteq T_d$ ($\tilde{T}_{u,j} = T \setminus T_{d,j}$) the set of locally discernible (truly unobservable) transitions of intruder j. It holds that $|P_{o,j}| = m_{o,j}$, $|P_{u,j}| = m_{u,j}$, $|T_{d,j}| = n_{d,j}$ and $|\tilde{T}_{u,j}| = \tilde{n}_{u,j}$. The matrices $\tilde{C}_{u,j} = \tilde{Post}_{u,j} - \tilde{Pre}_{u,j}$ and $C_{d,j} = Post_{d,j} - Pre_{d,j}$, respectively, denote the restrictions of C to $\tilde{T}_{u,j}$ and $T_{d,j}$.

The following assumptions are now introduced to this decentralized setting:

A1: Each intruder has full understanding of the system's structure and initial marking, but uses his own sensor configuration $\theta_j = (V_j, \delta_j)$ to track its evolution, where $V_j = (v_{ik}) \in \{0,1\}^{m_{o,j} \times m}$, such that $v_{ik} = 1$ if the i^{th} observable place by intruder j corresponds to place p_k and to 0, otherwise, and where δ_j is a local labeling function, such that $\delta_j(t) = \delta(t)$ if $t \in T_{d,j}$ and $\delta_j(t) = \varepsilon$, otherwise.

A2: An observable place is observed by at least one local intruder, i.e., $\bigcup_{j \in J} P_{o,j} = P_o$.

A3: A discernible transition is discerned by at least one local intruder, i.e., $\bigcup_{j \in J} T_{d,j} = T_d$.

A4: Let $\preceq$ be an ordering relation defined between sensor configurations, such that $\theta_i \preceq \theta_j$ if intruder j observes more than intruder i, i.e., $\theta_i \preceq \theta_j$ if $T_{d,i} \subseteq T_{d,j} \wedge P_{o,i} \subseteq P_{o,j}$.

Given $M[t\rangle M'$, a state transition, the obtained measurement with respect to a sensor configuration θ_j associated with the jth local intruder is defined as follows:

$$\rho_j(M,t) = \begin{cases} \varepsilon_p, & \text{if } (\hat{M} = \hat{M}') \wedge (\delta_j(t) = \varepsilon) \\ \hat{M}\delta_j(t)\hat{M}', & \text{otherwise.} \end{cases}$$

Given $\sigma = t_1 t_2 \dots t_h \in T^*$ and $Mt_1M_1t_2M_2 \dots M_{h-1}t_hM_h$, the evolution generated by firing σ at M, the measurement sequence associated with each local intruder j is defined as

$$\begin{aligned} \rho_j(M,\sigma) &= \rho_j(M,t_1)\rho_j(M_1,t_2)\dots\rho_j(M_{h-1},t_h) \\ &= \hat{M}e_1\hat{M}_1\hat{M}_1e_2\dots\hat{M}_{ho_j-1}\hat{M}_{ho_j-1}e_{ho_j}\hat{M}_{ho_j}, \end{aligned}$$

where $e_i \in E_j \cup \{\varepsilon\}$, $ho_j \leq h$, and the associated local observation sequence for each intruder j is defined as $w_j = \Lambda(\rho_j(M,\sigma)) = \hat{M}e_1\hat{M}_1e_2\dots\hat{M}_{ho_j-1}e_{ho_j}\hat{M}_{ho_j}$.

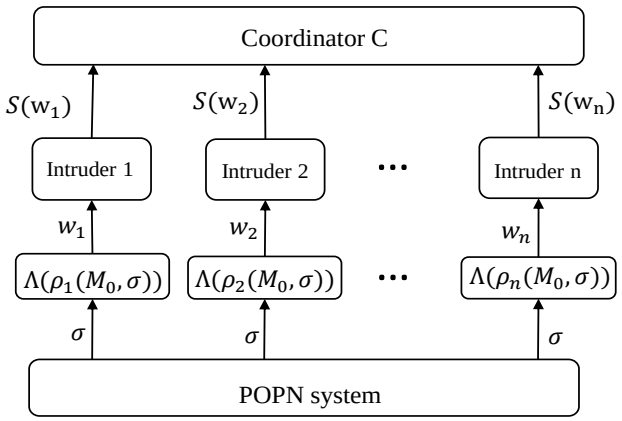

Figure 4. The decentralized architecture.

The firing sequences consistent with a local observation w_j are denoted by $\mathcal{S}_j(w_j) = \{\sigma \in T^* | \Lambda(\rho_j(M_0,\sigma)) = w_j\}$. As illustrated in Figure 4, after the firing of a transition sequence σ, each intruder j performs an event sequence estimation $\mathcal{S}_j(w_j)$, using its local observation sequence w_j, then sends it to a coordinator C. Based on the received information

from all intruders, the coordinator evaluates its general estimation $\Gamma(w)$ by finding the intersection of the received local estimations. Therefore, the coordination between intruders is called an intersection-based coordination protocol $\mathcal{P}$.

Definition 8. *An intersection-based coordination protocol $\mathcal{P}$ is defined by the following rules:*

R1: The $T_{u,j}$-induced subnet and the $T_{o,j}$-induced subnet are acyclic for any local intruder $j \in \mathcal{J}$.
R2: The general estimation is $\Gamma(w) = \bigcap_{j\in\mathcal{J}} \mathcal{S}_j(w_j)$.
R3: The more an intruder observes, the better its estimation will be, i.e., $\theta_i \preceq \theta_j$ if, for all $w \in \mathcal{O}(Q, M_0)$, $\mathcal{S}_j(w_j) \subseteq \mathcal{S}_i(w_i)$.
R4: The coordinator collects the event sequence estimation generated by each intruder without delay.

Remark 1. *Let Q be a POPN, and $\mathcal{P}$ be an intersection-based coordination protocol; $T_{d,c} \subseteq T_d$ and $P_{o,c} \subseteq P_o$ are, respectively, the discernible transitions and observable places of Q, with regard to a coordinator C. Protocol $\mathcal{P}$ allows the coordinator to see the system's original behavior: namely, $T_{d,c} = T_d$ and $P_{o,c} = P_o$.*

In detail, we have $\Gamma(w) = \bigcap_{j\in\mathcal{J}} \mathcal{S}_j(w_j)$; then, for all $w \in \mathcal{O}(Q, M_0)$, $\Gamma(w) \subseteq \mathcal{S}_j(w_j)$, $j \in \mathcal{J}$. We denote by θ_c the sensor configuration of the coordinator. Under Rule R3, $\theta_j \preceq \theta_c$ for $j \in \mathcal{J}$; thus, we can say that the coordinator observes more than all intruders $j \in \mathcal{J}$. According to Assumption (A4), for all $j \in \mathcal{J}$, we have $T_{d,j} \subseteq T_{d,c}$ and $P_{o,j} \subseteq P_{o,c}$. As $\bigcup_{j\in J} T_{d,j} = T_d$ and $\bigcup_{j\in J} P_{o,j} = P_o$, we have $T_d \subseteq T_{d,c}$ and $P_o \subseteq P_{o,c}$. Consequently, $T_{d,c} = T_d$ and $P_{o,c} = P_o$. Finally, we conclude that our coordination protocol allows the coordinator to see the system's original behavior, by taking off the observation masks associated with intruders.

Now, let us characterize LBO in a decentralized setting with coordination: namely, co-language-based opacity (co-LBO).

Definition 9. *Let $Q = (N, M_0, E, V, \delta)$ be a POPN and $L_s \subseteq L(N, M_0)$ be a secret language. Q is co-LBO, with regard to L_s, if and only if, for all $w \in \mathcal{O}(L_s)$, $\Gamma(w) \nsubseteq L_s$ holds.*

In simple terms, a PN system is co-LBO if it is language-opaque, with regard to the coordinator. Specifically, if two observations are indistinguishable for the coordinator, they are indistinguishable for any intruder $j \in \mathcal{J}$.

Proposition 4. *Let $Q = (N, M_0, E, V, \delta)$ be a POPN and $L_s \subseteq L(N, M_0)$ be a secret language. If there is at least $w \in \mathcal{O}(L_s)$ and $j \in \mathcal{J}$, such that $\mathcal{S}_j(w_j) \subseteq L_s$, then Q is not co-LBO, with regard to L_s.*

Proof. Let $w \in \mathcal{O}(L_s)$ be a secret observation sequence. By Rule R2, we have $\Gamma(w) = \bigcap_{j\in\mathcal{J}} \mathcal{S}_j(w_j)$, which implies $\Gamma(w) \subseteq \mathcal{S}_j(w_j)$, $j \in \mathcal{J}$. If there exists $w \in \mathcal{O}(L_s)$ and $j \in \mathcal{J}$, such that $\mathcal{S}_j(w_j) \subseteq L_s$, then $\Gamma(w) \subseteq L_s$ holds. Therefore, by Definition 9, Q is not co-LBO, with regard to L_s. □

The main feature of investigating decentralized opacity consists in preventing a comprehensive computation of all possible sequences that could have fired. According to Proposition 4, if we find at least one intruder j, such that $\mathcal{S}_j(w_j) \subseteq L_s$ holds, then Q is not co-LBO, with regard to L_s.

If there is no coordination between the intruders, we simply check the LBO for each intruder separately, as indicated in the previous sections. Accordingly, the achieved results can be extended from centralized to decentralized opacity. Therefore, we apply the results of Theorem 1 to each intruder $j \in J$ as follows:

Let $Q = (N, M_0, V, \delta)$ be a POPN and $L_s \subseteq L(N, M_0)$ be a secret language. Q is language-opaque, with regard to L_s, if, for all $w \in \mathcal{O}(L_s)$, the following ILPP,

$$\begin{cases} min & f \\ s.t. & \mathfrak{C}(M_0, C_{d,j}, \tilde{C}_{u,j}, w_j),\ j \in J \\ & \mathfrak{NS}(M_0, C_{d,j}, \tilde{C}_{u,j}, \sigma),\ \sigma \in L_s \end{cases} \tag{10}$$

admits at least one feasible solution.

Definition 10. *Let L_s be a secret language, $w = \hat{M}e_1\hat{M}_1e_2 \ldots \hat{M}_{ho-1}e_{ho}\hat{M}_{ho}(ho \geq 1)$ be a secret observation sequence, i.e., $w \in \mathcal{O}(L_s)$, and $w_j = \hat{M}'e'_1\hat{M}'_1e'_2 \ldots \hat{M}'_{ho_j-1}\hat{M}'_{ho_j-1}e'_{ho_j}\hat{M}'_{ho_j}$ $(ho_j \leq ho)$ be the word projection of w, with regard to the local intruder $j \in J$. We define by $J_{max}(w) = \{j \in J | h_j \geq h_{j'}, j' \in J\}$ the set of local intruders that can observe the maximum of w, called the maximal observers.*

Proposition 5. *Let $Q = (N, M_0, E, V, \delta)$ be a POPN and $L_s \subseteq L(N, M_0)$ be a secret language. Q is co-LBO, with regard to L_s, if and only if for each secret observation sequence $w \in \mathcal{O}(L_s)$ and, for at least $j \in J_{max}(w)$, there exists $\sigma \in \mathcal{S}(w_j) \setminus \mathcal{L}_s(w)$, such that for each local intruder $j' \in \mathcal{J}$, $\Lambda(\rho_{j'}(M_0, \sigma)) = w_{j'}$.*

Proof. (If) Suppose that, for all $w \in \mathcal{O}(L_s)$ and for at least $j \in J_{max}(w)$, there exists $\sigma \in \mathcal{S}(w_j) \setminus \mathcal{L}_s(w)$, such that for each local intruder $j' \in \mathcal{J}$, $\Lambda(\rho_{j'}(M_0, \sigma)) = w_{j'}$. Consequently, σ is a non-secret transition sequence that is consistent with $w_{j'}$ for $j' \in \mathcal{J}$ (i.e., $\sigma \in \mathcal{S}(w_{j'}), j' \in \mathcal{J}$).

We have $\Gamma(w) = \bigcap_{j \in \mathcal{J}} \mathcal{S}_j(w_j)$, and thus, $\sigma \in \Gamma(w) \setminus \mathcal{L}_s(w)$. Based on Proposition 4, Q is co-LBO, with regard to L_s.

(Only if) Let us suppose that Q is co-LBO, with regard to L_s, and for at least $w \in \mathcal{O}(L_s)$, and for each $j \in J_{max}(w)$, there does not exist $\sigma \in \mathcal{S}(w_j) \setminus L_s$, such that for each $j' \in \mathcal{J}$, $\Lambda(\rho_{j'}(M_0, \sigma)) = w_{j'}$; thus, $\bigcap_{j \in \mathcal{J}} \mathcal{S}_j(w_j) \subseteq L_s$. We have $\Gamma(w) = \bigcap_{j \in \mathcal{J}} \mathcal{S}_j(w_j) \subseteq L_s$ and, therefore, Q is not co-LBO, with regard to L_s, which opposes the hypothesis. □

Example 4. *Consider the POPN in Figure 1. Let $L_s = \{t_1t_3t_4, t_1t_3t_2t_5, t_1t_2t_3t_5\}$ be a secret language. Its associated secret observation sequences are given by $\mathcal{O}(L_s) = \{w_1, w_2\}$, where $w_1 = [1\ 0]^T a [0\ 1]^T b [0\ 1]$, $w_2 = [1\ 0]^T a [0\ 1]^T [1\ 0]^T$, $\mathcal{L}_s(w_1) = \{t_1t_3t_4\}$, and $\mathcal{L}_s(w_2) = \{t_1t_3t_2t_5, t_1t_2t_3t_5\}$. We have $\mathcal{S}(w_1) = \{t_1t_3t_4, t_1t_4t_3\}$ and $\mathcal{S}(w_2) = \{t_1t_2t_3t_5, t_1t_3t_2t_5\}$. Let $J = \{1, 2, 3\}$ be three local intruders satisfying Assumptions (A1–A4). Their observable places are $P_{o,1} = \{p_1\}$, $P_{o,2} = \{p_1, p_5\}$, and $P_{o,3} = \emptyset$, respectively, and their discernible transitions are $T_{d,1} = \{t_1, t_5\}$, $T_{d,2} = \{t_1\}$, and $T_{d,3} = \{t_1, t_5\}$, respectively.*

Based on the results reported in Table 6, we have:

- *$\mathcal{J}_{max}(w_1) = \{3\}$; then, we only need to consider the estimation of intruder 3, which is given by $\mathcal{S}_3(ab) = \{t_1t_3t_4, t_1t_4t_3, t_1t_3t_4t_5, t_1t_4t_3t_5\}$. We have $t_1t_4t_3 \notin L_s$, $\Lambda(\rho_1(M_0, t_1t_4t_3)) = [1]a[0]$ and $\Lambda(\rho_2(M_0, t_1t_4t_3)) = [1\ 0]^T[0\ 1]^T$; then, w_1 is co-LBO, with regard to L_s.*
- *$\mathcal{J}_{max}(w_2) = \{1, 2\}$. Since intruders 1 and 2 observe two discernible transitions, while intruder 3 only observes one discernible transition, then we only need to consider the estimation of either intruder 1 or 2. Let us consider the estimation of intruder 2, which is given by $\mathcal{S}_2([1\ 0]^T[0\ 1]^T[1\ 0]^T) = \{t_1t_2t_3t_5, t_1t_4t_3t_5, t_1t_3t_2t_5, t_1t_3t_4t_5\}$. We have $\mathcal{S}_2(w_2) \setminus \mathcal{L}_s(w_2) = \{t_1t_4t_3t_5, t_1t_3t_4t_5\}$. For intruder 3, $\Lambda(\rho_3(M_0, t_1t_3t_4t_5)) = \Lambda(\rho_3(M_0, t_1t_4t_3t_5)) = ab \neq a$, then w_2 is not co-LBO, with regard to L_s.*

Table 6. Secret words associated with each intruder.

Intruder	Projection of w_1	Projection of w_2
1	$[1]a[0]$	$[1]a[0][1]$
2	$[1\ 0]^T[0\ 1]^T$	$[1\ 0]^T[0\ 1]^T[1\ 0]^T$
3	ab	a

7.2. Study Case: Temperature Control System

Modern inference techniques, such as machine learning algorithms, can provide enough data to infer user activities, including house occupancy, sleeping patterns, and work schedules. For instance, innocuous data, such as in-home temperatures, could violate the user's privacy of location, which is the right of individuals to be present in a space without being tracked or monitored or without anyone knowing where they are.

We use a smart thermostat as a case study, which controls the heating and air conditioning in Alice and Bob's bedroom, shown in Figure 5. The bedroom mainly includes one bed, a window, a door, an air conditioner (AC), a smart thermostat connected to an HVAC (heating, ventilation, and air conditioning) system, a camera, and sensors to measure the necessary data. The thermostat allows for creating a heating and cooling schedule during the day, to save energy. It can also use sensors to check room occupancy, and it automatically sets itself to a standby temperature when the room is empty. The thermostat temperature-control system is modeled by a POPN, as shown in Figure 6. The system turns off (place p_1) at 8:00 a.m. (transition t_4), when Bob and Alice are outside for work, and starts around 7:00 p.m. (transition t_1), then goes directly into a standby state (place p_3), which prepares the room for the arrival of Bob and Alice in the evening. From this time on, the system is either in an ON state (place p_2), when the presence sensor detects someone inside the room (place p_4) or in a Standby state, when the presence sensor does not detect anyone inside the room. Note that the token number in place p_5 indicates the number of persons inside the bedroom. When the system goes into the ON state, a quick user identification process starts (place p_5), based on a smart facial recognition camera powered by artificial intelligence and placed in the room's upper left corner. At the end of this process, there are four possibilities:

- Bob is identified (transition t_9); then, his preferred reference temperature T_1^o (place p_6) will be displayed.
- Alice is identified (transition t_{10}) in the room. In this case, the preferred reference temperature will be T_2^o (place p_7).
- Bob and Alice are simultaneously present in the room (transition t_{11}); then, the reference temperature T_3^o (place p_8), on which Bob and Alice have agreed, will be displayed.
- Neither Bob nor Alice is identified (transition t_8).

The chosen temperature will remain activated for a duration of $20s$ (transitions t_{12}, t_{13}, t_{14}). At the end of this period, the system saves a temperature record (place p_9), to keep track of the user's schedule and habits, and it then programs itself to match his temperature patterns.

Now, let us check whether intruders can violate Bob and Alice's privacy of location. Let the secret information be

"Alice is alone inside the bedroom".

Intruder 1 is observing the occupancy sensor's activity, and intruder 2 is observing the thermostat's activity. We have:

$T_{d,1} = \{t_5, t_6\}$
$\tilde{T}_{u,1} = \{t_1, t_2, t_3, t_4, t_7, t_8, t_9, t_{10}, t_{11}, t_{12}, t_{13}, t_{14}, t_{15}\}$
$T_{d,2} = \{t_2, t_3, t_7, t_9, t_{10}, t_{11}, t_{12}, t_{13}, t_{14}, t_{15}, t_1, t_4\}$
$\tilde{T}_{u,2} = \{t_5, t_6, t_8\}$.

When Alice enters the bedroom at 7:30 pm, transition sequence $t_1t_6t_3t_7t_8$ fires, and intruder 1 observes $w_1 = [0][1]$, which makes him conclude that someone is alone inside the room, but he is not sure whether it is Alice or not.

On the other hand, intruder 2 observes $w_2 = [100000]^T a [001000]^T [010000]^T [000000]^T [000010]^T$, and the presence of a token in places p_7 indicates that Alice is inside the room, but intruder 2 is not sure whether she is alone or not. We check the opacity for each intruder separately, using Theorem 1. We find that both ILPPs $\mathcal{G}(w_1)$ and $\mathcal{G}(w_2)$ admit a feasible solution. Therefore, the temperature-control system is LBO, with regard to to intruders 1 and 2. However, there is no solution available for ILPP (10). Consequently, the coordination between intruders 1 and 2 can disclose the secret information and infer that Alice is alone inside the room.

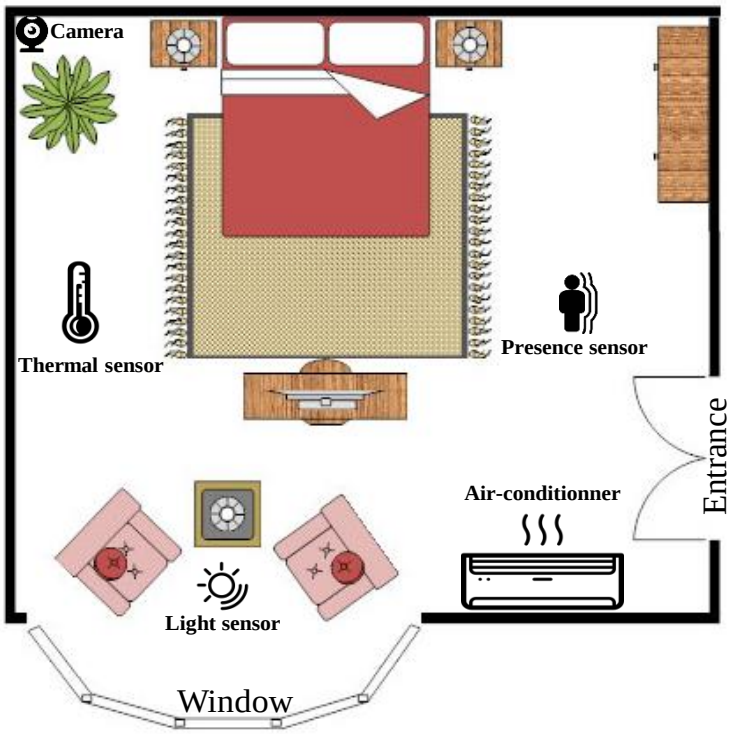

Figure 5. Bedroom layout.

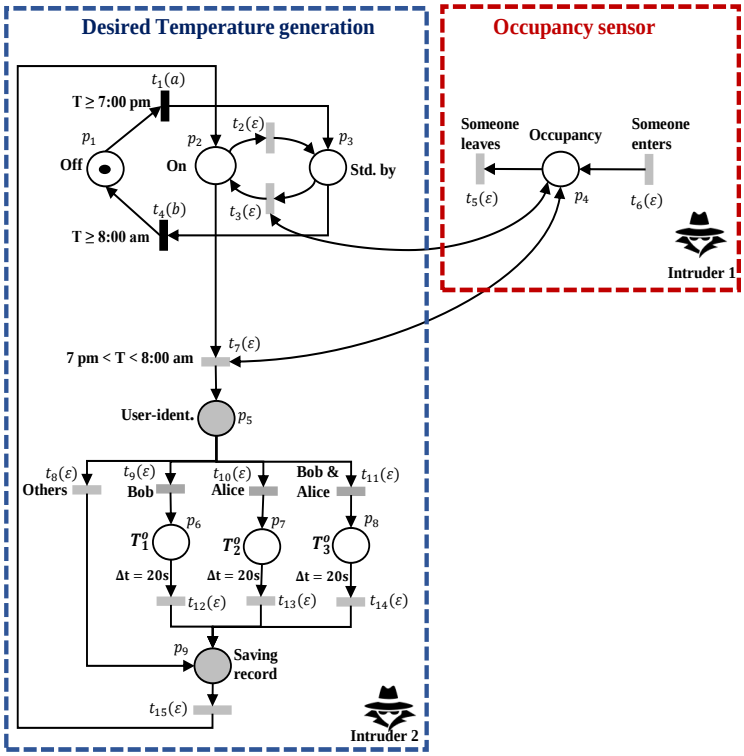

Figure 6. PN-based temperature generator.

8. Conclusions and Future Work

The formulation and verification of LBO in the context of DESs modeled with POPNs were discussed in this research. We provided a necessary and sufficient condition for LBO verification, based on a mathematical description of a net system. Specifically, we proposed linear constraints to check the consistency and non-secrecy aspects of transition sequences. Moreover, these findings were extended to encompass scenarios of decentralized opacity. In these scenarios, the system was examined by multiple intruders, who subsequently shared their observation outcomes with a coordinator, in order to disclose a secret behavior. The proposed approach is more general and scalable than the existing methods for LBO verification in LPNs, as it can be applied to both bounded and unbounded POPNs without requiring any expensive offline computation.

Our future study will extend the proposed ILP-based approach to fault diagnosis in a POPN. Furthermore, it would be of interest to investigate other security problems and potential vulnerabilities, such as unauthorized access, cyber-attacks, intrusion detection, prevention, etc.

Author Contributions: Methodology, I.S. and A.L.; software, A.L.; formal analysis, H.D.; resources, A.M.E.-S.; supervision, Z.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by the Guangzhou Innovation and Entrepreneurship Leading Team Project Funding under Grant No. 202009020008 and the Science and Technology Fund, FDCT, Macau SAR, under Grant 0064/2021/A2. The authors extend their appreciation to King Saud University, Saudi Arabia, for funding this work through Researchers Supporting Project number (RSP2023R133), King Saud University, Riyadh, Saudi Arabia.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Gaudin, B.; Marchand, H. Supervisory control and deadlock avoidance control problem for concurrent discrete event systems. In Proceedings of the 44th IEEE Conference on Decision and Control, Seville, Spain, 12–15 December 2005 ; pp. 2763–2768.
2. Jiao, T.; Wonham, W. Composite supervisory control for symmetric discrete-event systems. *Int. J. Control* **2020**, *93*, 1630–1636. [CrossRef]
3. Bryans, J.W.; Koutny, M.; Mazaré, L.; Ryan, P.Y. Opacity generalised to transition systems. In Proceedings of the International Workshop on Formal Aspects in Security and Trust, Newcastle upon Tyne, UK, 18–19 July 2005; pp. 81–95.
4. Basile, F.; De Tommasi, G.; Sterle, C. Non-interference enforcement in bounded Petri nets. In Proceedings of the 2018 IEEE Conference on Decision and Control (CDC), Miami Beach, FL, USA, 17–19 December 2018; pp. 4827–4832.
5. Lin, F. Opacity of discrete event systems and its applications. *Automatica* **2011**, *47*, 496–503. [CrossRef]
6. Lafortune, S.; Lin, F.; Hadjicostis, C.N. On the history of diagnosability and opacity in discrete event systems. *Annu. Rev. Control* **2018**, *45*, 257–266. [CrossRef]
7. Saadaoui, I.; Li, Z.; Wu, N. Current-state opacity modelling and verification in partially observed Petri nets. *Automatica* **2020**, *116*, 108907. [CrossRef]
8. Ben-Kalefa, M.; Lin, F. Opaque superlanguages and sublanguages in discrete event systems. *Cybern. Syst.* **2016**, *47*, 392–426. [CrossRef]
9. Badouel, E.; Bednarczyk, M.; Borzyszkowski, A.; Caillaud, B.; Darondeau, P. Concurrent secrets. *Discret. Event Dyn. Syst.* **2007**, *17*, 425–446. [CrossRef]
10. Tong, Y.; Ma, Z.; Li, Z.; Seactzu, C.; Giua, A. Verification of language-based opacity in Petri nets using verifier. In Proceedings of the American Control Conference, Boston, MA, USA, 6–8 July 2016; pp. 757–763.
11. Cong, X.; Fanti, M.P.; Mangini, A.M.; Li, Z. On-line verification of current-state opacity by Petri nets and integer linear programming. *Automatica* **2018**, *94*, 205–213. [CrossRef]
12. Wu, Y.C.; Lafortune, S. Comparative analysis of related notions of opacity in centralized and coordinated architectures. *Discret. Event Dyn. Syst.* **2013**, *23*, 307–339. [CrossRef]
13. Basile, F.; De Tommasi, G. An algebraic characterization of language-based opacity in labeled Petri nets. *IFAC-PapersOnLine* **2018**, *51*, 329–336. [CrossRef]
14. Paoli, A.; Lin, F. Decentralized opacity of discrete event systems. In Proceedings of the 2012 American Control Conference (ACC), Montreal, QC, Canada, 27–29 June 2012; pp. 6083–6088.
15. Lin, F.; Wonham, W.M. Decentralized supervisory control of discrete-event systems. *Inf. Sci.* **1988**, *44*, 199–224. [CrossRef]
16. Ramadge, P.J.; Wonham, W.M. The control of discrete event systems. *Proc. IEEE* **1989**, *77*, 81–98. [CrossRef]
17. Tripakis, S.; Rudie, K. Decentralized Observation of Discrete-Event Systems: At Least One Can Tell. *IEEE Control Syst. Lett.* **2022**, *6*, 1652–1657. [CrossRef]
18. Saadaoui, I.; Li, Z.; Wu, N.; Khalgui, M. Depth-first search approach for language-based opacity verification using Petri nets. *IFAC-PapersOnLine* **2020**, *53*, 378–383. [CrossRef]
19. Silva, M.; Terue, E.; Colom, J.M. Linear algebraic and linear programming techniques for the analysis of place/transition net systems. In Proceedings of the Advanced Course on Petri Nets, Dagstuhl, Germany, 7–18 October 1996; pp. 309–373.
20. Cabasino, M.P.; Giua, A.; Pocci, M.; Seatzu, C. Discrete event diagnosis using labeled Petri nets. An application to manufacturing systems. *Control Eng. Pract.* **2011**, *19*, 989–1001. [CrossRef]