Linear Algebraic Techniques for the Analysis of P/T Net Systems

Manuel Silva, Enrique Teruel, and José Manuel Colom *

Dep. Informática e Ingeniería de Sistemas, CPS, Universidad de Zaragoza,
Maria de Luna 3, E-50015 Zaragoza, Spain

Abstract. The structure theory of Place/Transition net systems is surveyed — incorporating new contributions — in a tutorial style, mainly from a linear algebraic perspective. Topics included are: state equation based analysis of safety properties (e.g., boundedness, mutual exclusion, deadlock-freeness, etc.), linear invariants, siphons and traps, implicit places and their application to improve the accuracy of the state equation, and rank theorems (structural conditions for liveness and boundedness based on the rank of the incidence matrix).

Contents

* This work was partially supported by Project TIC-94-0242 of the Spanish CICYT and Contract CHRX-CT94-0452 (MATCH) within the HCM Programme of the EU.

1 Introduction

A *Petri net (PN)* model of a dynamic system, i.e., a *net system*, consists of two parts: A *net structure*, comprising the state variables (places) and their transformers (transitions) and a *marking*, that represents a distributed overall state on the structure. The system dynamics or behaviour is given by the evolution rules for the marking. This separation allows one to reason on net based models at two different levels: *structural* and *behavioural*. From the former we may derive some "fast" conclusions on the possible behaviours of the modelled system. Purely behavioural reasonings can be more conclusive, but they may require costly computations, or even they may not be feasible. The structural reasoning can be regarded as an *abstraction* of the behavioural one: for instance, instead of studying whether a given system, i.e., a net structure with an initial marking, has a finite state space, we might investigate whether the state space is finite *for every* possible initial marking; or we could study whether *there exists* an initial marking that guarantees infinite activity rather than deciding this for a given one, etc.

Two intimately related families of techniques have extensively been used for structural reasoning: *graph theory* and *linear algebra*. In this work we deal with both kinds of techniques from a linear algebraic viewpoint, often expressing or interpreting in linear algebraic terms some classical graph theory based notions (such as those related to siphons and traps). We are interested in giving a general framework, conceptually simple and reasonably efficient, rather than presenting the most efficient algorithm for each particular property (as an example, liveness and boundedness of a free choice system can be decided in polynomial time after Corollary 50, but a graph based algorithm [46] performs more efficiently).

The material is presented mainly in a tutorial style, covering the main developments in the field from the seventies and also introducing some new perspectives or revisiting previous works. In order to improve readability, most proofs are given as previous explanations of the results, concepts and results are illustrated by several examples, additional information is contained in separated remarks, and bibliographical remarks have been collected in a final section.

The basic notions are recalled in Section 2, where also the main notations are introduced. (A brief recall to notions and results in linear programming and duality theory that are used throughout the paper is included as an appendix.)

The starting point for structure theory is the description of the behaviour of the system in structural terms, based on the net *state equation* and other structural objects. The presentation and comparison of these concepts forms Section 3.

Section 4 covers the analysis of important safety properties of net systems (e.g., boundedness, mutual exclusion, deadlock-freeness, etc.) through the state equation. This method is at the same time more efficient and accurate than the classical *invariant method*, which on the other hand has salient merits for the understanding. Special emphasis is given to show the bridge between results in the fields of structure theory and linear algebra/convex geometry. For instance,

classical results from the invariant method are derived applying duality theory to the linear programming formulations based on the state equation.

A major limitation of the state equation method to analyse net systems is the fact that structural descriptions of the behaviour, particularly those derived from the net state equation, are, in general, *relaxations*. Owing to this, the analysis allows only to semidecide the corresponding properties, i.e., find only necessary or sufficient conditions. Section 5 presents several techniques to improve the accuracy of structural descriptions, hence the resolution of structural methods.

Another limitation of the state equation method is that it is best suited to analyse safety properties, i.e., existence or non-existence of markings (and firing vectors). Other properties, particularly transition's liveness, cannot be dealt with directly. Nevertheless, some structural results based on the incidence matrix and the conflict structure of the net are helpful for this analysis, specially in the case of some net subclasses, as it is shown in Section 6.

The paper is concluded with bibliographical remarks in Section 7, to give an impression on the development of the field and to point at related topics that have not been exhaustively covered.

2 Nets and Net Systems: Basic Concepts and Notation

2.1 Vector and Matrix Notations

We denote vectors as $\mathbf{v} = [v_i]$; v_i is the i-th component of $\mathbf{v}$, alternatively written $\mathbf{v}[i]$. For matrices, we have $\mathbf{C} = [c_{ij}]$ and $\mathbf{C}[i,j] = c_{ij}$. Most often in net theory, vectors and matrices are indexed by the (arbitrarily) ordered sets of places and transitions. For instance, if $p \in P$ and $t \in T$, $\mathbf{C}[p,t]$ denotes the entry of $\mathbf{C}$ corresponding to row p and column t; if $P' \subseteq P$ and $T' \subseteq T$ we can write $\mathbf{C}[P',T']$ to refer to the submatrix of $\mathbf{C}$ corresponding to rows from P' and columns from T'; the column of $\mathbf{C}$ corresponding to transition $t \in T$ would be $\mathbf{C}[P,t]$. We often describe markings and other rather sparse vectors using a bag/formal sum notation. For instance, a marking that puts two tokens in p_1 and one in p_3 is denoted $2p_1 + p_3$ instead of $[2\,0\,1\,0\,\cdots\,0]$.

The transpose of a matrix is denoted by $\mathbf{C}^{\perp}$. (The transpose of a vector is not defined, i.e., vectors are not considered one-row or one-column matrices.) Operations are denoted as usual. For instance, $k \cdot \mathbf{C} = [k \cdot c_{ij}]$, $\mathbf{v} \cdot \mathbf{v}' = \sum_i v_i \cdot v'_i$, $\mathbf{v} \cdot \mathbf{C} = [\mathbf{v} \cdot \mathbf{C}[\cdot,i]]$, and $\mathbf{C} \cdot \mathbf{v} = [\mathbf{C}[i,\cdot] \cdot \mathbf{v}]$. Relational operators applied on vectors or matrices are interpreted componentwise. For instance, $\mathbf{v} > \mathbf{v}'$ means that $v_i > v'_i$ for every i. The following will be used too: $\mathbf{v} \gneqq \mathbf{v}'$ means that $v_i \geq v'_i$ for every i and some i exists such that $v_i > v'_i$ (not to be confused with $\mathbf{v} \ngeq \mathbf{v}'$ meaning that $\mathbf{v} \geq \mathbf{v}'$ is false). The support of a vector $\mathbf{v}$ — the set of indices of non-null elements — is denoted by $\|\mathbf{v}\|$.

We denote $\mathbf{0}$ and $\mathbf{1}$ the vectors/matrices with every entry equal to zero and one, respectively, $\mathbf{I}$ the identity matrix, $\mathbf{1}_i$ the vector whose only non null entry is i, which takes value one (a characteristic vector), and $\mathbf{1}_S = \sum_{i \in S} \mathbf{1}_i$ (the characteristic vector of S).

2.2 Place/Transition Net Systems

We concentrate here on the formalism of *Place/Transition (P/T) net systems*. We denote a *P/T net* as $\mathcal{N} = \langle P, T, \mathbf{Pre}, \mathbf{Post} \rangle$, where P and T are the sets of *places* and *transitions*, and **Pre** and **Post** are the $|P| \times |T|$ sized, natural valued, *incidence matrices*. $\mathbf{Post}[p,t] = w$ means that there is an *arc* from t to p with *weight* (or *multiplicity*) w, and $\mathbf{Pre}[p,t] = 0$ indicates no arc from p to t. (We assume without loss of generality that nets are *connected*.)

A *marking* is a $|P|$ sized, natural valued, vector. A *P/T system* is a pair $\mathcal{S} = \langle \mathcal{N}, \mathbf{m_0} \rangle$, where $\mathbf{m_0}$ is the *initial marking*. A transition t is *enabled* at $\mathbf{m}$ iff $\mathbf{m} \geq \mathbf{Pre}[P,t]$; its *occurrence* or *firing*, denoted by $\mathbf{m} \xrightarrow{t} \mathbf{m}'$, yields a new marking $\mathbf{m}' = \mathbf{m} + \mathbf{C}[P,t]$, where $\mathbf{C} = \mathbf{Post} - \mathbf{Pre}$ is called the *token flow matrix*. (In *pure nets*, i.e., without self-loops, positive and negative entries in $\mathbf{C}$ completely represent the post- and pre- incidence functions, and then $\mathbf{C}$ can be properly called the *incidence matrix* of the pure net.)

An *occurrence sequence* from $\mathbf{m}$ is a sequence of transitions $\sigma = t_1 \cdots t_k \cdots$ such that $\mathbf{m} \xrightarrow{t_1} \mathbf{m}_1 \cdots \mathbf{m}_{k-1} \xrightarrow{t_k} \cdots$. The set of all the occurrence sequences, or *language*, from $\mathbf{m_0}$ is denoted by $\mathrm{L}(\mathcal{N}, \mathbf{m_0})$, and the set of all the reachable markings, or *reachability set*, from $\mathbf{m_0}$, is denoted by $\mathrm{RS}(\mathcal{N}, \mathbf{m_0})$. The reachability relation is conventionally represented by a *reachability graph* $\mathrm{RG}(\mathcal{N}, \mathbf{m_0})$ where the nodes are the reachable markings and there is an arc labeled t from node $\mathbf{m}$ to $\mathbf{m}'$ iff $\mathbf{m} \xrightarrow{t} \mathbf{m}'$.

For pre- and postsets we use the conventional dot notation, e.g., ${}^\bullet t = \{p \in P \mid \mathbf{Pre}[p,t] \neq 0\}$. A transition t such that $|t^\bullet| > 1$ (resp. $|{}^\bullet t| > 1$) is called a *fork* (resp. a *join*). A place p such that $|{}^\bullet p| > 1$ (resp. $|p^\bullet| > 1$) is called an *collector* (resp. a *distributor*). Distributor places are required to model conflicts. The output transitions of a distributor place are said to be in *structural conflict* relation. The *coupled conflict* relation is defined as the transitive closure of the structural conflict relation. The equivalence class (or *coupled conflict set*) of transition t is denoted by $\mathrm{CCS}(t)$ and the quotient set is SCCS. When $\mathbf{Pre}[P,t] = \mathbf{Pre}[P,t'] \neq \mathbf{0}$, t and t' are in *equal conflict* (EQ) relation, meaning that they are both enabled whenever one is. This is also an equivalence relation on the set of transitions. The equivalence class (or *equal conflict set*) of transition t is denoted by $\mathrm{EQS}(t)$ and the quotient set is SEQS.

Weighted (or multiple) arcs permit the abstract modelling of bulk services and arrivals. For instance, they appear naturally when the presence of symmetries allows to "decolour" a high level model. If $\mathbf{Pre}[p, p^\bullet] = w\mathbf{1}$ we say that the weighting is *homogeneous* on p (e.g., the weighting of the input places of an equal conflict set). If this holds for every place, the weighting of the net is homogeneous. A historically and conceptually interesting subclass of P/T nets with homogeneous weighting are *ordinary* nets, where every arc weight is one, which lead to a straightforward but important generalisation of automata models. Although it is possible to simulate/implement weighted P/T systems by ordinary ones preserving the (projected) language (with transformations like the one shown in Figure 12 later in the paper), several reasons justify dealing with weighted P/T systems directly rather than with their ordinary implementations: the models

are more concise, the transformations do not preserve concurrent semantics appropriately in general, and the ordinary implementations fall typically out of the subclasses which enjoy strong analytical results, even in the simplest cases.

A subset of places $\Theta \subseteq P$ such that $\Theta^\bullet \subseteq {}^\bullet\Theta$ is called a *trap* because once it becomes marked it remains marked (tokens are "trapped"). A subset of places $\Sigma \subseteq P$ such that ${}^\bullet\Sigma \subseteq \Sigma^\bullet$ is called a *siphon* because once it becomes unmarked it remains unmarked (it cannot be "refilled" with tokens).

By reversing arcs or interchanging places and transitions we get the *reverse net*, $\mathcal{N}^r$, or the *dual net*, $\mathcal{N}^d$, of $\mathcal{N}$. Both transformations together lead to the *reverse-dual net*, $\mathcal{N}^{rd}$. Sometimes in net theory relations are established between a net and its reverse, dual, or reverse-dual, e.g., siphons and traps are reverse objects: a siphon of $\mathcal{N}$ is a trap of $\mathcal{N}^r$, etc.

$\mathcal{N}$	$\langle P, T, \mathbf{Pre}, \mathbf{Post}\rangle$	$\mathbf{C}$
$\mathcal{N}^r$	$\langle P, T, \mathbf{Post}, \mathbf{Pre}\rangle$	$-\mathbf{C}$
$\mathcal{N}^d$	$\langle T, P, \mathbf{Post}^\perp, \mathbf{Pre}^\perp\rangle$	$-\mathbf{C}^\perp$
$\mathcal{N}^{rd}$	$\langle T, P, \mathbf{Pre}^\perp, \mathbf{Post}^\perp\rangle$	$\mathbf{C}^\perp$

A net $\mathcal{N}'$ is *subnet* of $\mathcal{N}$ (written $\mathcal{N}' \subseteq \mathcal{N}$) when $P' \subseteq P, T' \subseteq T$ and its pre- and post-incidence matrices are $\mathbf{Pre}' = \mathbf{Pre}[P', T']$ and $\mathbf{Post}' = \mathbf{Post}[P', T']$. (In what follows, for the sake of readability, whenever a net or system is defined it "inherits" the definition of all the characteristic sets, functions, parameters,... with names conveniently marked.) Subnets are generated by subsets of places *and* transitions. When a subnet is generated by a subset V of nodes of a single kind, it is assumed that it is generated by $V \cup {}^\bullet V \cup V^\bullet$. Subnets generated by a subset of places (transitions) are called P- (T-) subnets.

2.3 Analysis of Logical Properties

A major goal of the mathematical modelling of systems is to allow their automatic analysis. In general, verification consists in checking that a system model satisfies its logic specification (e.g., some temporal logic formulae). Here we are interested in the verification of some selected properties of "good behaviour" that are often part of the specification of systems (specially reactive ones), or appear as precondition for the temporal analysis or performance evaluation of the interpreted model.

A P/T system is *bounded* when every place is bounded, i.e., its token content is less than some bound at every reachable marking (when the bound is one, then it is said to be *safe*). It is *live* when every transition is live, i.e., it can ultimately occur from every reachable marking, and it is *deadlock-free* when every reachable marking enables some transition. A marking is a *home state* when it is reachable from every reachable marking, and a net system is *reversible* when the initial marking (hence every marking) is a home state. Two places are in *mutual exclusion* when they are never marked simultaneously. Boundedness precludes overflows, liveness ensures that no single action in the system can become unattainable, existence of home states informs on the possibility to return to

certain states, and mutual exclusion is required between places that represent the use of a common resource or the presence in a critical section.

Conventionally, analysis methods of PN models are classified as follows:

- Enumeration Techniques: If the system is bounded, the reachability graph can be used as the computational model for a proof system or for decision procedures and tools for automatic verification. Two major problems of this approach are the size of the state space of a concurrent system, that can be palliated in some cases where not every state needs to be computed, and the necessity to repeat the analysis for each initial marking of interest. Unbounded systems can be partially analysed using a similar approach.
- Transformation Techniques: To facilitate the analysis of a large and complex system it can be transformed (typically reduced) preserving the properties to be analysed. Transformation rules somehow preserve the behaviour while they are often supported by structural arguments as simple, and efficient, sufficient conditions.
- Structural Techniques: The basic idea is to obtain useful information about the behaviour reasoning on the structure of the net and the initial marking. Two crucial advantages of this approach are the *deep understanding* of the system behaviour that is gained, and the possible *efficiency* of the algorithms. Two intimately related families of techniques have extensively been used: graph theory and linear algebra/convex geometry. The rest of the paper is devoted to describe these techniques in some detail from a linear algebraic viewpoint (over the non-negative integers or reals).

The above groups of techniques are not to be understood as mutually exclusive, but they should be effectively combined for the analysis in practice.

General net systems are difficult to analyse. As in all theories, it is a common trend in net theory to consider particular *subclasses* of models by introducing appropriate restrictions, either on the behaviour or the structure (or syntax) of the model. A possible way of obtaining syntactical subclasses is restricting the inscriptions (e.g., nets with every weight equal to one are ordinary) or the topology, usually aiming at limiting the interplay between conflicts and synchronisations. The latter can be achieved either by giving a general restriction, typically on distributor places and/or join transitions (e.g., there are no distributors), or by giving rules to construct models (e.g., sequential functional entities are synchronised by some restricted message passing). These restrictions are intended to facilitate the analysis (we shall give some examples through the paper) at the price of losing some modelling capabilities. The designer must find a compromise between modelling power and availability of powerful analysis tools, while one of the theoretician's goals is obtaining better results for increasingly larger — and more practical — subclasses.

3 Linear Descriptions and Structural Objects

The starting point for the structural analysis of P/T systems by linear algebraic techniques is the description of the state space by some system of linear equa-

tions. In fact, as we shall see, these descriptions are often *relaxations* with a different degree of accuracy. In principle, there is a trade-off between the accuracy of the description and the efficiency of verification algorithms, but this is not necessarily the case: in some net subclasses, relaxations that — in general — are less accurate than others describe exactly the state space; it is also worth noticing that some linear descriptions that are extensively used are less accurate *and* less efficient than others (although they may have salient merits for the understanding).

3.1 The State Equation

Recall that when a transition t is enabled at $\mathbf{m}$ ($\mathbf{m} \geq \mathbf{Pre}[P,t]$), the new marking reached by its firing ($\mathbf{m} \xrightarrow{t} \mathbf{m}'$) is $\mathbf{m}' = \mathbf{m} + \mathbf{C}[P,t]$. Analogously, a *step* $\mathbf{s}$ (where $\mathbf{s}[t]$ is the number of times t occurs in that step) is enabled when $\mathbf{m} \geq \mathbf{Pre} \cdot \mathbf{s}$, and its firing leads to:

$$\mathbf{m}' = \mathbf{m} + \mathbf{C} \cdot \mathbf{s} \tag{1}$$

This equation resembles the state equation of a *discrete-time linear system*, where $\mathbf{m}$ is the *current state (vector)*, $\mathbf{m}'$ is the *next state*, and $\mathbf{s}$ is the *inputs vector* (there is one input per transition). Differently from general linear systems, here the *dynamic matrix* is identity (all the eigenvalues are one, what corresponds to integrators or counters: the state is memorised in the absence of inputs) and not every action is possible in a given state because inputs and state variables are defined to be non-negative integers.

Assume that it is known that every marking $\mathbf{m}$ is reachable in (at most) k steps from $\mathbf{m_0}$ (this is true for bounded systems, even it is possible to structurally compute a finite such k when the net is *structurally bounded*). Then we could describe *exactly* the set of reachable markings by the following system of linear inequalities ($\mathbf{m}_i \in \mathbb{N}^{|P|}, \mathbf{s}_i \in \mathbb{N}^{|T|}$):

$$\begin{aligned} \mathbf{m_0} &\geq \mathbf{Pre} \cdot \mathbf{s}_0 \\ \mathbf{m}_1 &= \mathbf{m_0} + \mathbf{C} \cdot \mathbf{s}_0 \\ &\cdots \\ \mathbf{m}_{k-1} &\geq \mathbf{Pre} \cdot \mathbf{s}_{k-1} \\ \mathbf{m} &= \mathbf{m}_{k-1} + \mathbf{C} \cdot \mathbf{s}_{k-1} \end{aligned} \tag{2}$$

Although we can easily eliminate the $\mathbf{m}_i$ variables by substitution, this linear description is of course highly impractical due to the size of k, that moreover depends on the initial marking. In what follows, we look for more concise linear descriptions — although we expect that they are not exact descriptions but only more or less accurate approximations.

If we integrate Equation (1) over a sequence σ of inputs (transitions or steps) from the initial state $\mathbf{m_0}$ and yielding $\mathbf{m}$, denoting by $\boldsymbol{\sigma}$ the *firing count vector* of sequence σ ($\boldsymbol{\sigma}[t] = \#(t,\sigma)$ is the number of times t occurs in the sequence) we obtain:

$$\mathbf{m} = \mathbf{m_0} + \mathbf{C} \cdot \boldsymbol{\sigma} \tag{3}$$

Since every reachable marking is obtained by the occurrence of some sequence from the initial marking, it is clear that, for every reachable marking (state) there exists some $\boldsymbol{\sigma} \in \mathbb{N}^{|T|}$ such that Equation (3) holds. This is why (3) is referred to as the *net state equation*.

It is straightforward to derive the following linear description of the set of reachable markings (notice that integrality of $\boldsymbol{\sigma}$ ensures integrality of $\mathbf{m}$):

Definition 1. Let $\mathcal{S}$ be a P/T system. Its *linearised reachability set (using the state equation)* is defined as:

$$\mathrm{LRS}^{\mathrm{SE}}(\mathcal{S}) = \{\mathbf{m} \in \mathbb{N}^{|P|} \mid \exists \boldsymbol{\sigma} \in \mathbb{N}^{|T|} \text{ such that } \mathbf{m} = \mathbf{m_0} + \mathbf{C} \cdot \boldsymbol{\sigma}\}$$

This description is suitable for the incorporation of the state equation into a set of linear constraints, e.g., in the restrictions of an integer programming problem, for analysis purposes (see Section 4).

Remark 2. Although integer programming is NP-complete, some analysis problems can be solved efficiently using the state equation over the integers. For instance, a sufficient condition for non reachability of a given marking $\mathbf{m}$ in $\mathcal{S}$ is non existence of $\boldsymbol{\sigma}' \in \mathbb{Z}^{|T|}$ such that $\mathbf{C} \cdot \boldsymbol{\sigma}' = \mathbf{m} - \mathbf{m_0}$, which is polynomial time [45,76]. Actually, if $\mathcal{N}$ is *consistent*, i.e., an $\mathbf{x} > \mathbf{0}$ exists such that $\mathbf{C} \cdot \mathbf{x} = \mathbf{0}$, this is equivalent to $\mathbf{m} \notin \mathrm{LRS}^{\mathrm{SE}}(\mathcal{S})$, because from a $\boldsymbol{\sigma}' \in \mathbb{Z}^{|T|}$, a $\boldsymbol{\sigma} \in \mathbb{N}^{|T|}$ can be obtained as $\boldsymbol{\sigma} = \boldsymbol{\sigma}' + k\mathbf{x}$. □

The inclusion $\mathrm{RS}(\mathcal{S}) \subseteq \mathrm{LRS}^{\mathrm{SE}}(\mathcal{S})$ may well be proper, since Equation (3) does not check whether there is a sequence of intermediate markings such that some $\sigma \in \mathrm{L}(\mathcal{S})$ with firing count vector $\boldsymbol{\sigma}$ is actually fireable. (In other words, we have removed from (2) the inequalities requiring the fireability of the steps, i.e., the non-negativity of the intermediate $\mathbf{m}_i$ variables, and then we have eliminated these variables.) The markings in $\mathrm{LRS}^{\mathrm{SE}}(\mathcal{S}) - \mathrm{RS}(\mathcal{S})$ will be called *spurious (with respect to the state equation)*.

Similarly to the reachability graph, we can represent a *linearised reachability graph (using the state equation)*, $\mathrm{LRG}^{\mathrm{SE}}(\mathcal{S})$, where the nodes are the markings in $\mathrm{LRS}^{\mathrm{SE}}(\mathcal{S})$ and there is na arc labeled t from node $\mathbf{m}$ to $\mathbf{m}'$ iff $\mathbf{m} \xrightarrow{t} \mathbf{m}'$. Figure 1 shows a P/T system together with its $\mathrm{LRG}^{\mathrm{SE}}$, where the spurious markings are shaded. As another example, the marking p_2 is spurious in the system shown in Figure 2 top-left; it is reached by the "occurrence" of $t_1 + t_2 + t_3$ (notice that in every possible "sequence" with this firing count vector some intermediate marking variable becomes negative).

We can further relax the description by dropping integrality constraints, as it is typical in the mathematical modelling of systems with large state spaces (e.g., population models). This further relaxation introduces more spurious solutions. For instance, the marking $2p_2$ is spurious in the system shown in Figure 2 bottom-left; it is reached by the "occurrence" of $0.5t_1 + t_2 + t_3$. On the other hand, this relaxation allows to use *linear* programming instead of *integer* programming in the verification, leading to polynomial time algorithms. Sometimes

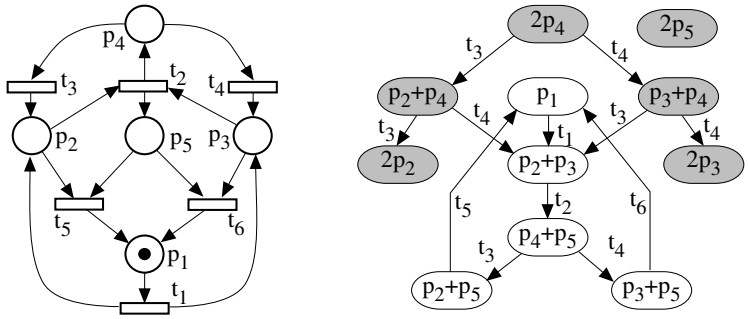

Fig. 1. A P/T system and its $\mathrm{LRG}^{\mathrm{SE}}$.

only the integrality of firing variables is disregarded, either because other restrictions guarantee the integrality of the marking or because mixed integer linear programming is used. As an example, the following linear programming problem can be used to analyse reachability of a given marking $\mathbf{m}$ (if the problem is infeasible, then $\mathbf{m}$ is proven unreachable):

$$\max\{\mathbf{0}\cdot\boldsymbol{\sigma} \mid \ \mathbf{C}\cdot\boldsymbol{\sigma} = \mathbf{m}-\mathbf{m_0} \wedge \boldsymbol{\sigma}\geq \mathbf{0}\} \tag{4}$$

In summary, we define a second linear description of the set of reachable markings:

Definition 3. Let $\mathcal{S}$ be a P/T system. Its *linearised reachability set (using the state equation over the reals)* is defined as:

$$\mathrm{LRS}^{\mathrm{SE}\mathbb{R}}(\mathcal{S}) = \{\mathbf{m}\in\mathbb{N}^{|P|} \mid \ \exists\boldsymbol{\sigma}\geq\mathbf{0} \text{ such that } \mathbf{m}=\mathbf{m_0}+\mathbf{C}\cdot\boldsymbol{\sigma}\}$$

Remark 4. We have relaxed the description of the state space applying two principles: *path integration* and *continuisation* (or *fluidisation*). These principles can also be applied in the reverse order, first continuisation and then path integration, leading again to $\mathrm{LRS}^{\mathrm{SE}\mathbb{R}}$. By disregarding first the integrality of variables, we get *continuous P/T net systems* [27]. In these models, "fluid tokens" are contained in "deposits" (the places), the "level" of which (the marking) captures the state of the system (as in *Forrester diagrams* [26]). Transitions are regarded as "mixing valves" whose firing (opening) consumes fluid from the input places and produces fluid onto the output places in a given proportion, according to the following firing rule: t is enabled in some amount $\lambda>0$ at marking $\mathbf{m}$ when $\mathbf{m}\geq\lambda\mathbf{Pre}[P,t]$, and its occurrence leads to the marking $\mathbf{m}'=\mathbf{m}+\lambda\mathbf{C}[P,t]$. The set of reachable markings of system $\mathcal{S}$ is denoted by $\mathrm{CRS}(\mathcal{S})$, standing for *continuous reachability set*. These nets are interesting in the modelling of certain continuous systems, and also as an approximation of systems where there are large amounts of (discrete) tokens. When used as an approximation, they naturally suffer from the presence of spurious solutions. For instance, the marking $2p_2$ is spurious in the system shown in Figure 2 top-right; it is reached by the "occurrence" of $0.5t_2$. □

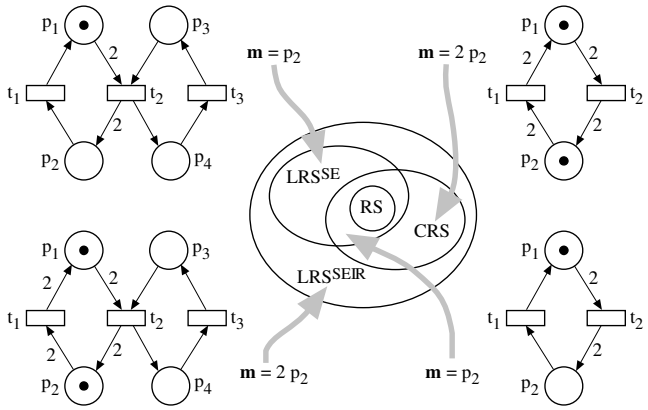

Fig. 2. Diverse relaxations of the state space applying path integration and continuisation.

Figure 2 illustrates the relaxations we have discussed so far. Notice that the spurious markings induced by the use of LRS^{SE} or CRS are not necessarily disjoint: the marking p_2 is spurious in the system shown in Figure 2 bottom-right; it is reached by the "occurrence" of either $t_1 + t_2$ or $0.5t_2$.

Clearly, the presence of spurious solutions may prevent reaching conclusions using analysis techniques based on a relaxed description of the state space. We shall give later techniques to remove spurious solutions by adding more information to the state equation after carefully considering the net structure. It is also worth noticing that some net subclasses enjoy special properties on their spurious solutions that palliate the problem, as we shall illustrate (e.g., see Subsection 3.4)

3.2 Linear Invariants from the State Equation

Since every reachable marking must satisfy the state equation, it can be regarded as a set of *linear invariant laws*, one per place, containing marking and firing variables. The firing variables can be eliminated, by multiplying the equation by a suitable vector, in order to obtain linear invariant laws involving marking variables only. For instance, if $\mathbf{y}$ is such that $\mathbf{y} \cdot \mathbf{C} = \mathbf{0}$ — vector $\mathbf{y}$ is called a *P-flow* — then, *for every initial marking* $\mathbf{m_0}$, every reachable marking $\mathbf{m}$ satisfies:

$$\mathbf{y} \cdot \mathbf{m} = \mathbf{y} \cdot \mathbf{m_0} + \mathbf{y} \cdot \mathbf{C} \cdot \boldsymbol{\sigma} = \mathbf{y} \cdot \mathbf{m_0} = k$$

This provides a "token balance law": if the positive and negative parts of $\mathbf{y}$ are separated: $\mathbf{y} = \mathbf{y}_+ - \mathbf{y}_-$, where $\mathbf{y}_+, \mathbf{y}_- \geq \mathbf{0}$, then for every reachable marking $\mathbf{y}_+ \cdot \mathbf{m} = \mathbf{y}_- \cdot \mathbf{m} + k$, that is, the tokens in $\|\mathbf{y}_+\|$ and $\|\mathbf{y}_-\|$ are somehow "balanced". Conversely, in net systems where all transitions can fire at least once, *every* linear token conservation law is associated with a P-flow. This can

be deduced as follows: Assume $\mathbf{y} \cdot \mathbf{m} = \mathbf{y} \cdot \mathbf{m_0}$ holds for every reachable marking. It must be shown that $\mathbf{y} \cdot \mathbf{C} = \mathbf{0}$. Consider an arbitrary t and $\mathbf{m}_t \xrightarrow{t} \mathbf{m}'_t$. Then $\mathbf{m}'_t = \mathbf{m}_t + \mathbf{C}[P, t]$, hence $\mathbf{y} \cdot \mathbf{m}'_t = \mathbf{y} \cdot \mathbf{m}_t + \mathbf{y} \cdot \mathbf{C}[P, t]$. Since $\mathbf{y} \cdot \mathbf{m}'_t = \mathbf{y} \cdot \mathbf{m}_t$, it follows that $\mathbf{y} \cdot \mathbf{C}[P, t] = 0$. In summary:

Theorem 5. *Let $\mathcal{N}$ be a P/T net.*

1. *If $\mathbf{y} \cdot \mathbf{C} = \mathbf{0}$ then for every $\mathbf{m_0}$: $\mathbf{y} \cdot \mathbf{m} = \mathbf{y} \cdot \mathbf{m_0}$ for every $\mathbf{m} \in \mathrm{RS}(\mathcal{N}, \mathbf{m_0})$.*
2. *Let $\mathcal{S} = \langle \mathcal{N}, \mathbf{m_0} \rangle$. Assume that for every $t \in T$ some $\mathbf{m}_t \in \mathrm{RS}(\mathcal{S})$ exists such that $\mathbf{m}_t \geq \mathbf{Pre}[P, t]$. If $\mathbf{y} \cdot \mathbf{m} = \mathbf{y} \cdot \mathbf{m_0}$ for every $\mathbf{m} \in \mathrm{RS}(\mathcal{S})$, then $\mathbf{y} \cdot \mathbf{C} = \mathbf{0}$.*

The P-flows of a net $\mathcal{N}$ form a vector space. Using $\mathbf{B}$, a matrix whose rows form a basis of P-flows, we obtain a new linear description of the set of reachable markings where only marking variables appear:

Definition 6. Let $\mathcal{S}$ be a P/T system. Its *linearised reachability set (using a basis $\mathbf{B}$ of P-flows)* is defined as:

$$\mathrm{LRS}^{\mathrm{Pf}}(\mathcal{S}) = \{\mathbf{m} \in \mathbb{N}^{|P|} \mid \mathbf{B} \cdot \mathbf{m} = \mathbf{B} \cdot \mathbf{m_0}\}$$

Token balance laws become specially useful when $\mathbf{y} \geq \mathbf{0}$ — in such case, $\mathbf{y}$ is called a *P-semiflow* — because, taking into account that $\mathbf{m} \geq \mathbf{0}$, from $\mathbf{y} \cdot \mathbf{m} = k$ we can deduce, for instance, that all the places in $\|\mathbf{y}\|$ are bounded. The invariants that we obtain are "token conservation laws": for every reachable marking the weighted sum of tokens in $\|\mathbf{y}\|$ remains constant.

Besides the actual invariant law, a major interest of P-semiflows is the *decomposed view* of the model that they provide. The P-subnet generated by the support of a P-semiflow is called a *conservative component* of the net, meaning that it is a part of the net that conserves its weighted token content. In the case that $\mathbf{y} > \mathbf{0}$ such that $\mathbf{y} \cdot \mathbf{C} = \mathbf{0}$ exists, the whole net is a conservative component, and it is said that the net is *conservative*, what obviously implies that it is bounded for every (finite) initial marking.

The token conservation laws induced by P-semiflows are by far the most popular invariant laws, to the point that the classical invariant method considers P-semiflows only, which, historically, are very often called *P-invariants* in the literature. Anyhow, it is important to realise that there are three notions that should be differentiated:

- The P-semiflow (a vector).
- The token conservation law or marking invariant (an equation).
- The conservative component (a net).

Actually, apart from those derived from P-semiflows, there are other invariant laws and components, as it shall be shown.

A P-semiflow $\mathbf{y}$ is said to be *minimal* when the positive y_i are relatively prime and no P-semiflow $\mathbf{y}'$ exists such that $\|\mathbf{y}'\| \subset \|\mathbf{y}\|$. In order to prove properties, only minimal P-semiflows need to be considered because every P-semiflow can be obtained as a non-negative linear combination — possibly with rational coefficients — of minimal P-semiflows. In a net $\mathcal{N}$, the set of all the minimal P-semiflows, called the *fundamental set of P-semiflows*, is unique.

Remark 7. The supports of two minimal P-semiflows are non comparable, since otherwise we would be able to obtain by difference another P-semiflow whose support would be contained in one of the former. Therefore, a bound for the cardinality of the fundamental set of P-semiflows — which is reached in some cases although it is generally quite high — is the number of possible combinations of $\lceil |P|/2 \rceil$ out of $|P|$ elements:

$$\binom{|P|}{\lceil |P|/2 \rceil}$$

□

Algorithm 8 gives a simple procedure to compute the fundamental set of P-semiflows from the incidence matrix of the net. A row $\mathbf{\Phi}[i]$ memorises the coefficients of the positive linear combination of rows of matrix $\mathbf{C}$ which generate $\mathbf{A}[i]$. In Step 3 of the algorithm, all the rows of $\mathbf{A}$ have been made null, so each row $\mathbf{\Phi}[i]$ is a P-semiflow: $\mathbf{\Phi}[i] \cdot \mathbf{C} = \mathbf{0}$.

Algorithm 8 (Computation of P-semiflows)

Input - The incidence matrix $\mathbf{C}$.
Output - A matrix $\mathbf{\Phi}$ whose rows are the fundamental set of P-semiflow.

1. Let $\mathbf{A} = \mathbf{C}$ and $\mathbf{\Phi} = \mathbf{I}$ { $\mathbf{I}$ is the identity matrix of dimension $|P|$ }
2. **for** $i = 1$ **to** $|T|$ **do**
   2.1 Add to the matrix $[\mathbf{\Phi}|\mathbf{A}]$ all rows which are natural linear combinations of pairs of rows of $[\mathbf{\Phi}|\mathbf{A}]$ and which annul the i-th column of $\mathbf{A}$
   2.2 Eliminate from $[\mathbf{\Phi}|\mathbf{A}]$ the rows in which the i-th column of $\mathbf{A}$ is non-null
3. Remove from $\mathbf{\Phi}$ all rows whose support is not minimal, and divide each other by the g.c.d. of its non-null elements

During execution of Algorithm 8, the number of rows in $[\mathbf{\Phi}|\mathbf{A}]$ typically grows beyond the cardinality of the fundamental set of P-semiflows. To improve the efficiency of the algorithm, instead of annulling the columns of $\mathbf{A}$ in their order, some heuristics for the selection of the column to annul next drastically reduce the growth of $[\mathbf{\Phi}|\mathbf{A}]$. Moreover, introducing some tests of non minimality during the execution of the algorithm — including the application of certain rank properties, combining net theory with linear algebraic techniques — it is also possible to discard many rows that cannot lead to a minimal P-semiflow before completing the computations (see [57,22]).

The fundamental set of P-semiflows provides another linear description of the set of reachable markings where only marking variables appear. Let $\mathbf{\Phi}$ be a matrix whose rows are the fundamental set of P-semiflows of $\mathcal{N}$:

Definition 9. Let $\mathcal{S}$ be a P/T system. Its *linearised reachability set (using the fundamental set of P-semiflows)* is defined as:

$$\mathrm{LRS}^{\mathrm{Psf}}(\mathcal{S}) = \{\mathbf{m} \in \mathbb{N}^{|P|} \mid \boldsymbol{\Phi} \cdot \mathbf{m} = \boldsymbol{\Phi} \cdot \mathbf{m_0}\}$$

The dual notion of P-flows are *T-flows* (in the sense that the P-flows of $\mathcal{N}^r$ or $\mathcal{N}^{rd}$ are the T-flows of $\mathcal{N}$). If $\mathbf{x}$ is such that $\mathbf{C} \cdot \mathbf{x} = \mathbf{0}$:

$$\mathbf{m} = \mathbf{m_0} + \mathbf{C} \cdot \mathbf{x} = \mathbf{m_0}$$

T-flows become specially useful when $\mathbf{x} \geq \mathbf{0}$ — in such case, $\mathbf{x}$ is called a *T-semiflow* — because, in that case they correspond to cyclic sequences. (Note that the firing count vector of a cyclic sequence is a T-semiflow, but possibly for a given initial marking it is *not* possible to fire a sequence whose firing count vector is a given T-semiflow.) The fundamental set of T-semiflows can be readily computed applying Algorithm 8 to $\mathbf{C}^{\perp}$. Similarly to P-semiflows, T-semiflows provide an interesting decomposed view of the model. The T-subnet generated by the support of a T-semiflow is called a *consistent component* of the net. With an appropriate initial marking, a consistent component is able to exhibit a cyclic or repetitive behaviour. In the case that $\mathbf{x} > \mathbf{0}$ such that $\mathbf{C} \cdot \mathbf{x} = \mathbf{0}$ exists, the whole net is a consistent component, and it is said that the net is *consistent*. When a net is not consistent it cannot be lively and boundedly marked (see Proposition 10 and Theorem 45). In fact, historically the name consistent is due to the fact that in a live and bounded system, the equation system $\mathbf{C} \cdot \mathbf{x} = \mathbf{0}$ must be (algebraically) consistent.

Generalising flows and semiflows, other multipliers of $\mathbf{C}$ may provide useful information. For instance, a vector $\mathbf{y} \geq \mathbf{0}$ such that $\mathbf{y} \cdot \mathbf{C} \leq \mathbf{0}$ indicates that the weighted token content of the places in $\|\mathbf{y}\|$ cannot be increased (it can be decreased if $\mathbf{y} \cdot \mathbf{C} \lneq \mathbf{0}$). If $\mathbf{y} > \mathbf{0}$ it follows that the net is bounded for whichever initial marking, or *structurally bounded*. Similarly, a vector $\mathbf{x} \geq \mathbf{0}$ such that $\mathbf{C} \cdot \mathbf{x} \geq \mathbf{0}$ indicates that the occurrence of a sequence with firing count vector $\mathbf{x}$ (if some such sequence was fireable) would not decrease the marking, so it could be repeated once and again (the marking would be increased at each execution if $\mathbf{C} \cdot \mathbf{x} \gneq \mathbf{0}$). If $\mathbf{x} > \mathbf{0}$ the net is said to be *structurally repetitive*. A net that can be lively marked is structurally repetitive, because there must be sequences leading from a marking to a greater or equal one (equal in case of boundedness) involving every transition, the firing count vector of which proves structural repetitiveness:

Proposition 10. *If $\mathcal{S}$ is a live P/T system then $\mathcal{N}$ is structurally repetitive. If $\mathcal{S}$ is also bounded then $\mathcal{N}$ is consistent.*

3.3 Proving Properties Through Linear Invariants

Let us illustrate with an example the usability of (minimal) P-semiflows to prove properties, and the decomposed view that they provide.

A production cell and a P/T description of its local controller are shown in Figure 3 (taken from [82]). The places "wait_raw", "load", "op_1", "wait_dep.",

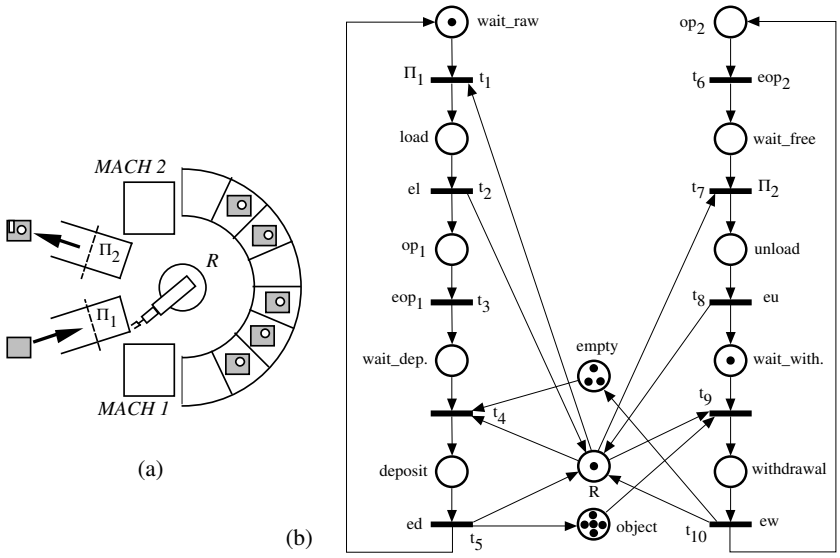

Fig. 3. A production cell with two machines, one robot, and a store, and a P/T description of its behaviour.

and "deposit" represent the possible states of *MACH 1*. The place "R" is marked when the robot is available. The places "empty" and "object" contain as many tokens as empty slots or parts are available in the temporary buffer, etc. In this model actions are associated with places, and transitions represent atomic instantaneous changes of state, e.g., *MACH 2* performs its operations while place "op_2" is marked, and the event of finishing is modelled by the firing of "eop_2".

The marking linear invariants induced by the minimal P-semiflows of the net in Figure 3 are the following:

$$\mathbf{m}[\text{wait\_raw}] + \mathbf{m}[\text{load}] + \mathbf{m}[\text{op}_1] + \mathbf{m}[\text{wait\_dep.}] + \mathbf{m}[\text{deposit}] = 1 \quad (5)$$

$$\mathbf{m}[\text{op}_2] + \mathbf{m}[\text{wait\_free}] + \mathbf{m}[\text{unload}] + \mathbf{m}[\text{wait\_with.}] + \mathbf{m}[\text{withdrawal}] = 1 \quad (6)$$

$$\mathbf{m}[\text{empty}] + \mathbf{m}[\text{deposit}] + \mathbf{m}[\text{object}] + \mathbf{m}[\text{withdrawal}] = 8 \quad (7)$$

$$\mathbf{m}[\text{R}] + \mathbf{m}[\text{load}] + \mathbf{m}[\text{unload}] + \mathbf{m}[\text{deposit}] + \mathbf{m}[\text{withdrawal}] = 1 \quad (8)$$

Since markings are non-negative, the following can be easily stated from the previous equations:

- The marking bound of every place is one, except for "empty" and "object", that is seven.
- The places in each of the following sets are in (pairwise) marking mutual exclusion:
  - {wait_raw, load, op_1, wait_dep., deposit}
  - {op_2, wait_free, unload, wait_with., withdrawal}

- {R, load, unload, deposit, withdrawal}

Using the invariants in (5–8), it is also possible to prove that the net system in Figure 3 is deadlock-free. We proceed by contradiction, more precisely we try to construct a marking $\mathbf{m}$ that satisfies (5–8) and in which no transition is fireable. In such a marking, the places "load", "op_1", "deposit", "op_2", "unload", and "withdrawal" should be unmarked, because these are the only input places of their corresponding transitions, so the token conservation laws in (5–8) reduce to:

$$\mathbf{m}[\text{wait\_raw}] + \mathbf{m}[\text{wait\_dep.}] = 1 \tag{9}$$

$$\mathbf{m}[\text{wait\_free}] + \mathbf{m}[\text{wait\_with.}] = 1 \tag{10}$$

$$\mathbf{m}[\text{empty}] + \mathbf{m}[\text{object}] = 8 \tag{11}$$

$$\mathbf{m}[\text{R}] = 1 \tag{12}$$

Since the above implies that "R" should be marked, to prevent the firing of t_1 and t_7, the places "wait_raw" and "wait_free" should be unmarked too. The token conservation laws are reduced once more, leading to:

$$\mathbf{m}[\text{wait\_dep.}] = 1 \tag{13}$$

$$\mathbf{m}[\text{wait\_with.}] = 1 \tag{14}$$

$$\mathbf{m}[\text{empty}] + \mathbf{m}[\text{object}] = 8 \tag{15}$$

$$\mathbf{m}[\text{R}] = 1 \tag{16}$$

Since the above implies that "wait dep." and "wait with." should be marked, to prevent the firing of t_4 and t_9, both "empty" and "object" should be unmarked, against (15), so the net system is proven deadlock-free. The above "ad hoc" proof is generalised and fully automatised in Subsection 4.6.

As an example of the loss of information when non minimal P-semiflows are used instead of minimal ones, observe that summing up (5–8) we obtain a P-invariant involving all the places which does not allow to prove any of the properties we have deduced from the minimal ones (it allows to prove 10-boundedness of the net, though).

In the example of Figure 3, the only minimal T-semiflow is $\mathbf{1}$, meaning that every cyclic sequence fires all the transitions in the same proportion, so, due to boundedness, in the "long run" all the transitions occur the same number of times per time unit. Therefore, under boundedness, the existence of a unique minimal T-semiflow ensures that deadlock-freeness implies liveness, because every infinite behaviour must contain all the transitions, so from our previous proof of deadlock-freeness we deduce liveness.

Finally, in Figure 4, the decomposed view induced by the minimal P-semiflows is graphically presented. This view is even useful to derive an *implementation*. For instance, it shows that the net system in Figure 3 could be made up with two sequential processes (for *MACH 1* and *MACH 2*) and three semaphores: "object", "empty", and "R" — where "R" is a mutual exclusion semaphore. (Having a unique minimal T-semiflow, no T-decomposition exists in this case.)

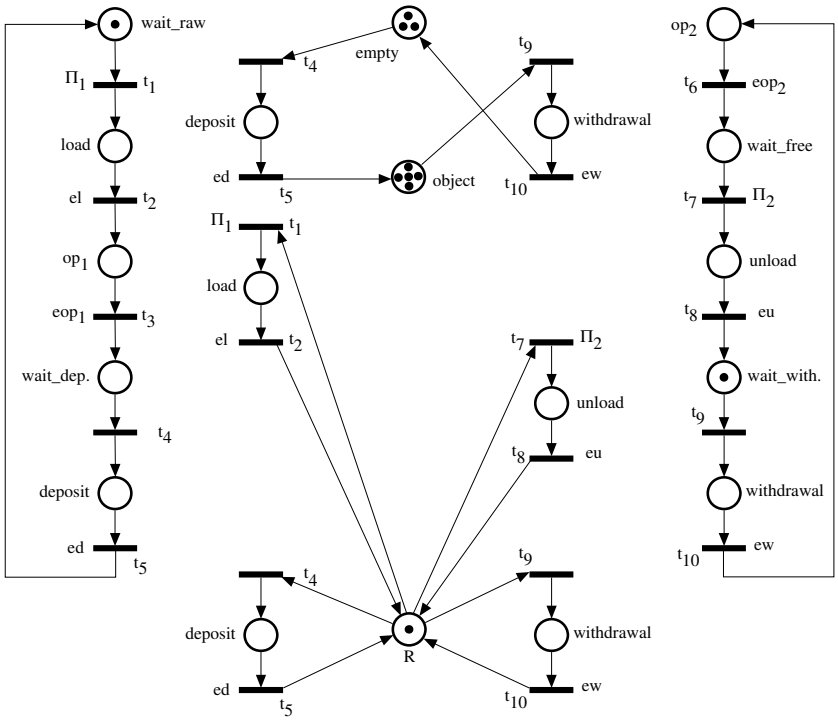

Fig. 4. A decomposed view of the net system in Figure 3.

3.4 Comparison of Linear Descriptions

We announced that the diverse linear descriptions have different degrees of accuracy. With respect to those presented so far, it can easily be established that:

Theorem 11. *Let $\mathcal{S}$ be a P/T system.*

$$\mathrm{RS}(\mathcal{S}) \subseteq \mathrm{LRS}^{\mathrm{SE}}(\mathcal{S}) \subseteq \mathrm{LRS}^{\mathrm{SE}\mathbb{R}}(\mathcal{S}) \subseteq \mathrm{LRS}^{\mathrm{Pf}}(\mathcal{S}) \subseteq \mathrm{LRS}^{\mathrm{Psf}}(\mathcal{S})$$

All the above inclusions may well be proper. We have already shown some examples for the first and second in Subsection 3.1. The P/T systems in Figure 5 give examples for the others.

The vector $p_1 + p_2 + p_5$ forms a basis of P-flows of the net of Figure 5 (a), so every marking $\alpha p_3 + p_5$, with $\alpha \geq 0$, is in $\mathrm{LRS}^{\mathrm{Pf}}(\mathcal{S})$. On the other hand, since $\mathbf{C}[p_4, T] > \mathbf{C}[p_5, T]$ and $\mathbf{m_0}[p_4] = \mathbf{m_0}[p_5]$, $\mathbf{m}[p_4] \geq \mathbf{m}[p_5]$ in every $\mathbf{m} \in \mathrm{LRS}^{\mathrm{SE}\mathbb{R}}(\mathcal{S})$, what is false in the case of $\alpha p_3 + p_5$.

For the net of Figure 5 (b), $p_1 + p_2$ is the only minimal P-semiflow, so every marking $p_1 + \alpha p_3 + \beta p_4$, with $\alpha, \beta \geq 0$, is in $\mathrm{LRS}^{\mathrm{Psf}}(\mathcal{S})$. Nevertheless, the dimension of the space of P-flows is two; the P-flow $p_3 - p_4$, that together with the minimal P-semiflow forms a basis, shows that markings $p_1 + \alpha p_3 + \beta p_4$ with $\alpha \neq \beta$ are not in $\mathrm{LRS}^{\mathrm{Pf}}(\mathcal{S})$.

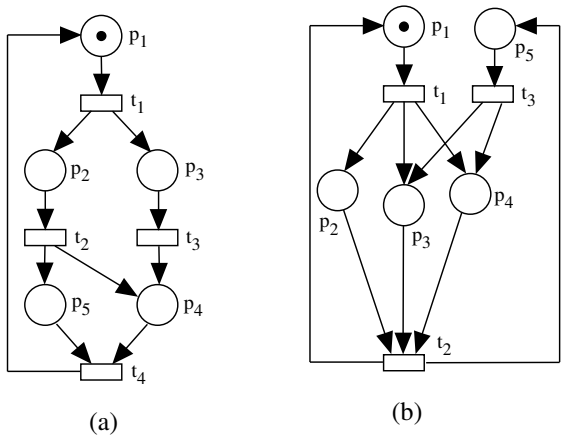

Fig. 5. Two P/T systems showing that $\mathrm{LRS}^{\mathrm{SE\mathbb{R}}} \overset{(a)}{\subset} \mathrm{LRS}^{\mathrm{Pf}} \overset{(b)}{\subset} \mathrm{LRS}^{\mathrm{Psf}}$.

In some cases, the above inclusions reduce to equalities. The interest of the following result is that it allows to use only the "more convenient" P-semiflows, instead of general P-flows, in the case of conservativeness, and that it guarantees that P-flows are as accurate as the state equation over the reals in the case of consistency:

Proposition 12. *Let $\mathcal{S}$ be a P/T system.*

1. *If $\mathcal{N}$ is conservative, then* $\mathrm{LRS}^{\mathrm{Pf}}(\mathcal{S}) = \mathrm{LRS}^{\mathrm{Psf}}(\mathcal{S})$.
2. *If $\mathcal{N}$ is consistent, then* $\mathrm{LRS}^{\mathrm{SE\mathbb{R}}}(\mathcal{S}) = \mathrm{LRS}^{\mathrm{Pf}}(\mathcal{S})$.

Proof. For Part 1, consider a $\mathbf{y} > \mathbf{0}$ such that $\mathbf{y} \cdot \mathbf{C} = \mathbf{0}$. A basis formed by P-semiflows only can easily be obtained from a basis of P-flows by adding a multiple of $\mathbf{y}$ to each P-flow.

For Part 2, if $\mathbf{m} \in \mathrm{LRS}^{\mathrm{Pf}}(\mathcal{S})$, then $\boldsymbol{\sigma}'$ (possibly $\boldsymbol{\sigma}' \not\geq \mathbf{0}$) exists such that $\mathbf{C} \cdot \boldsymbol{\sigma}' = \mathbf{m} - \mathbf{m_0}$. Using a $\mathbf{x} > \mathbf{0}$ such that $\mathbf{C} \cdot \mathbf{x} = \mathbf{0}$, a $\boldsymbol{\sigma} = \boldsymbol{\sigma}' + k\mathbf{x} \geq \mathbf{0}$ can be obtained so that the state equation is satisfied. □

In some net subclasses, stronger relations have been found. For instance, in live *state machines* (ordinary nets where every transition has one input and one output place; they are always conservative, and the only minimal P-semiflow is $\mathbf{1}$; liveness is equivalent to strong connectedness and non empty marking) $\mathrm{RS} = \mathrm{LRS}^{\mathrm{Psf}}$. In live *marked graphs* (ordinary nets where every place has one input and one output place; they are always consistent, and the only minimal T-semiflow is $\mathbf{1}$; liveness is equivalent to every directed circuit being marked) $\mathrm{RS} = \mathrm{LRS}^{\mathrm{SE\mathbb{R}}}$ [25,36,62]. In live consistent *weighted T-systems* (every place has one input and one output transition: marked graphs with weights; when consistent, they have only one minimal T-semiflow, $\mathbf{x}$, and then liveness is equivalent to $\mathbf{x}$ being fireable) $\mathrm{RS} = \mathrm{LRS}^{\mathrm{SE}}$ [88], although the inclusion $\mathrm{LRS}^{\mathrm{SE}} \subseteq \mathrm{LRS}^{\mathrm{SE\mathbb{R}}}$

can be proper because $\mathbf{C}$ is not *unimodular* as in marked graphs. Even in the case of weighted T-systems, these relations allow to prove properties more easily. For example, the reachability problem (i.e., is $\mathbf{m}$ reachable?) in a live consistent weighted T-system can be solved in polynomial time proceeding as indicated in Remark 2.

Other subclasses enjoy weaker but still useful properties. We give some examples. *Structurally persistent systems* are those with no distributor places, i.e., $|p^\bullet| \leq 1$ for every place p. (These systems, when strongly connected and consistent, are conservative and have only one minimal T-semiflow, $\mathbf{x}$. In such case, liveness and reversibility is equivalent to $\mathbf{x}$ being fireable.) In live, bounded, and reversible structurally persistent systems the reachable markings are the vectors in $\mathrm{LRS}^{\mathrm{SE}}$ from which $\mathbf{x}$ is fireable [90]. In live, bounded, and reversible (extended) *free choice systems* (ordinary nets where $\mathrm{CCS}(t) = \mathrm{EQS}(t)$ for every t; in the sequel, by free choice we always mean extended free choice) the reachable markings are the vectors in $\mathrm{LRS}^{\mathrm{Psf}}$ that mark every trap [30]. In live *equal conflict systems* (for every t, $\mathrm{CCS}(t) = \mathrm{EQS}(t)$: free choice with weights) every two solutions of the state equation have a common successor [92]. This implies existence of home states in live and bounded equal conflict systems, and the property that no spurious solution with respect to the state equation is a deadlock marking, what allows to analyse liveness using the state equation (see Subsection 4.6).

To conclude, an important consequence of the comparison is that the state equation, even over the reals, provides *more* information than the P-semiflows, even though the state equation leads to *more* efficient verification techniques (remember that the cardinality of the fundamental set of P-semiflows may grow exponentially with $|P|$). Therefore, in what follows, the state equation is used as the basic linear description of the state space.

3.5 Traps and Siphons

Besides the marking invariants that can be obtained from the net state equation, that were described in Subsection 3.2, other marking invariants can be formulated for net systems. In particular, traps and siphons, which are also structural objects, lead to new kinds of invariants. Differently from those associated with flows, possibly the invariant laws associated with traps and siphons do not hold in every marking, but once they become true they remain true for whichever future evolution (i.e., they are *stable predicates*): traps remain marked once they become marked, and siphons remain unmarked once they become unmarked. The same as in the case of semiflows, invariants, and components, we can distinguish three notions (idem for siphons):

- The trap (a set of places).
- A trap invariant (a stable predicate).
- The trap subnet (the P-subnet generated by the trap).

When a set of places is both a siphon and a trap, e.g., the support of a P-semiflow, the P-subnet that it generates is called a *siphon-trap component*. Notice, though,

that in the case of P-semiflows the P-invariant (token conservation law) is more informative than the siphon and trap invariants that are deduced from the corresponding siphon-trap component.

Traps and siphons have been extensively used for the structural analysis of (mainly ordinary) net systems (see Bibliographical Remarks). As a mere example, we state the following properties:

- In an ordinary deadlocked system, the subset of unmarked places is a siphon, because otherwise one of its input transitions would be enabled. Owing to this property of siphons, they are often called deadlocks, what is somehow misleading.
- Taking into account that traps remain marked, if every siphon of an ordinary net contains an initially marked trap, then the system is deadlock-free. In the case of *asymmetric choice* or *simple* systems (at most one of the input places to a join transition is a distributor place) the condition that every siphon contains a marked trap is sufficient for liveness, and in the case of free choice nets it is also necessary [41].
- If $\mathbf{m}$ is a home state of a live system, then every trap must be marked, because otherwise once the trap becomes marked — and it will eventually do by liveness — $\mathbf{m}$ cannot be reached any more. In the case of live and bounded free choice systems the converse is also true [7].

To our discussion, it is specially relevant to point out that some traps and siphons may contain information about the reachable markings that is *not* contained in the state equation. Therefore, they are potentially useful to improve the linear description of the state space provided by the state equation, as we shall discuss in Section 5. Consider the net in Figure 1. Clearly, $\Theta = \{p_1, p_2, p_5\}$ is an initially marked trap, so it must remain marked ($\mathbf{m}[p_1] + \mathbf{m}[p_2] + \mathbf{m}[p_5] \geq 1$), what allows to conclude that $2p_3$, $2p_4$, and $p_3 + p_4$ were spurious solutions to the state equation. Regarding siphons, for instance, in every spurious solution of the system of Figure 6 the (initially unmarked) siphon $\Sigma = \{p_1, p_2, p_3, p_6, p_7\}$ is marked, so taking the siphon invariant into account all the spurious solutions are proven non reachable. (Pragmatically, siphon invariants are less useful, since the existence of unmarked siphons in the initial marking is often considered undesirable: it implies that all its output transitions are dead.)

Next we describe a linear algebraic method to obtain a *generating family* of traps, i.e., a set such that every trap of the net is a union of traps in this set (observe that the union of two traps is a trap, they are *stable under union*). Notice that a generating family *includes* all the *minimal traps*, i.e., those which do not contain any other, but the set of minimal traps may not be generating. (The same approach can be applied to compute siphons, using that a siphon is a trap of the reverse net, or siphon-trap components.)

The method is based on the following property, which characterises traps in terms of non-negative solutions to a linear system of inequalities:

Theorem 13. *Let $\mathcal{N} = \langle P, T, \mathbf{Pre}, \mathbf{Post} \rangle$ be a P/T net. Define*

$$\mathcal{N}_\Theta = \langle P, T, \mathbf{Pre}, \mathbf{Post}_\Theta \rangle$$

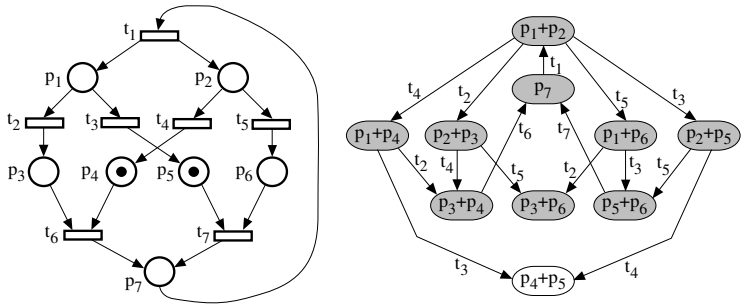

Fig. 6. A non live net system and its $\mathrm{LRG}^{\mathrm{SE}}$, where every spurious solution marks an initially unmarked siphon.

such that $\mathbf{Post}_\Theta[p,t] = 0$ *iff* $\mathbf{Post}[p,t] = 0$, *and* $\mathbf{Post}_\Theta[p,t] \geq \sum_{p' \in {}^\bullet t} \mathbf{Pre}[p',t]$ *otherwise.*

A set $\Theta \subseteq P$ *is a trap of* $\mathcal{N}$ *iff* $\mathbf{y} \geq \mathbf{0}$ *exists such that* $\|\mathbf{y}\| = \Theta$ *and* $\mathbf{y} \cdot \mathbf{C}_\Theta \geq \mathbf{0}$.

Proof. The inequality $\mathbf{y} \cdot \mathbf{C}_\Theta \geq \mathbf{0}$ means that in $\langle \mathcal{N}_\Theta, \mathbf{m_0} \rangle$ the weighted (according to $\mathbf{y}$) token content would never be decreased by the firing of transitions. If $\|\mathbf{y}\|$ was not a trap, then the firing of a transition in $\|\mathbf{y}\|^\bullet$ but not in ${}^\bullet\|\mathbf{y}\|$ would decrease the token content. On the other hand, by the definition of $\mathcal{N}_\Theta$, the firing of any transition puts at least as many tokens in each output place as it removes from the input places, so it is clear that when Θ is a trap the vector $\mathbf{1}_\Theta$ is a suitable $\mathbf{y}$. □

Steps 1 and 2 of Algorithm 8 can be used to compute the $\mathbf{y}$ vectors, hence the sought traps, by simply changing Step 2.2 so that only rows in which the i-th column of $\mathbf{A}$ is negative are eliminated. In order to provide a link to the well-known notion of P-semiflows, observe that the non-negative solutions to $\mathbf{y} \cdot \mathbf{C}_\Theta \geq \mathbf{0}$ are the non-negative solutions to

$$[\mathbf{y}\ \mathbf{z}] \cdot \begin{bmatrix} \mathbf{C}_\Theta \\ -\mathbf{I} \end{bmatrix} = \mathbf{0} \tag{17}$$

where $\mathbf{z}$ are *slack* variables. Therefore, the vectors $[\mathbf{y}\ \mathbf{z}]$ are P-semiflows of a net $\widetilde{\mathcal{N}_\Theta}$ that is obtained from $\mathcal{N}_\Theta$ by adding a source input place to every transition.

As an example, consider the net in Figure 1. A corresponding $\widetilde{\mathcal{N}_\Theta}$ is shown in Figure 7. The support of the P-semiflow of $\widetilde{\mathcal{N}_\Theta}$ that corresponds to the trap $\{p_1, p_2, p_5\}$ is shaded.

The above characterisation of traps allows to express some trap properties in linear algebraic terms. For instance:

– Initially marked traps remain marked: If $\mathbf{m}$ is reachable then the following system must be infeasible:

$$\mathbf{y} \cdot \mathbf{C}_\Theta \geq \mathbf{0} \wedge \mathbf{y} \geq \mathbf{0} \wedge \mathbf{y} \cdot \mathbf{m_0} > 0 \wedge \mathbf{y} \cdot \mathbf{m} = 0 \tag{18}$$

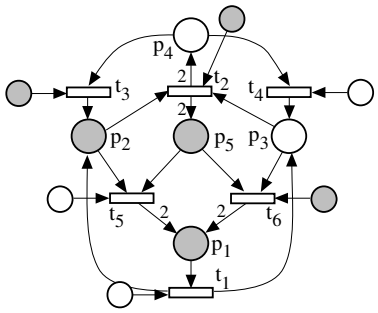

Fig. 7. A $\widetilde{\mathcal{N}_\Theta}$ corresponding to the net in Figure 1. The support of the P-semiflow associated with the trap $\{p_1, p_2, p_5\}$ is shaded.

where $\|\mathbf{y}\|$ is a trap (according to Theorem 13), that is initially marked ($\mathbf{y} \cdot \mathbf{m_0} > 0$) but not marked under $\mathbf{m}$ ($\mathbf{y} \cdot \mathbf{m} = 0$)

– In every home state of a live system every trap is marked: If $\mathbf{m}$ is a home state then the following system must be infeasible:

$$\mathbf{y} \cdot \mathbf{C}_\Theta \geq \mathbf{0} \wedge \mathbf{y} \geq \mathbf{0} \wedge \mathbf{y} \cdot \mathbf{m} = 0 \tag{19}$$

Notice that not every trap (or siphon) property can be expressed as a single system of linear inequalities. For instance, we cannot express that *when a trap becomes marked* it remains marked.

4 Analysis of Properties Using the State Equation

4.1 Overview

As we indicated in Subsection 3.1 the linear description — more precisely, relaxation — of the state space (and fireable sequences) provided by the state equation can be readily used for the analysis: the state equation is incorporated in the set of (linear) constraints of a linear (or integer) programming problem, where the property to analyse is either part of the linear constraints or appears in the cost function. We gave in (4) a straightforward application of this method to analyse reachability of a given marking. Owing to the possible presence of spurious solutions, in general this kind of analysis allows to *semidecide* only; in the case of reachability, for instance, the structural condition is only necessary. For other properties, only sufficient conditions are obtained.

Using this method we can analyse properties stated as existence or non existence of markings and/or firing sequences that satisfy some restrictions expressed in terms of linear inequalities. For existence we obtain necessary conditions, and sufficient for non-existence. Also we can compute (bounds for) the maximum of a linear function, or semidecide whether it exists. Some examples are submarking reachability, boundedness, repetitiveness, implicitness of a place, mutual exclusion, or deadlock-freeness:

- Is $\mathbf{m} \geq \mathbf{m}_x$ reachable? Existence of $\mathbf{m}$ such that $\mathbf{m} \geq \mathbf{m}_x$.
- Is p k-bounded? Non-existence of $\mathbf{m}$ such that $\mathbf{m}[p] > k$.
- Bound of p? Maximise $\mathbf{m}[p]$.
- Is t repetitive? Unbounded maximisation of $\boldsymbol{\sigma}[t]$.
- Is p implicit? Non-existence of $\mathbf{m}$ and t where $\mathbf{m}[p]$ is the only marking variable that prevents the firing of t, i.e., non-existence of $\mathbf{m}$ and t such that $\mathbf{m}[P - \{p\}] \geq \mathbf{Pre}[P - \{p\}, t]$ and $\mathbf{m}[p] < \mathbf{Pre}[p, t]$.
- Are the places in $\Pi \subseteq P$ in pairwise mutual exclusion? Non-existence of $\mathbf{m}$ that marks a pair of them.
- Deadlock-freeness? Non-existence of $\mathbf{m}$ where no transition is enabled.

In the following subsections we discuss the analysis of some of these properties in more detail, and we show how several classical results in structure theory of net systems are a rather direct consequence of this analysis, typically by making use of basic results on linear programming and duality theory.

It is quite apparent from the few examples of properties listed above that their expression can be more or less complicated. Formally, they are logic propositions where atoms are linear inequalities on marking and firing count variables. When these propositions are atomic or linked *conjunctively*, as it is the case of boundedness, repetitiveness, implicitness, or mutual exclusion between two places (Subsections 4.2 to 4.5) their inclusion in the set of linear constraints poses no particular problem.

When they are linked *disjunctively*, the verification requires checking a number of systems of linear inequalities for existence of solutions. The case of mutual exclusion is illustrative: For the mutual exclusion between two places, p and p', we check non-existence of $\mathbf{m}$ such that $\mathbf{m}[p] > 0$ and $\mathbf{m}[p'] > 0$; for the pairwise mutual exclusion between three places, p, p', and p'', we must check, in principle, non-existence of $\mathbf{m}$ such that $\mathbf{m}[p] > 0$ and $\mathbf{m}[p'] > 0$, non-existence of $\mathbf{m}$ such that $\mathbf{m}[p] > 0$ and $\mathbf{m}[p''] > 0$, and non-existence of $\mathbf{m}$ such that $\mathbf{m}[p'] > 0$ and $\mathbf{m}[p''] > 0$ (in general, for n places, non-existence of solution to $n(n-1)/2$ systems of linear inequalities). If the places were known to be safe, for instance, it would suffice to check just one system, because we could express the property as non-existence of $\mathbf{m}$ such that $\mathbf{m}[p] + \mathbf{m}[p'] + \mathbf{m}[p''] > 1$. Subsection 4.6 studies the case of deadlock-freeness, and illustrates how this problem with disjunctions can be palliated, even overcome in most practical cases.

Properties discussed so far are *safety properties* expressed as first order logic predicates where the domain is defined by only one type of quantifier (either existential or universal). One may conceive the analysis of other properties where the domain is defined by a combination of quantifiers of both types. However, since we can only obtain necessary conditions for existentially quantified predicates, and sufficient conditions for universally quantified ones, due to the possible presence of spurious solutions, as a result we would not even semidecide on such properties — leaving apart the case where one of the quantifications can be unrolled because the elements of the definition domain are finite and known a priori. Consider reversibility: $\mathbf{m_0}$ reachable from every reachable marking, i.e., *for all* $\mathbf{m}$ reachable from $\mathbf{m_0}$ *there exists* a sequence σ leading from $\mathbf{m}$ to $\mathbf{m_0}$.

If we used the state equation relaxation, we would analyse:

$$(\forall \mathbf{m})(\exists \boldsymbol{\sigma})(\mathbf{m} = \mathbf{m_0} + \mathbf{C} \cdot \boldsymbol{\sigma}' \geq \mathbf{0} \wedge \mathbf{m_0} = \mathbf{m} + \mathbf{C} \cdot \boldsymbol{\sigma} \geq \mathbf{0} \wedge \boldsymbol{\sigma}, \boldsymbol{\sigma}' \geq \mathbf{0}) \quad (20)$$

Validity of the above predicate is neither sufficient for reversibility, because the $\boldsymbol{\sigma}$'s may not be fireable, nor necessary, because an $\mathbf{m}$ invalidating it may be spurious. Liveness (of a transition) suffers from the very same problem. Of course, in net subclasses where these "difficult" properties are equivalent to others that can be analysed using the state equation method, the problem is solved. For instance, liveness of bounded strongly connected equal conflict systems is equivalent to deadlock-freeness [92], and live equal conflict systems do not have spurious deadlocks, so absence of deadlock markings which are solution to the state equation proves liveness (see Remark 35).

As a last comment, many of the properties that we consider here are particular — but specially relevant — cases of general *synchronic properties*. For instance, the *synchronic lead* of a subset of transitions with respect to another accounts for the maximum difference between the (possibly weighted) number of firings of the former and the latter. Boundedness of a place is a matter of boundedness of the synchronic lead of its input transitions with respect to its output transitions, and mutual exclusion between two safe places is equivalent to the synchronic lead of their input transitions with respect to their output transitions taking value one. The structural analysis of synchronic properties, which is presented in [83], is a more compact or abstract view of the analysis of many safety properties that we describe here.

4.2 Place Marking Bounds and Structural Boundedness

The marking bound of a place p in a net system $\mathcal{S}$ is defined as:

$$\mathbf{b}[p] = \max\{\mathbf{m}[p] \mid \ \mathbf{m} \in \mathrm{RS}(\mathcal{S})\} \quad (21)$$

When this bound is finite, the place is said to be bounded. Using the state equation, and writing $\mathbf{m}[p]$ as $\mathbf{1}_p \cdot \mathbf{m}$, we define the *structural bound* of a place p as:

$$\mathbf{sb}[p] = \max\{\mathbf{1}_p \cdot \mathbf{m} \mid \ \mathbf{m} - \mathbf{C} \cdot \boldsymbol{\sigma} = \mathbf{m_0} \wedge \mathbf{m}, \boldsymbol{\sigma} \geq \mathbf{0}\} \quad (22)$$

In principle, Equation (22) is an integer programming problem. If integrality constraints (on $\mathbf{m}$ and $\boldsymbol{\sigma}$) are disregarded then (22) is a linear programming problem, that can be solved in polynomial time.

According to Theorem 11, the marking for which the structural bound is reached could be spurious, so clearly, in general, we have $\mathbf{sb}[p] \geq \mathbf{b}[p]$. Therefore, if we were investigating the k-boundedness of p (i.e., is $\mathbf{m}[p] \leq k$?), an efficient *sufficient* condition is $\mathbf{sb}[p] \leq k$. We insist that the condition is only sufficient, and even it is possible that the structural bound for a bounded place does not exist (i.e., the programming problem is unbounded; we note at this point that, although the structural bound using integer programming can be

more accurate than using linear programming, when this particular linear programming problem is unbounded so it is the integer version — see Appendix). For instance, place p in the live and safe system of Figure 8 is bounded, but the linear programming problem (22) is unbounded.

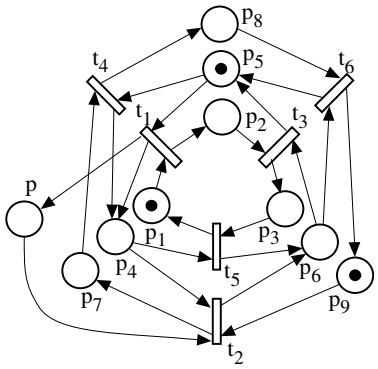

Fig. 8. Place p is bounded but not structurally bounded.

In the sequel we apply results from duality theory of linear programming. Although duality results are available for integer programming, we concentrate on linear programming, partly owing to the objective of re-encountering some classical results in net theory, and also to the pragmatic reason that the derived algorithms are more efficient, actually polynomial time. The dual linear programming problem of (22) is:

$$\mathbf{sb}[p]' = \min\{\mathbf{y} \cdot \mathbf{m_0} \mid \ \mathbf{y} \cdot \mathbf{C} \leq \mathbf{0} \wedge \mathbf{y} \geq \mathbf{1}_p\} \tag{23}$$

Since (22) has always a feasible solution ($\mathbf{m} = \mathbf{m_0}$, $\boldsymbol{\sigma} = \mathbf{0}$), both problems (22) and (23) are bounded iff a feasible solution for (23) exists, according to the duality and unboundedness theorems. In other words, if $\mathbf{y} \geq \mathbf{1}_p$ exists such that $\mathbf{y} \cdot \mathbf{C} \leq \mathbf{0}$, then p is structurally bounded, i.e., bounded for every initial marking. (Moreover, in such case $\mathbf{sb}[p] = \mathbf{sb}[p]'$.)

Remark 14. The linear programming problem (23) performs in polynomial time a *search* for the vector $\mathbf{y}$ that allows to obtain the best structural bound for p among all the P-semiflows and other vectors $\mathbf{y} \geq \mathbf{0}$ such that $\mathbf{y} \cdot \mathbf{C} \leq \mathbf{0}$. Clearly, it gives a more accurate bound than considering the P-semiflows only (besides being more efficient!) □

By the alternatives theorem, existence of $\mathbf{y} \geq \mathbf{1}_p$ such that $\mathbf{y} \cdot \mathbf{C} \leq \mathbf{0}$ is equivalent to non existence of $\mathbf{x} \geq \mathbf{0}$ such that $\mathbf{C} \cdot \mathbf{x} \geq \mathbf{1}_p$. Observe that, if such $\mathbf{x}$ exists, then we can find a sufficiently large initial marking $\mathbf{m_0}$ allowing to fire once and again a sequence with firing count vector $\mathbf{x}$, what shows that p is

unbounded. Therefore, non existence of $\mathbf{x} \geq \mathbf{0}$ such that $\mathbf{C} \cdot \mathbf{x} \geq \mathbf{1}_p$ is necessary for structural boundedness of p. In the example of Figure 8, $\mathbf{x} = t_1 + t_3 + t_5$ disproves structural boundedness of p. Notice that no sequence with such firing count vector is fireable under the marking shown in the figure; nevertheless, $t_1\, t_3\, t_5$ becomes fireable with $\mathbf{m_0} = p_1 + p_5 + p_6 + p_9$, and p becomes unbounded. (Interpreting boundedness as *stability* of the dynamic system, it can be said that non structurally bounded systems which are bounded are only *conditionally stable*, in the sense that an increment of the initial marking may lead to unstable behaviour.)

In summary:

Theorem 15. *Let $\mathcal{N}$ be a P/T net, and p one of its places. The following three statements are equivalent:*

1. *p is structurally bounded, i.e., bounded for every $\mathbf{m_0}$.*
2. *There exists $\mathbf{y} \geq \mathbf{1}_p$ such that $\mathbf{y} \cdot \mathbf{C} \leq \mathbf{0}$.*
3. *There does not exist $\mathbf{x} \geq \mathbf{0}$ such that $\mathbf{C} \cdot \mathbf{x} \geq \mathbf{1}_p$.*

The above statement, the same as several others in the sequel, contains two dual perspectives of a property, one *place-based* (item 2) and the other *transition-based* (item 3). We remark that the *duality* in this kind of net properties is rooted on *duality* theory in linear programming.

Applying the above characterisation of structural boundedness of p to every place of $\mathcal{N}$:

Corollary 16. *Let $\mathcal{N}$ be a P/T net. The following three statements are equivalent:*

1. *$\mathcal{N}$ is structurally bounded, i.e., every place is bounded for every $\mathbf{m_0}$.*
2. *There exists $\mathbf{y} > \mathbf{0}$ such that $\mathbf{y} \cdot \mathbf{C} \leq \mathbf{0}$.*
3. *There does not exist $\mathbf{x} \geq \mathbf{0}$ such that $\mathbf{C} \cdot \mathbf{x} \gneq \mathbf{0}$.*

Remark 17. It is well known that nets with *inhibitor arcs* — which is a widely used extension of P/T systems — can be simulated with plain P/T systems using the *complementary place* construction in the case that the inhibiting places are bounded. This is particularly true for structurally bounded places, with the additional advantage that the transformation can be done for whichever initial marking. In the sequel, it goes without saying that all the results can be extended to nets with inhibitor arcs in the case of structural boundedness. □

4.3 Transition Fireability Bounds and Structural Repetitiveness

The fireability or repetitiveness bound of a transition t in a net system $\mathcal{S}$ is defined as:

$$\mathbf{r}[t] = \max\{\boldsymbol{\sigma}[t] \mid \ \sigma \in \mathrm{L}(\mathcal{S})\} \tag{24}$$

If this bound is zero, then the transition is *dead* (or *0-live*). If the bound takes a finite positive value, the transition can only be fired a finite number of

times (it is *1-live*). When the bound does not exist, transition t is said to be *repetitive* (or *2-live*), and this is necessary for liveness of t.

Using the state equation, we define the *structural repetitiveness bound* of a transition t as:

$$\mathbf{sr}[t] = \max\{\mathbf{1}_t \cdot \boldsymbol{\sigma} \mid \mathbf{m} - \mathbf{C} \cdot \boldsymbol{\sigma} = \mathbf{m_0} \wedge \mathbf{m}, \boldsymbol{\sigma} \geq \mathbf{0}\} \tag{25}$$

Clearly, if the transition is repetitive, then (25) is unbounded, since $\mathbf{sr}[t] \geq \mathbf{r}[t]$. (The converse is not true because the $\boldsymbol{\sigma}$'s that make the problem unbounded may not correspond to actually fireable sequences: all the transitions in the nets of Figure 2 are structurally repetitive and none of them is repetitive.)

The dual linear programming problem of (25) is:

$$\mathbf{sr}[t]' = \min\{\mathbf{y} \cdot \mathbf{m_0} \mid \mathbf{y} \cdot \mathbf{C} \leq -\mathbf{1}_t \wedge \mathbf{y} \geq \mathbf{0}\} \tag{26}$$

Since (25) has always a feasible solution ($\mathbf{m} = \mathbf{m_0}$, $\boldsymbol{\sigma} = \mathbf{0}$), it is unbounded iff (26) is infeasible, according to the duality and unboundedness theorems. In other words, if some $\mathbf{y} \geq \mathbf{0}$ exists such that $\mathbf{y} \cdot \mathbf{C} \leq -\mathbf{1}_t$, then t is not structurally repetitive, i.e., not repetitive for any initial marking.

By the alternatives theorem, non existence of $\mathbf{y} \geq \mathbf{0}$ such that $\mathbf{y} \cdot \mathbf{C} \leq -\mathbf{1}_t$ is equivalent to existence of $\mathbf{x} \geq \mathbf{1}_t$ such that $\mathbf{C} \cdot \mathbf{x} \geq \mathbf{0}$. Observe that, if such $\mathbf{x}$ exists, then we can find a sufficiently large initial marking $\mathbf{m_0}$ allowing to fire once and again a sequence with firing count vector $\mathbf{x}$, what shows that t is repetitive. Therefore, existence of $\mathbf{x} \geq \mathbf{1}_t$ such that $\mathbf{C} \cdot \mathbf{x} \geq \mathbf{0}$ is sufficient for structural repetitiveness of t. In summary:

Theorem 18. *Let $\mathcal{N}$ be a P/T net, and t one of its transitions. The following three statements are equivalent:*

1. *t is structurally repetitive, i.e., repetitive for some $\mathbf{m_0}$.*
2. *There does not exist $\mathbf{y} \geq \mathbf{0}$ such that $\mathbf{y} \cdot \mathbf{C} \leq -\mathbf{1}_t$.*
3. *There exists $\mathbf{x} \geq \mathbf{1}_t$ such that $\mathbf{C} \cdot \mathbf{x} \geq \mathbf{0}$.*

Applying the above characterisation of structural repetitiveness of t to every transition of $\mathcal{N}$:

Corollary 19. *Let $\mathcal{N}$ be a P/T net. The following three statements are equivalent:*

1. *$\mathcal{N}$ is structurally repetitive, i.e., every transition is repetitive for some $\mathbf{m_0}$.*
2. *There does not exist $\mathbf{y} \geq \mathbf{0}$ such that $\mathbf{y} \cdot \mathbf{C} \lneq \mathbf{0}$.*
3. *There exists $\mathbf{x} > \mathbf{0}$ such that $\mathbf{C} \cdot \mathbf{x} \geq \mathbf{0}$.*

Comparing Corollaries 16 and 19 (or Theorems 15 and 18) it is quite apparent that structural boundedness and structural repetitiveness are dual notions: a net with incidence matrix $\mathbf{C}$ is structurally repetitive iff the reverse dual, with incidence matrix $-\mathbf{C}^{\perp}$, is structurally bounded. Both properties together are equivalent to conservativeness and consistency: Let $\mathbf{x} > \mathbf{0}$ and $\mathbf{y} > \mathbf{0}$ be the

vectors such that $\mathbf{C}\cdot\mathbf{x} \geq \mathbf{0}$ and $\mathbf{y}\cdot\mathbf{C} \leq \mathbf{0}$, respectively. By $\mathbf{C}\cdot\mathbf{x} \geq \mathbf{0}$, $\mathbf{y}\cdot(\mathbf{C}\cdot\mathbf{x}) \geq \mathbf{0}$, while by $\mathbf{y}\cdot\mathbf{C} \leq \mathbf{0}$, $(\mathbf{y}\cdot\mathbf{C})\cdot\mathbf{x} \leq \mathbf{0}$. Therefore $\mathbf{y}\cdot\mathbf{C}\cdot\mathbf{x} = \mathbf{0}$, hence $\mathbf{C}\cdot\mathbf{x} = \mathbf{0}$ and $\mathbf{y}\cdot\mathbf{C} = \mathbf{0}$. Since (structural) repetitiveness is necessary for (structural) liveness, i.e., liveness *for some* initial marking, see Proposition 10, we can state:

Theorem 20. *Let $\mathcal{N}$ be a P/T net. If $\mathcal{N}$ is structurally live and structurally bounded, then (equivalently):*

1. *$\mathcal{N}$ is structurally repetitive and structurally bounded.*
2. *$\mathcal{N}$ is conservative and consistent.*

Taking into account Proposition 12:

Corollary 21. *Let $\mathcal{N}$ be a P/T net. If $\mathcal{N}$ is structurally live and structurally bounded, then for every $\mathbf{m_0}$:*

$$\mathrm{LRS}^{\mathrm{SE}\mathbb{R}}(\mathcal{N},\mathbf{m_0}) = \mathrm{LRS}^{\mathrm{Pf}}(\mathcal{N},\mathbf{m_0}) = \mathrm{LRS}^{\mathrm{Psf}}(\mathcal{N},\mathbf{m_0})$$

4.4 Implicit Places and Structurally Implicit Places

In general, places impose constraints on the firing of their output transitions. When they never do, they could be removed without affecting the behaviour of the rest of the system. (However, even being redundant, they might still be useful — one such application is developed in Section 5.)

A place whose removal does not affect the behaviour of the system is called an *implicit place*. Here, by behaviour we understand the *interleaving semantics*, i.e., the sequential observations or language, although the notion of implicit place can be directly extended to cope with a *step semantics* (see Remark 27).

Definition 22. Let $\mathcal{S} = \langle P \cup \{p\}, T, \mathbf{Pre}, \mathbf{Post}, \mathbf{m_0}\rangle$ be a P/T system. The place p is *implicit* iff $\mathrm{L}(\mathcal{S}) = \mathrm{L}(\langle P, T, \mathbf{Pre}[P,T], \mathbf{Post}[P,T], \mathbf{m_0}[P]\rangle)$.

In other words, p is never the unique place that prevents the firing of a transition, i.e., $\mathbf{m} \geq \mathbf{Pre}[P,t] \Rightarrow \mathbf{m}[p] \geq \mathbf{Pre}[p,t]$ for all $t \in p^\bullet$, so it produces fictitious synchronisations in its output transitions. For instance, in Figure 9 (a), t_4 seems to represent a synchronisation, but whenever p_3 is marked so it is p, so p is implicit. It is worth noticing at this point that, in general, $\mathbf{m}[p]$ may not be computable from the marking of other places. For instance, the marking of p in Figure 9 (a) cannot be deduced from the marking of the other places, it depends also on the number of occurrences of t_1. When the marking of an implicit place can be computed from the marking of other places, we say that it is *marking implicit*. In Figure 9 (b), both p_1 and p_4 are marking implicit: $\mathbf{m}[p_1] = \mathbf{m}[p]+\mathbf{m}[p_3]$ and $\mathbf{m}[p_4] = \mathbf{m}[p]+\mathbf{m}[p_2]$. Therefore, they can be removed without affecting the behaviour of the system. The net in Figure 9 (b) shows also that, naturally, being implicit or not is sometimes a matter of the initial marking. With $p_1 + p_4 + 2p$ as initial marking, p is marking implicit instead of p_1 and p_4, and $\mathbf{m}[p] = \mathbf{m}[p_1] + \mathbf{m}[p_4]$: the behaviour of the system is that of

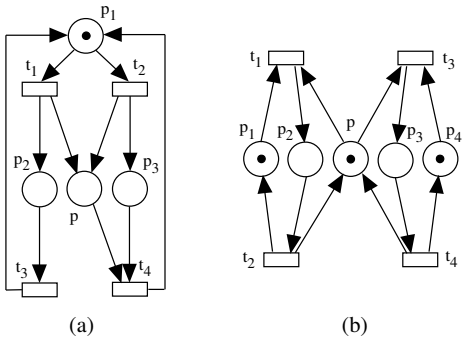

Fig. 9. Implicit and marking implicit places. Place p is implicit but not marking implicit in (a). Places p_1 and p_4 are marking implicit in (b). If we add a token to p in (b), then it becomes marking implicit instead.

two cyclic processes, $(t_1\, t_2)^*$ and $(t_3\, t_4)^*$ that evolve in parallel independently, in spite of the apparent synchronisation introduced by p.

Notice that once the initial marking of the rest of the system is fixed, implicitness of a place is *monotonic* with respect to its initial marking (i.e., if the place is implicit for some initial marking, it is also implicit with a greater one).

To decide whether a given p is implicit, in principle, we should check that no reachable marking exists such that for some t we have $\mathbf{m}[P] \geq \mathbf{Pre}[P,t]$ and $\mathbf{m}[p] < \mathbf{Pre}[p,t]$. Of course, it is necessary that p is not the only input place of its output transitions. This syntactical check rapidly tells that, for instance, p_2 and p_3 cannot be implicit in the net of Figure 9 (b).

The condition that no reachable marking exists such that for some t (actually, $t \in p^\bullet$) we have $\mathbf{m}[P] \geq \mathbf{Pre}[P,t]$ and $\mathbf{m}[p] < \mathbf{Pre}[p,t]$ can be expressed conveniently using a "transition selector" $\mathbf{s}$, i.e., the characteristic vector of one transition ($\mathbf{s} \geq \mathbf{0}$ and $\mathbf{1} \cdot \mathbf{s} = 1$): Place p is implicit in $\mathcal{S} = \langle P \cup \{p\}, T, \mathbf{Pre}, \mathbf{Post}, \mathbf{m_0} \rangle$ iff there are no $\mathbf{m}$ and $\mathbf{s}$ which are solution to:

$$\begin{aligned} \mathbf{m} &\in \mathrm{RS}(\mathcal{S}) \\ \mathbf{m}[P] - \mathbf{Pre}[P, p^\bullet] \cdot \mathbf{s} &\geq \mathbf{0} \\ \mathbf{m}[p] - \mathbf{Pre}[p, p^\bullet] \cdot \mathbf{s} &< 0 \\ \mathbf{1} \cdot \mathbf{s} &= 1 \\ \mathbf{s} &\geq \mathbf{0} \end{aligned} \tag{27}$$

where $\mathbf{m}[P] \geq \mathbf{Pre}[P, p^\bullet] \cdot \mathbf{s}$ states that $\|\mathbf{s}\|$ is enabled by P, and $\mathbf{m}[p] < \mathbf{Pre}[p, p^\bullet] \cdot \mathbf{s}$ states that it is not enabled by p.

Replacing for the first condition the state equation, as usual, we obtain a structural *sufficient* condition for p implicit, in terms of non existence of solution $\mathbf{m}$, $\boldsymbol{\sigma}$, and $\mathbf{s}$ to:

$$\begin{aligned} \mathbf{m} - \mathbf{C} \cdot \boldsymbol{\sigma} &= \mathbf{m_0} \\ \mathbf{m}[P] - \mathbf{Pre}[P, p^\bullet] \cdot \mathbf{s} &\geq \mathbf{0} \\ \mathbf{m}[p] - \mathbf{Pre}[p, p^\bullet] \cdot \mathbf{s} &< 0 \\ \mathbf{1} \cdot \mathbf{s} &= 1 \\ \mathbf{m}, \boldsymbol{\sigma}, \mathbf{s} &\geq \mathbf{0} \end{aligned} \tag{28}$$

Since $\mathbf{m}[p] = \mathbf{m_0}[p] + \mathbf{C}[p, T] \cdot \boldsymbol{\sigma}$, the above is equivalent to $\mathbf{m_0}[p]$ being greater than or equal to the optimal value of the following linear programming problem:

$$\begin{aligned} \max\{\mathbf{Pre}[p, p^\bullet] \cdot \mathbf{s} - \mathbf{C}[p, T] \cdot \boldsymbol{\sigma} \mid \quad & \mathbf{m}[P] - \mathbf{C}[P, T] \cdot \boldsymbol{\sigma} = \mathbf{m_0}[P] \\ & \mathbf{m}[P] - \mathbf{Pre}[P, p^\bullet] \cdot \mathbf{s} \geq \mathbf{0} \\ & \mathbf{1} \cdot \mathbf{s} = 1 \\ & \mathbf{m}, \boldsymbol{\sigma}, \mathbf{s} \geq \mathbf{0}\} \end{aligned} \tag{29}$$

The initial marking does not appear in the constraints of the dual problem of (29), which is:

$$\begin{aligned} \min\{\mathbf{y} \cdot \mathbf{m_0} + \mu \mid \quad & \mathbf{y} \cdot \mathbf{C}[P, T] \leq \mathbf{C}[p, T] \\ & \mathbf{j} \cdot \mathbf{Pre}[P, p^\bullet] + \mu \mathbf{1} \geq \mathbf{Pre}[p, p^\bullet] \\ & \mathbf{y} \geq \mathbf{j} \geq \mathbf{0}\} \end{aligned} \tag{30}$$

Considering $\mathbf{j} = \mathbf{y}$ does not affect the solution, leading to the following sufficient condition for p implicit:

Theorem 23. *Let $\mathcal{S} = \langle P \cup \{p\}, T, \mathbf{Pre}, \mathbf{Post}, \mathbf{m_0} \rangle$ be a P/T system. If $\mathbf{m_0}[p]$ is greater than or equal to the optimal value of the following linear programming problem:*

$$\begin{aligned} \min\{\mathbf{y} \cdot \mathbf{m_0}[P] + \mu \mid \quad & \mathbf{y} \cdot \mathbf{C}[P, T] \leq \mathbf{C}[p, T] \\ & \mathbf{y} \cdot \mathbf{Pre}[P, p^\bullet] + \mu \mathbf{1} \geq \mathbf{Pre}[p, p^\bullet] \\ & \mathbf{y} \geq \mathbf{0}\} \end{aligned} \tag{31}$$

then p is implicit.

Let us consider the places which can be made implicit *for every* initial marking of the rest of the system (i.e., we are abstracting from the initial marking, as for other structural versions of properties). We call such places *structurtally implicit*:

Definition 24. Let $\mathcal{N} = \langle P \cup \{p\}, T, \mathbf{Pre}, \mathbf{Post} \rangle$ be a P/T net. The place p is *structurally implicit* iff for every $\mathbf{m_0}[P]$, a $\mathbf{m_0}[p]$ exists such that p is implicit in $\mathcal{S} = \langle P \cup \{p\}, T, \mathbf{Pre}, \mathbf{Post}, \mathbf{m_0} \rangle$.

The linear programming problem in Theorem 23 is feasible iff a $\mathbf{y} \geq \mathbf{0}$ exists such that $\mathbf{C}[p, T] \geq \mathbf{y} \cdot \mathbf{C}[P, T]$, because the constraint containing μ can always be satisfied taking a sufficiently large μ. When a place fulfills such condition (or

its alternative formulation), it becomes implicit with a sufficiently large initial marking, at least with that given by the linear programming problem, so it is structurally implicit. (Notice that it may become implicit also with a smaller marking in some cases, see later Remark 39.)

On the other hand, consider a net $\mathcal{N}$ and a place p for which no $\mathbf{y} \geq \mathbf{0}$ exists such that $\mathbf{C}[p,T] \geq \mathbf{y} \cdot \mathbf{C}[P,T]$. By the alternatives theorem, a $\mathbf{x} \geq \mathbf{0}$ exists such that $\mathbf{C}[P,T] \cdot \mathbf{x} \geq \mathbf{0}$ and $\mathbf{C}[p,T] \cdot \mathbf{x} < 0$. With a sufficiently large initial marking $\mathbf{m_0}[P]$, a sequence with firing count vector $\mathbf{x}$ can be fired repeatedly in $\langle \mathcal{N}, \mathbf{m_0}[P] \rangle$, and this would empty p for whichever $\mathbf{m_0}[p]$, so p is not structurally implicit.

In summary, a characterisation of structurally implicit places is given by the following result:

Theorem 25. *Let $\mathcal{N} = \langle P \cup \{p\}, T, \mathbf{Pre}, \mathbf{Post} \rangle$. The place p is structurally implicit iff (equivalently):*

1. *A $\mathbf{y} \geq \mathbf{0}$ exists such that $\mathbf{C}[p,T] \geq \mathbf{y} \cdot \mathbf{C}[P,T]$*
2. *No $\mathbf{x} \geq \mathbf{0}$ exists such that $\mathbf{C}[P,T] \cdot \mathbf{x} \geq \mathbf{0}$ and $\mathbf{C}[p,T] \cdot \mathbf{x} < \mathbf{0}$*

Remark 26. Similarly as structural boundedness and structural repetitiveness are dual notions, the dual notion of a structural implicit place can be defined. The resulting objects are called *structural bypass transitions*, which have also a behavioural interpretation (see [73]). □

When $\mathbf{C}[p,T] = \mathbf{y} \cdot \mathbf{C}[P,T]$ in Theorem 25.1, the place p is *marking structurally implicit* and its marking can be computed from the marking of other places:

$$\begin{aligned} \mathbf{m}[p] &= \mathbf{m_0}[p] + \mathbf{C}[p,T] \cdot \boldsymbol{\sigma} \\ &= \mathbf{m_0}[p] + \mathbf{y} \cdot \mathbf{C}[P,T] \cdot \boldsymbol{\sigma} \\ &= \mathbf{m_0}[p] + \mathbf{y} \cdot (\mathbf{m}[P] - \mathbf{m_0}[P]) \end{aligned} \tag{32}$$

As an example of (marking) structurally implicit place, consider p with $\mathbf{C}[p,T] = t_3 - t_5$ added to the net system in Figure 1. The optimal solution to the linear programming problem in Theorem 23 is zero, and it is obtained for $\mathbf{y} = p_1 + p_2 + p_5$ and $\mu = -1$. Therefore, p is (marking) structurally implicit ($\mathbf{C}[p,T] = \mathbf{y} \cdot \mathbf{C}[P,T]$), and with $\mathbf{m_0}[p] = 0$ it is (marking) implicit — as can be easily checked. It is interesting to observe in this case that, once p is added, the places p_1, p_3, p_4, and p are the support of a *new* P-semiflow, which induces the new marking invariant $\mathbf{m}[p_1] + \mathbf{m}[p_3] + \mathbf{m}[p_4] + \mathbf{m}[p] = 1$. This invariant reveals, in particular, that p_3 and p_4 are safe and in mutual exclusion, what could not be proven using the state equation of the original net, where $2p_3$, $2p_4$, and $p_3 + p_4$ were spurious marking. The addition of an implicit place, which *does not affect the behaviour*, may be useful to allow proving properties by structural techniques!

An example of implicit place which is not structurally implicit, effectively showing that the structural condition is only sufficient, is p in Figure 8. (In

practice, though, specially under liveness and boundedness, implicit places are most often structurally implicit.)

Remark 27. A place is *concurrent implicit* when it preserves not only the transition firing sequences but also the *step* firing sequences. Any concurrent implicit place is of course (sequential) implicit, and (sequential) implicit places without self-loops are also concurrent implicit — notice that they differ only when a transition is enabled *several times* by the marking of the other places while it is merely enabled by the implicit place.

The sufficient condition for a place p to be implicit given by Theorem 23 can be generalised to cope with concurrent implicit places by substituting the cost function in (31) by $\mathbf{y} \cdot \mathbf{m_0}[P] + \alpha \cdot \mu$, where α is a positive integer, not necessarily one. A sensible choice for α must be greater than or equal to the maximum *enabling bound* of the transitions in $p^\bullet$, where the enabling bound of a transition is the maximum number of times that it can occur in a step. A structural enabling bound for a transition t, possibly greater than the actual enabling bound, can be computed as:

$$\mathbf{seb}[t] = \max\{k \mid \ \mathbf{m} \geq k\mathbf{Pre}[P,t] \wedge \mathbf{m} - \mathbf{C} \cdot \boldsymbol{\sigma} = \mathbf{m_0} \wedge \mathbf{m}, \boldsymbol{\sigma} \geq \mathbf{0}\} \tag{33}$$

□

4.5 Mutual Exclusion and Concurrency Relations

We consider now the verification of mutual exclusion between two places, which is the basic property to analyse in order to investigate pairwise mutual exclusions, as was discussed in Subsection 4.1.

The problem of pairwise concurrency of transitions is closely related. Even it can be transformed into non mutual exclusion between places by splitting the analysed transitions into paths transition → place → transition. More directly, for transitions that are in pairwise concurrency relation we can find, for each pair t and t', a marking $\mathbf{m}$ such that $\mathbf{m} \geq \mathbf{Pre}[P,t] + \mathbf{Pre}[P,t']$.

We concentrate on the mutual exclusion of two places, p and p':

Theorem 28. *Let $\mathcal{S}$ be a P/T system.*

If there is no solution to

$$\mathbf{m} - \mathbf{C} \cdot \boldsymbol{\sigma} = \mathbf{m_0} \wedge \mathbf{m} \geq \mathbf{1}_{\{p,p'\}} \wedge \boldsymbol{\sigma} \geq \mathbf{0} \tag{34}$$

or (equivalently) there exists solution to

$$\mathbf{y} \cdot \mathbf{C} \leq \mathbf{0} \wedge \mathbf{y} \cdot \mathbf{1}_{\{p,p'\}} - \mathbf{y} \cdot \mathbf{m_0} > 0 \wedge \mathbf{y} \geq \mathbf{0} \tag{35}$$

then p and p' are in mutual exclusion.

Remark 29. As usual, the alternative formulation given by (35) is only equivalent when considering (34) over the reals. For instance, consider the net on the right and top in Figure 2 initially marked with $2p_2$ instead of $p_1 + p_2$ (which now becomes a spurious solution). While (35) does not allow to prove mutual exclusion of p_1 and p_2, (34) does if it is interpreted over the integers. □

The interpretation of (35) is the following: From $\mathbf{y} \cdot \mathbf{m_0} < \mathbf{y} \cdot \mathbf{1}_{\{p,p'\}}$ and $\mathbf{y} \cdot \mathbf{C} \leq \mathbf{0}$ it follows that in every reachable marking $\mathbf{y} \cdot \mathbf{m} < \mathbf{y} \cdot \mathbf{1}_{\{p,p'\}}$, so $\mathbf{m} \geq \mathbf{1}_{\{p,p'\}}$ is impossible, what proves mutual exclusion. (In structurally repetitive nets, the condition $\mathbf{y} \cdot \mathbf{C} \leq \mathbf{0}$ can be replaced by $\mathbf{y} \cdot \mathbf{C} = \mathbf{0}$, because, according to Proposition 19, it is never the case that $\mathbf{y} \cdot \mathbf{C} \lneq \mathbf{0}$.)

The dual formulation of the mutual exclusion problem given by (35) proceeds in the same way as the dual formulation of the boundedness problem, efficiently searching for a suitable vector $\mathbf{y}$ to prove the property (see Remark 14). An additional advantage of this formulation compared to (34) is that the obtained $\mathbf{y}$ provides an *explanation* of the property and may be useful to derive other mutual exclusion relations by the way. As an example, consider again the net system in Figure 1 after adding the place p with $\mathbf{C}[p,T] = t_3 - t_5$, and suppose that we are trying to prove the mutual exclusion between p_1 and p_4 applying (35). A solution $\mathbf{y} = p_1 + p_2 + p_4 + p$ is obtained. The corresponding marking invariant, $\mathbf{m}[p_1] + \mathbf{m}[p_3] + \mathbf{m}[p_4] + \mathbf{m}[p] = 1$, proves not only mutual exclusion of p_1 and p_4, but also mutual exclusion of any other pair, i.e., it proves at once the pairwise mutual exclusion of p_1, p_3, p_4, and p.

4.6 Deadlock-freeness and Termination Properties

Proving that a concurrent system cannot reach a deadlock condition (i.e., no activity because every process is waiting for some other to continue) may be difficult when the size of the system becomes large and its structure intricate due to complex and perhaps subtle interactions. For instance, (flexible) manufacturing systems are indeed a characteristic domain where the study of deadlocks is specially relevant [3,34,94,95]. The tasking behaviour of Ada programs is another example [64].

We apply here the state equation method to analyse the absence of deadlocks in P/T models. The approach is conceptually simple but, since the condition for a marking to be a deadlock is relatively complex, it still requires a considerable computational effort. We present here several techniques to improve the performance, often reducing the problem to checking a single system of linear inequalities for existence of solutions.

In net terms, a deadlock corresponds to a marking from which no transition is fireable. That t is disabled at $\mathbf{m}$ can be expressed:

$$\bigvee_{p \in {}^\bullet t} \mathbf{m}[p] < \mathbf{Pre}[p,t] \tag{36}$$

Clearly, every reachable deadlock is a solution to the state equation where every transition is disabled, what leads to the following basic general sufficient condition for deadlock-freeness:

Theorem 30. *Let $\mathcal{S}$ be a P/T system.*

If there is no (integer) solution to

$$\begin{array}{c} \mathbf{m} - \mathbf{C} \cdot \boldsymbol{\sigma} = \mathbf{m_0} \\ \mathbf{m}, \boldsymbol{\sigma} \geq \mathbf{0} \\ \bigvee_{p \in {}^\bullet t} \mathbf{m}[p] < \mathbf{Pre}[p,t] \quad \forall t \in T \end{array} \tag{37}$$

then $\mathcal{S}$ is deadlock-free.

In general, the disabledness conditions are not linear, because they consist of linear inequalities linked disjunctively. Anyway we can rewrite (37) as a set of systems of linear inequalities/equations, i.e., as a logic sum of products, applying the distributive property to the actual product of sums: If for every mapping $\alpha : T \to P$ that assigns to each transition one of its input places there is no solution to:

$$\begin{array}{c} \mathbf{m} - \mathbf{C} \cdot \boldsymbol{\sigma} = \mathbf{m_0} \\ \mathbf{m}, \boldsymbol{\sigma} \geq \mathbf{0} \\ \mathbf{m}[\alpha(t)] < \mathbf{Pre}[\alpha(t), t] \quad \forall t \in T \end{array} \tag{38}$$

then $\mathcal{S}$ is deadlock-free.

The problem now is that we have to check (38) for *every* mapping α, so we have to check $\prod_{t \in T} |{}^\bullet t|$ linear systems. In general this number might be large if there are many synchronisations (join transitions). We aim at reducing this number as much as we can, preserving the decision power, that is, the set of *integer* solutions to (37).

Firstly, we can obviously remove from (37) a disabledness condition if there is another disabledness condition that is "weaker". In other words, if $\mathbf{Pre}[P,t] \leq \mathbf{Pre}[P,t']$ for some t and t', the disabledness condition for t' can be removed without affecting the set of solutions to (37), even over the reals.

It is also quite obvious that we can remove from (37) the disabledness conditions of transitions that are known to be *dead* (or *0-live*, they do not appear in any sequence from $\mathbf{m_0}$) because whenever all the other transitions are disabled, we are certainly in a deadlock situation. This does not affect either the set of real solutions to (37).

Clearly, a sufficient condition for t dead is non-existence of solution to:

$$\mathbf{m} - \mathbf{C} \cdot \boldsymbol{\sigma} = \mathbf{m_0} \wedge \boldsymbol{\sigma} \geq \mathbf{0} \wedge \mathbf{m} \geq \mathbf{Pre}[P,t] \tag{39}$$

Considering (39) over the reals, and applying the alternatives theorem, we obtain an alternative sufficient condition for t dead, in terms of *existence* of solution to:

$$\mathbf{y} \cdot \mathbf{C} \leq \mathbf{0} \wedge \mathbf{y} \geq \mathbf{0} \wedge \mathbf{y} \cdot (\mathbf{Pre}[P,t] - \mathbf{m_0}) > \mathbf{0} \tag{40}$$

If there were a solution $\mathbf{y}$, it would induce the marking invariant $\mathbf{y} \cdot \mathbf{m} \leq \mathbf{y} \cdot \mathbf{m_0}$. If some $\mathbf{m}$ enabled t, then $\mathbf{m} \geq \mathbf{Pre}[P,t]$, so $\mathbf{y} \cdot \mathbf{m} \geq \mathbf{y} \cdot \mathbf{Pre}[P,t] > \mathbf{y} \cdot \mathbf{m_0}$, contradiction.

A simpler (and weaker) sufficient condition for t dead is that there exists $p \in {}^\bullet t$ such that $\mathbf{sb}[p] < \mathbf{Pre}[p,t]$. For the application of the following results, the knowledge of $\mathbf{sb}$ is assumed, so this weaker condition can be applied with no extra computational effort.

Remark 31. Termination properties can often be rephrased in terms of fireability (or *quasi-liveness*) of an artificial transition that removes the tokens from the desired final state and restores the initial marking (e.g., see [10]). Proving that such transition is dead, e.g., using (39) or (40), is sufficient — of course not necessary — to disprove termination. □

Although disregarding some transitions applying the above arguments may be helpful, typically the more drastic reduction in the number of systems to check is produced by the results that we present in the sequel. They provide rules to rewrite the disabledness condition of a transition in a less complex way while preserving the set of *integer* solutions to (37). (Notice that if the systems were finally checked disregarding the integrality of variables, these rules might diminish the decision power.)

Proposition 32. *Let t be a transition such that for every $p \in \pi \subseteq {}^\bullet t$ the following holds: $\mathbf{sb}[p] \leq \mathbf{Pre}[p,t]$. Replacing in (37) for the disabledness condition corresponding to transition t the following (less complex) condition:*

$$\left(\sum_{p\in\pi} \mathbf{m}[p] < \sum_{p\in\pi} \mathbf{Pre}[p,t]\right) \vee \left(\bigvee_{p\in {}^\bullet t\setminus\pi} \mathbf{m}[p] < \mathbf{Pre}[p,t]\right)$$

the set of integer *solutions is preserved.*

Proof. From $\mathbf{m}[p] \leq \mathbf{sb}[p]$ for every $p \in P$ and $\mathbf{m} \in \mathrm{LRS}^{\mathrm{SE}}(\mathcal{S})$, and $\mathbf{sb}[p] \leq \mathbf{Pre}[p,t]$ for every $p \in \pi$, it follows that $\sum_{p\in\pi}(\mathbf{m}[p] - \mathbf{Pre}[p,t]) \leq 0$ for every $\mathbf{m} \in \mathrm{LRS}^{\mathrm{SE}}(\mathcal{S})$. Since all the addends are non-positive, the "$<$" holds for the sum iff it holds for one of them. □

By the application of this result to a transition t, the number of linear systems to be solved is divided by $\frac{|{}^\bullet t|}{|{}^\bullet t|-|\pi|+1}$, what is deduced from the ratio between the number of input places to the transition and the actual number of them that need being considered separately.

In the particular case where $\pi = {}^\bullet t$, the disabledness condition is reduced to a linear inequality. Therefore a *single* system of linear inequalities is needed for structurally safe systems (i.e., those having all places with structural bound equal to one), for instance.

Also when *all but one* of the input places of a transition are such that their structural bound equals to the weight of the arc, the disabledness condition of the transition can be reduced to a linear inequality, applying the following result, which generalises Proposition 32:

Proposition 33. *Let t be a transition such that ${}^\bullet t = \pi \cup \{p'\}$, where $\mathbf{sb}[p'] > 0$ and $\mathbf{sb}[p] \leq \mathbf{Pre}[p,t]$ for every $p \in \pi$. Replacing in (37) for the disabledness condition corresponding to transition t the following (less complex) condition:*

$$\mathbf{sb}[p'] \cdot \sum_{p \in \pi} \mathbf{m}[p] + \mathbf{m}[p'] < \mathbf{sb}[p'] \cdot \sum_{p \in \pi} \mathbf{Pre}[p,t] + \mathbf{Pre}[p',t] \tag{41}$$

the set of integer *solutions is preserved.*

Proof. Let us rewrite first the condition in (37) for t disabled:

$$\left(\bigvee_{p \in \pi} \mathbf{m}[p] < \mathbf{Pre}[p,t] \right) \vee (\mathbf{m}[p'] < \mathbf{Pre}[p',t]) \tag{42}$$

It must be shown that (42) $\Leftrightarrow$ (41).
For "$\Rightarrow$", we distinguish two cases for t disabled:

1. Some $p'' \in \pi$ is such that $\mathbf{m}[p''] < \mathbf{Pre}[p'',t]$, so $\mathbf{m}[p''] \leq \mathbf{Pre}[p'',t] - 1$. Using also that $\mathbf{m}[p] \leq \mathbf{sb}[p]$ for every $p \in P$, and that $\mathbf{sb}[p] \leq \mathbf{Pre}[p,t]$ for every $p \in \pi$, we get:

$$\begin{aligned}
&\mathbf{sb}[p'] \cdot \textstyle\sum_{p \in \pi} \mathbf{m}[p] + \mathbf{m}[p'] = \\
&\mathbf{sb}[p'] \cdot \textstyle\sum_{p \in \pi \setminus \{p''\}} \mathbf{m}[p] + \mathbf{sb}[p'] \cdot \mathbf{m}[p''] + \mathbf{m}[p'] \leq \\
&\mathbf{sb}[p'] \cdot \textstyle\sum_{p \in \pi \setminus \{p''\}} \mathbf{Pre}[p,t] + \mathbf{sb}[p'] \cdot (\mathbf{Pre}[p'',t] - 1) + \mathbf{m}[p'] = \\
&\mathbf{sb}[p'] \cdot \textstyle\sum_{p \in \pi} \mathbf{Pre}[p,t] - (\mathbf{sb}[p'] - \mathbf{m}[p']) \leq \\
&\mathbf{sb}[p'] \cdot \textstyle\sum_{p \in \pi} \mathbf{Pre}[p,t] < \\
&\mathbf{sb}[p'] \cdot \textstyle\sum_{p \in \pi} \mathbf{Pre}[p,t] + \mathbf{Pre}[p',t]
\end{aligned}$$

2. Now p' is such that $\mathbf{m}[p'] < \mathbf{Pre}[p',t]$. Using that $\mathbf{m}[p] \leq \mathbf{sb}[p]$ for every $p \in P$, and that $\mathbf{sb}[p] \leq \mathbf{Pre}[p,t]$ for every $p \in \pi$, the result follows.

For "$\Leftarrow$", let us rewrite first (41):

$$\sum_{p \in \pi} (\mathbf{m}[p] - \mathbf{Pre}[p,t]) < \frac{\mathbf{Pre}[p',t] - \mathbf{m}[p']}{\mathbf{sb}[p']} \tag{43}$$

Assume contrary: (43) holds and t is enabled. In particular, $\mathbf{m}[p'] \geq \mathbf{Pre}[p',t]$, hence $\sum_{p \in \pi} (\mathbf{m}[p] - \mathbf{Pre}[p,t]) < 0$. Since $\mathbf{m}[p] - \mathbf{Pre}[p,t] \leq 0$ for every $p \in \pi$, it follows that there exists $p \in \pi$ such that $\mathbf{m}[p] - \mathbf{Pre}[p,t] < 0$. Thus, t is not enabled, against the hypothesis. □

By the application of this result to a transition t, the number of linear systems to be solved is obviously divided by $|{}^\bullet t|$.

Let us illustrate this latter rule. The idea is to describe the set of solutions to the state equation for which t is disabled by means of a linear inequality. In the case of Figure 10, the markings for which t is disabled are those for which $\mathbf{m}[p] < 3$ and those for which $\mathbf{m}[p'] < 1$. Observe that, taking into account that

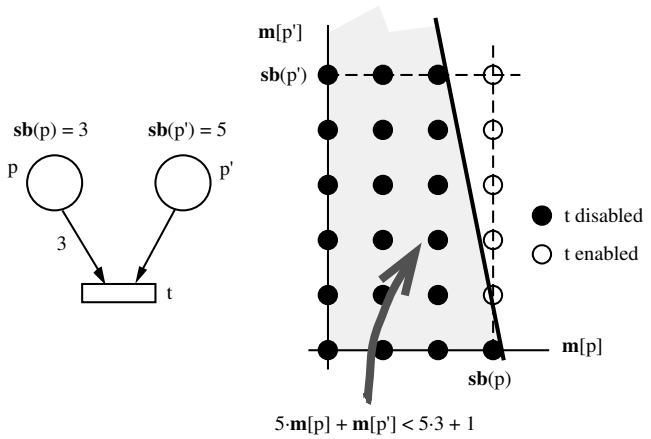

Fig. 10. Illustration of Proposition 33.

the solutions to the state equation respect, in particular, the structural bounds, all the solutions for which t is disabled are among the integer points in the region described by $5 \cdot \mathbf{m}[p] + \mathbf{m}[p'] < 5 \cdot 3 + 1$. Notice that the slope of the boundary of such region could be greater than that given by $\mathbf{sb}[p']$, provided it is not vertical. (Therefore, in case we finally check non-existence of solutions using integer programming, it makes no difference to use any greater value.)

In summary, up to now, apart from disregarding transitions with a stronger or equal precondition than others and dead transitions, we are able to write as a linear inequality the disabledness condition for all transitions $t \in T$ that have *at most* one input place p such that $\mathbf{sb}[p] > \mathbf{Pre}[p, t]$.

Proposition 32 can also be applied to reduce the number of terms when $|\pi - {}^\bullet t| > 1$, and $|\pi| > 1$, although in this case the disabledness condition is not reduced to a single term.

We can still further reduce the number of systems to solve by preapplying a transformation to the system that preserves deadlock-freeness (actually, it preserves the projected language). The transformation, illustrated in Figure 11, can be applied as needed to every place p with homogeneous weighting. After the transformation, we have one more transition ($t^{(p)}$ in the figure), the disabledness condition of which can be written as a linear inequality because the structural bound of $p^{(c)}$ is one. On the other hand, the structural bound of $p^{(b)}$ is also one, thus we have in each transition in $p^\bullet$ one input place less with structural bound greater than the weight (perhaps only one, or even none, remains, and then the disabledness condition for such transition can be written as a linear inequality too).

After the presented results, clearly the state equation based verification of deadlock-freeness reduces to checking non-existence of (integer) solution to a *single* linear system of inequalities in the case of structurally bounded P/T systems with homogeneous weighting — in particular ordinary and equal conflict

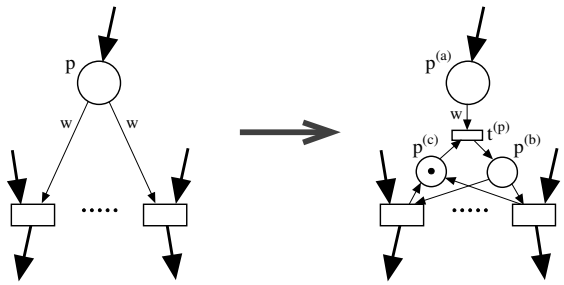

Fig. 11. A transformation preserving the projected language (in particular, preserving deadlock-freeness).

systems — because the transformation in Figure 11 can be applied as necessary to enable Propositions 32 or 33. Moreover, since every P/T system can be simulated by another with homogeneous weighting preserving the projected language (see Figure 12 for an illustrative example of the kind of transforma-

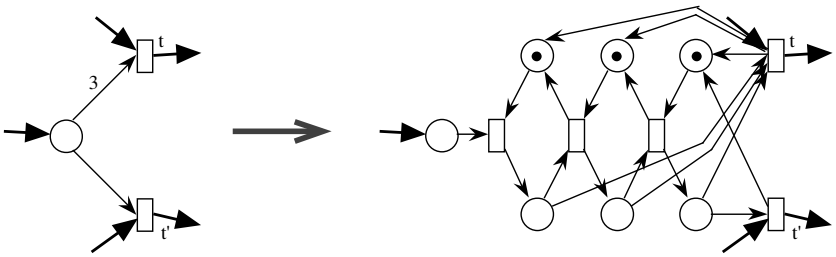

Fig. 12. Simulating weights with ordinary nets preserving the projected language.

tion used; more compact transformations exist for particular cases), it follows that *every* structurally bounded P/T system (or merely known to be k-bounded, because these can always be made structurally bounded using the complementary place construction) can be transformed to require a single linear system of inequalities:

Theorem 34. *Let $\mathcal{S}$ be a structurally bounded P/T system. Then (37) in the sufficient condition for deadlock-freeness given by Theorem 30 can be rewritten as a single system of linear inequalities.*

Remark 35. Since live equal conflict systems do not have "killing" spurious solutions (i.e., spurious deadlocks), existence of a solution guarantees non liveness. If no deadlock solution is found the system is proven deadlock-free, what in bounded strongly connected equal conflict systems implies liveness. Observe that applying the transformation illustrated in Figure 11 to an equal conflict system

leads to another equal conflict system. This allows in this class to characterise liveness, in the presence of boundedness, through non-existence of integer solution to a single system of linear inequalities [92]. □

Let us illustrate the application of some of the above techniques to a simple example. Consider the system in Figure 13 (a), where all the structural marking

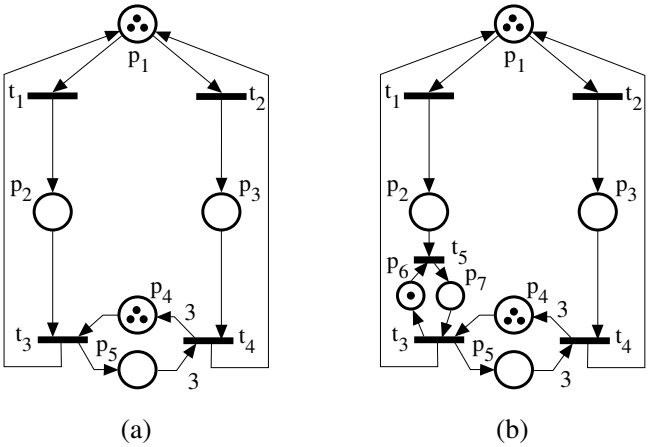

Fig. 13. Example of deadlock-freeness analysis.

bounds are three. The disabledness condition for t_2 can be disregarded. Transition t_1 has only one input place, so its disabledness condition is linear. Transition t_4 has only one input place (p_3) with structural bound greater than the weight, so its disabledness condition can be written linearly applying Proposition 33. The structural bounds of both input places of t_3 are greater than the corresponding weights, but we can apply the transformation in Figure 11 to one of them, e.g., to p_2, see Figure 13 (b). Now the disabledness conditions of t_3 and t_5 can both be written linearly.

Therefore, finally, the analysis can be done using a single linear system. That is, if $\mathbf{m}$ is a deadlock then the following linear system has a solution:

$$\begin{aligned} \mathbf{m} - \mathbf{C} \cdot \boldsymbol{\sigma} = \mathbf{m_0} \wedge \mathbf{m}, \boldsymbol{\sigma} \geq \mathbf{0} \ &\text{(state eq.)} \\ \mathbf{m}[p_1] \leq 0 \qquad &(t_1) \\ 3 \cdot \mathbf{m}[p_5] + \mathbf{m}[p_3] \leq 9 \qquad &(t_4) \\ 3 \cdot \mathbf{m}[p_7] + \mathbf{m}[p_4] \leq 3 \qquad &(t_3) \\ 3 \cdot \mathbf{m}[p_6] + \mathbf{m}[p_2] \leq 3 \qquad &(t_5) \end{aligned}$$

Solving (with the simplex algorithm) we obtain, for instance, the following solution: $\mathbf{m} = 3p_3 + 3p_4 + p_6$, $\boldsymbol{\sigma} = 3t_2$, which in this case is actually reachable.

Remark 36. Notice that Proposition 33 and transformations like that illustrated in Figure 11 are intended to express the disabledness of a transition linearly.

Therefore, they are not only useful for deadlock-freeness analysis but for every analysis that requires checking the disabledness of some transitions. In particular, they can be applied to analyse the non fireability of *facts* [37]. □

5 Improving the State Equation

A major limitation of the analysis methods based on the state equation that we have described in the previous sections stems from the existence of *spurious solutions*: only necessary or sufficient conditions for the analysed properties are obtained. As an example, for the net system in Figure 1, due to the spurious solutions, the state equation based analysis does not allow to prove safeness, deadlock-freeness, or pairwise mutual exclusion between p_2, p_3, and p_4. This limitation motivates the interest of the techniques that we present in this section to remove some, ideally all, of the spurious solutions (with respect to $\mathrm{LRS}^{\mathrm{SE}\mathbb{R}}$).

In Subsection 3.5 we observed that traps and siphons induce new invariant laws that add information to the state equation. We concentrate here on traps. Assume Θ is an initially marked trap of $\mathcal{S}$. To fix ideas, consider $\Theta = \{p_1, p_2, p_5\}$ of the net in Figure 1. An initially marked trap induces a trap invariant as an inequality, which in the example is:

$$\mathbf{m}[p_1] + \mathbf{m}[p_2] + \mathbf{m}[p_5] \geq 1 \tag{44}$$

Now assume that we have a P-semiflow $\mathbf{y}$ whose support includes Θ, which induces another marking invariant concerning the places of Θ. The P-semiflow $\mathbf{y} = 2p_1 + p_2 + p_3 + p_4 + p_5$ in the example, induces the marking invariant:

$$2\mathbf{m}[p_1] + \mathbf{m}[p_2] + \mathbf{m}[p_3] + \mathbf{m}[p_4] + \mathbf{m}[p_5] = 2 \tag{45}$$

By subtracting (44) from (45) we obtain another inequality invariant:

$$\mathbf{m}[p_1] + \mathbf{m}[p_3] + \mathbf{m}[p_4] \leq 1 \tag{46}$$

By introducing a *slack* variable, we can transform the latter inequality into an equality. The non-negative slack variable can be interpreted as the marking of a new place, p_Θ:

$$\mathbf{m}[p_1] + \mathbf{m}[p_3] + \mathbf{m}[p_4] + \mathbf{m}[p_\Theta] = 1 \tag{47}$$

Observe that the new place is marking (structurally) implicit. In the example, and taking into account (45):

$$\begin{aligned} \mathbf{m}[p_\Theta] &= 1 - (\mathbf{m}[p_1] + \mathbf{m}[p_3] + \mathbf{m}[p_4]) \\ &= 1 - (2 - (\mathbf{m}[p_1] + \mathbf{m}[p_2] + \mathbf{m}[p_5])) \\ &= \mathbf{m}[p_1] + \mathbf{m}[p_2] + \mathbf{m}[p_5] - 1 \end{aligned} \tag{48}$$

The place p_Θ coincides with the place p that we considered in Subsection 4.4 after Theorem 25. Remember that, when marked initially with the optimum value of (31), which is zero in this case, the addition of this place, which is (marking)

implicit, removes some spurious solutions. Actually, from (48) it is not difficult to see that the removed solutions are precisely those that violate the trap invariant (44): both perspectives, in terms of trap invariants and marking (structurally) implicit places, are "dual" ways of interpreting the same removal of spurious solutions from the state equation.

The above reasoning shows that basic trap invariants, i.e., the information that a set of places must remain marked, can be *coded* in a marking structurally implicit place that is added to the original net system. The converse is also true: when a marking structurally implicit place removes, or *cuts*, spurious solutions, the cut can be interpreted in terms of trap invariants, as we show in Subsection 5.1. Although not complete, this leads to a procedure to add cutting implicit places to improve the accuracy of a given state equation, that we describe in Subsection 5.2. Making use of the linear algebraic formulation of the property that initially marked traps remain marked given in (18), it is also possible to improve the state equation by directly adding a generator of trap invariants in order to require this property, what is presented in Subsection 5.3.

5.1 Cutting Implicit Places

After realising that structural implicit places can improve the state equation by cutting spurious solutions, several questions naturally arise:

- Which structural implicit places do cut?
- Which spurious solutions are cut?
- Is it possible to eliminate in this way *any* spurious solution?

We devote this subsection to answering them.

Let p be a structurally implicit place with $\mathbf{m_0}[p]$ equal to the optimal value of (31), and let $\mathbf{y}$ and μ be a corresponding optimal solution. If $\mu \geq 0$, then

$$\mathbf{m}[p] = \mathbf{m_0}[p] + \mathbf{C}[p,T] \cdot \boldsymbol{\sigma} \geq \mathbf{y} \cdot (\mathbf{m_0}[P] + \mathbf{C}[P,T] \cdot \boldsymbol{\sigma}) = \mathbf{y} \cdot \mathbf{m}[P] \geq 0 \quad (49)$$

so the inequality $\mathbf{m}[p] \geq 0$ becomes *redundant* in the state equation ($\mathbf{m}[p]$ is a *non-extremal variable* in the terminology of convex geometry), hence its addition or removal does not affect $\mathrm{LRS}^{\mathrm{SE}\mathbb{R}}$. (In the particular case of live marked graphs, where $\mathrm{RS} = \mathrm{LRS}^{\mathrm{SE}\mathbb{R}}$, every implicit place is of this kind.) On the contrary, when $\mu < 0$ in every optimal solution, the constraint on non-negativity of $\mathbf{m}[p]$ is not redundant, so the state equation with p has less solutions that the state equation without it; since p is implicit, the difference are spurious solutions, precisely those where $\mathbf{m}[p]$ would have been negative. Taking into account that

$$\begin{aligned} \mathbf{m}[p] &= \mathbf{y} \cdot \mathbf{m_0}[P] + \mu + \mathbf{C}[p,T] \cdot \boldsymbol{\sigma} \\ &= \mathbf{y} \cdot (\mathbf{m}[P] - \mathbf{C}[P,T] \cdot \boldsymbol{\sigma}) + \mu + \mathbf{C}[p,T] \cdot \boldsymbol{\sigma} \end{aligned} \quad (50)$$

we can state that:

Theorem 37. *Let $\mathcal{S}$ be a P/T system. Let p be a structurally implicit place, and let $\mathbf{m_0}[p]$ be the optimal value of (31) corresponding to the optimal solution $\mathbf{y} \geq \mathbf{0}$ and $\mu < 0$. The addition of p to $\mathcal{S}$ cuts at least one spurious solution (with respect to* $\mathrm{LRS}^{\mathrm{SE\mathbb{R}}}$*), i.e., p is a* cutting implicit place*, iff there is no solution to*

$$\begin{gathered} \mathbf{y}' \cdot \mathbf{C}[P,T] \leq \mathbf{C}[p,T] \\ \mathbf{y}' \cdot \mathbf{Pre}[P,p^\bullet] + \mu' \mathbf{1} \geq \mathbf{Pre}[p,p^\bullet] \\ \mathbf{y}' \cdot \mathbf{m_0}[P] + \mu' = \mathbf{m_0}[p] \\ \mathbf{y}', \mu' \geq \mathbf{0} \end{gathered} \tag{51}$$

The spurious solutions that are cut are those that fulfill:

$$\mathbf{y} \cdot \mathbf{m}[P] + (\mathbf{C}[p,T] - \mathbf{y} \cdot \mathbf{C}[P,T]) \cdot \boldsymbol{\sigma} < -\mu \tag{52}$$

In other words, the addition of the cutting implicit place p expresses the additional restriction that in every reachable marking the following holds:

$$\mathbf{y} \cdot \mathbf{m}[P] \geq -\mu - (\mathbf{C}[p,T] - \mathbf{y} \cdot \mathbf{C}[P,T]) \cdot \boldsymbol{\sigma} \tag{53}$$

Observe that $(\mathbf{C}[p,T] - \mathbf{y} \cdot \mathbf{C}[P,T]) \cdot \boldsymbol{\sigma} \geq 0$ and that it depends on $\boldsymbol{\sigma}$. For the case of a marking structurally implicit place, $\mathbf{C}[p,T] = \mathbf{y} \cdot \mathbf{C}[P,T]$, and $\mathbf{y} \cdot \mathbf{m}[P] \geq -\mu$. This reminds us a trap invariant, where $\|\mathbf{y}\|$ is the trap and $-\mu$ is the minimal (weighted) token content. To demonstrate this fact, rewriting $\mathbf{C}$ as $\mathbf{Post} - \mathbf{Pre}$ we get:

$$\mathbf{y} \cdot \mathbf{Post}[P,T] - \mathbf{y} \cdot \mathbf{Pre}[P,T] = \mathbf{Post}[p,T] - \mathbf{Pre}[p,T] \tag{54}$$

Assume without loss of generality that p is pure, i.e., $p^\bullet \cap {}^\bullet p = \emptyset$. (If p was not pure, the same place cancelling the self-loops would be also marking structurally implicit, possibly with a lesser value of μ, i.e., with a "more negative" μ.) Considering separately $p^\bullet$, ${}^\bullet p$, and $T - (p^\bullet \cup {}^\bullet p)$, we have:

$$\mathbf{y} \cdot \mathbf{Post}[P,p^\bullet] - \mathbf{y} \cdot \mathbf{Pre}[P,p^\bullet] = -\mathbf{Pre}[p,p^\bullet] < \mathbf{0} \tag{55}$$

$$\mathbf{y} \cdot \mathbf{Post}[P,{}^\bullet p] - \mathbf{y} \cdot \mathbf{Pre}[P,{}^\bullet p] = \mathbf{Post}[p,{}^\bullet p] > \mathbf{0} \tag{56}$$

$$\mathbf{y} \cdot \mathbf{Post}[P,T - (p^\bullet \cup {}^\bullet p)] - \mathbf{y} \cdot \mathbf{Pre}[P,T - (p^\bullet \cup {}^\bullet p)] = \mathbf{0} \tag{57}$$

Since $\mathbf{y} \cdot \mathbf{Pre}[P,p^\bullet] + \mu \mathbf{1} \geq \mathbf{Pre}[p,p^\bullet]$, because $\mathbf{y}$, μ is a solution to (31), Equation (55) becomes:

$$\mathbf{y} \cdot \mathbf{Post}[P,p^\bullet] \geq -\mu \mathbf{1} > \mathbf{0} \tag{58}$$

From (56–58), for every transition t we have $\mathbf{y} \cdot (\mathbf{Post}[P,t] - \mathbf{Pre}[P,t]) \geq 0$. It follows that if $\mathbf{y} \cdot \mathbf{Pre}[P,t] \neq 0$, then $\mathbf{y} \cdot \mathbf{Post}[P,t] \neq 0$, i.e., if $t \in \|\mathbf{y}\|^\bullet$ then $t \in {}^\bullet\|\mathbf{y}\|$, so $\|\mathbf{y}\|$ is effectively a trap. In summary:

Theorem 38. *Let $\mathcal{S}$ be a P/T system. Let p be a marking structurally implicit place, and let $\mathbf{m_0}[p]$ be the optimal value of (31) corresponding to the optimal solution $\mathbf{y} \geq \mathbf{0}$ and $\mu < 0$. If p is a cutting implicit place (according to Theorem 37) then $\|\mathbf{y}\|$ is a trap of $\mathcal{N}$ such that its weighted token content, defined by $\mathbf{y} \cdot \mathbf{m}[P]$, is never less than $-\mu$.*

In order to further illustrate the "dual" interpretation of the cut in terms of implicit places and traps, and also to introduce the limitations of this approach, consider the example shown in Figure 14, which is a most spare PN representation of a prototype distributed mutual exclusion algorithm. Let $q = r = 1$, to

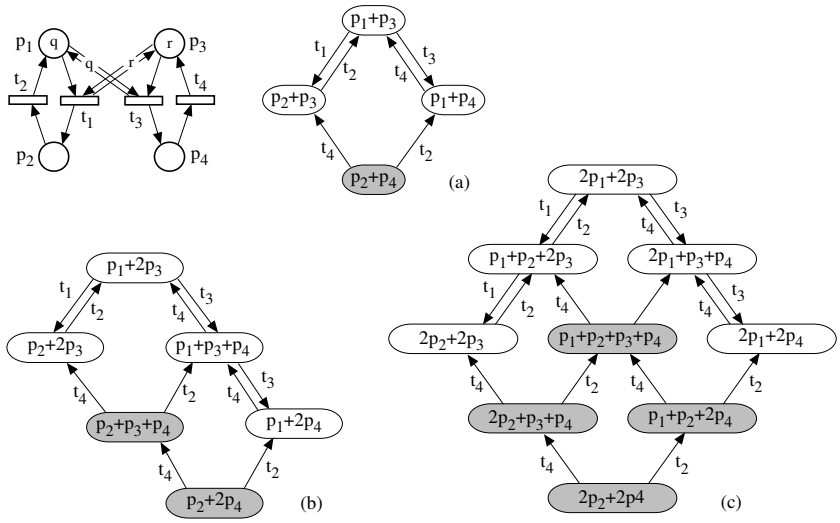

Fig. 14. A family of P/T systems where p_2 and p_4 are in mutual exclusion, and the $\mathrm{LRG}^{\mathrm{SE}}$ for (a) $q = r = 1$, (b) $q = 1, r = 2$, and (c) $q = r = 2$.

start with. The marking $p_2 + p_4$ is a (spurious) solution to the state equation, which prevents proving mutual exclusion of p_2 and p_4 using the state equation (notice that the same happens in the net where the self-looped transitions are splitted, which is merely more cumbersome). The place p: $\mathbf{C}[p, T] = t_2 + t_4 - t_1 - t_3$, initially marked with one token, is marking implicit; an optimal solution to (31) is $\mathbf{y} = p_1 + p_3$, $\mu = -1$. Its addition cuts the markings where $\mathbf{y} \cdot \mathbf{m} = \mathbf{m}[p_1] + \mathbf{m}[p_3] < 1 = -\mu$, that is, the markings where the initially marked trap $\{p_1, p_3\}$ is unmarked. In this case the only spurious marking is cut, so mutual exclusion can be proven using the state equation method in the net with the implicit place.

Let $q = 1$ and $r = 2$. Now the markings $p_2 + p_3 + p_4$ and $p_2 + 2p_4$ are spurious (and prevent us from proving mutual exclusion the same as before). The place p used above, marked with two tokens, is marking implicit, but does not allow to prove mutual exclusion because it does not cut the spurious marking $p_2 + p_3 + p_4$. Fortunately, a different choice of the weights of the arcs helps. The place p': $\mathbf{C}[p', T] = 2t_2 + t_4 - 2t_1 - t_3$, initially marked with two tokens, is marking implicit; an optimal solution to (31) is $\mathbf{y} = 2p_1 + p_3$, $\mu = -2$. Its addition cuts the markings where $2\mathbf{m}[p_1] + \mathbf{m}[p_3] < 2$, that is, the markings where the weighted token content of the initially marked trap $\{p_1, p_3\}$ is *less than two*. (Again in this

case all the spurious markings are cut.) What is remarkable in this case is the fact that we are removing even spurious markings, namely $p_2+p_3+p_4$, in which the trap *is marked* (with a weighted token content below a minimum, though). We shall come back to this case in Subsection 5.2.

Finally, let $q = r = 2$. Although the addition of the place p: $\mathbf{C}[p,T] = t_2+t_4-t_1-t_3$, initially marked with two tokens, cuts the markings where $\mathbf{m}[p_1]+\mathbf{m}[p_3] < 2$, the spurious marking $p_1+p_2+p_3+p_4$ is not cut. Notice that this spurious marking *cannot* be cut by any other place because it is a positive linear combination of two reachable markings: $2p_2+2p_3$ and $2p_1+2p_4$ (the reachability set is *not convex* in this case). This example manifests the incompleteness of the addition of cutting implicit places in order to remove spurious solutions: they *improve* the linear description, but *not always* completely.

Remark 39. The incompleteness of the addition of cutting implicit places refers to applying this method *alone*. It is conceivable that the net to be analysed can be transformed in such a way that the properties under study are preserved and in the transformed net spurious solutions can be removed by cutting implicit places, e.g., with the transformation rule shown in Figure 11. For instance, mutual exclusion of p_2 and p_4 of the net system in Figure 14 with $q = r = 2$ is equivalent to mutual exclusion of p_2' and p_4 in the net system in Figure 15 (a). Place p shown in (b) with $\mathbf{m_0}[p] = 2$ is marking (structurally) implicit and removes the spurious solutions where p_2' and p_4 are simultaneously marked, what allows to conclude through state equation analysis that p_2 and p_4 were in mutual exclusion in the original system.

It is remarkable that in this case the initial marking of p computed from (31) is four instead of two, so it *is not* the minimal marking that makes p implicit, showing that Theorem 23 is only sufficient for p implicit. □

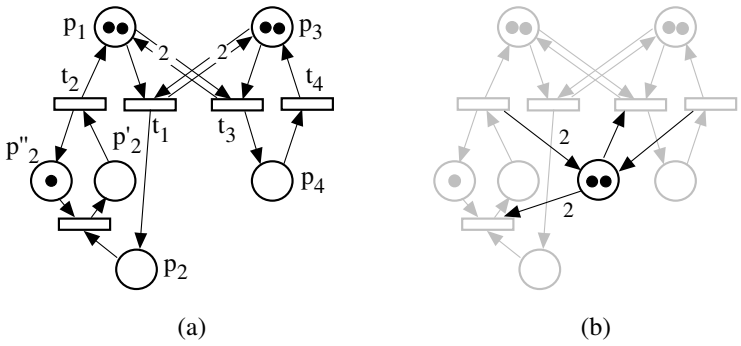

Fig. 15. Mutual exclusion of p_2 and p_4 of the net system in Figure 14 with $q = r = 2$ is equivalent to mutual exclusion of p_2' and p_4 in the net system (a). Adding the implicit place p shown in (b) allows to prove mutual exclusion.

5.2 Improving the State Equation with Implicit Places

When faced to a net system the state equation of which is to be improved by adding cutting implicit places, a question arises: how can we select the candidate places? We outline here some indications to answer this question.

From the association of cutting implicit places and initially marked traps that are insufficiently marked in some spurious solutions, the following procedure naturally comes to mind:

1. Compute (minimal) initially marked traps.
2. For each trap Θ a marking structurally implicit place is obtained as $\mathbf{C}[p_\Theta, T] = \mathbf{1}_\Theta \cdot \mathbf{C}[P, T]$, that is, taking $\mathbf{y} = \mathbf{1}_\Theta$ in Theorem 25.
3. Let the initial marking for p_Θ be the optimal value of (31) fixing $\mathbf{y} = \mathbf{1}_\Theta$. If $\mu < 0$ in the optimal solution, then apply Theorem 37 to verify that it cuts and to obtain the expression of the achieved cut, and add the place if it cuts.

The above procedure removes all the spurious solutions that can be proven non reachable observing that the *non weighted* token content of an initially marked trap is less than a minimum value, particularly those where the trap is unmarked. For instance, to improve the state equation of the net system in Figure 1, we would consider the initially marked traps $\Theta_1 = \{p_1, p_2, p_5\}$, $\Theta_2 = \{p_1, p_3, p_5\}$, and $\Theta_3 = \{p_1, p_2, p_3, p_4\}$, that lead to the corresponding marking structurally implicit places shown in Figure 16 (a). Their initial markings,

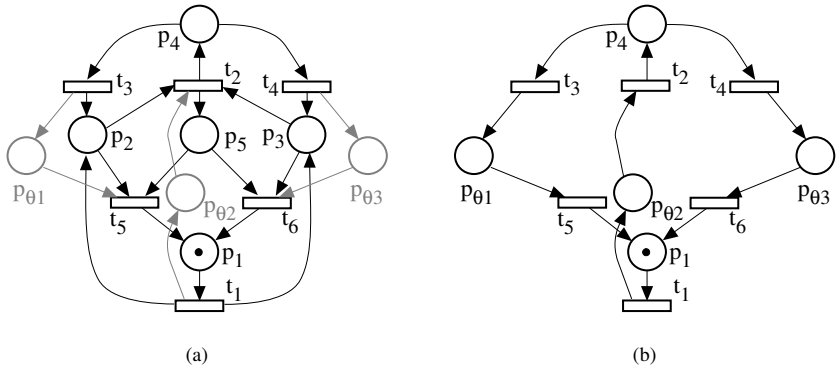

Fig. 16. Adding implicit places to the net system in Figure 1 cuts every spurious solution. Some original places become implicit when the cutting implicit places are added.

obtained from (31), are all zero, and all of them cut spurious solutions (we have already discussed in detail the case of p_{Θ_1}). Actually, in this case, the reader can easily check that *every* spurious solution is cut. It can also be checked that places p_2, p_3, and p_5 are implicit in the net system of Figure 16 (a). Their removal leads to the system in Figure 16 (b), which is a live and safe state machine isomorphic to the reachability graph.

Remark 40. If a different $\mathbf{y}$ is taken in the above procedure to compute cutting implicit places from given traps, different solutions appear. For the example in Figure 1 and 16, an alternative set of cutting implicit places is shown in [23]. □

Let us come back to the example of Figure 14 with $q = 1$ and $r = 2$. If we apply the above procedure, from the initially marked trap $\Theta_1 = \{p_1, p_3\}$ we obtain the marking structurally implicit place p_{Θ_1}: $\mathbf{C}[p_{\Theta_1}] = t_2 + t_4 - t_1 - t_3$. The initial marking, obtained from (31), is two, corresponding to $\mu = -1$, so the addition of p_{Θ_1} introduces the new P-invariant $\mathbf{m}[p_2]+\mathbf{m}[p_4]+\mathbf{m}[p_{\Theta_1}] = 2$, which cuts the spurious solution $p_2 + 2p_4$ but not $p_2 + p_3 + p_4$. In principle, it seems that the latter spurious solution cannot be cut because, as we discussed, the matter is the *weighted* token content of the trap, not merely the token content. It may be surprising at first sight that it is cut if the above procedure *is applied iteratively*. A little thought reveals that this is quite natural, since the addition of new places originates new traps, so new chances for improving. (Of course, only traps including newly added places are useful now.) Effectively, besides other traps, after the addition of p_{Θ_1}, $\Theta_2 = \{p_1, p_{\Theta_1}\}$ is an initially marked trap, that leads to p_{Θ_2}: $\mathbf{C}[p_{\Theta_2}] = 2t_2 + t_4 - 2t_1 - t_3$. As we saw, when initially marked with two tokens, this place removes all the spurious solutions. (In fact, once p_{Θ_2} is added, p_{Θ_1} becomes redundant and can be removed.)

We can proceed until no spurious solution is removed, i.e., no cutting implicit place is added, in an iteration. (Naturally, the procedure stops after a finite number of iterations in structurally bounded nets, because the number of possible spurious solutions is finite.) We insist that this *does not* guarantee that every spurious solution has been removed.

Remark 41. An alternative way of finding a cutting implicit place, in the case that we are trying to disprove reachability of some (sub)markings that we suspect are spurious, is postulating a *monitor place* that forbids reaching them, and then analysing whether it is implicit or not. Consider again the example of Figure 14 with $q = 1$ and $r = 2$. As we want to prove that p_2 and p_4 are in mutual exclusion, but the state equation does not allow us to conclude, we introduce a place that forces the mutual exclusion of p_2 and p_4, that is a place p such that $2\mathbf{m}[p_2] + \mathbf{m}[p_4] + \mathbf{m}[p] = 2$ becomes a (new) P-invariant. As we know, the place is p: $\mathbf{C}[p, T] = 2t_2 + t_4 - 2t_1 - t_3$, initially marked with two tokens, which is effectively implicit, so we are done.

This approach is specially suitable for structurally safe systems, where the place that forbids a given (sub)marking $\mathbf{m} = \mathbf{1}_{\|\mathbf{m}\|}$ is simply p: $\mathbf{C}[p, T] = -\mathbf{m} \cdot \mathbf{C}[P, T]$, initially marked with $\|\mathbf{m}\| - 1 - \mathbf{m} \cdot \mathbf{m_0}[P]$ tokens (this is the kind of *constraints* introduced by Patil, see [67]). Actually, since doing so we can remove any one precise marking, if the spurious solutions were known, all of them could be removed. This implies that for any given structurally repetitive and structurally safe system a "place completed" version with the same behaviour exists such that $\mathrm{RS} = \mathrm{LRS}^{\mathrm{Psf}}$. □

5.3 Improving the State Equation with a Generator of Trap Invariants

Equation (18) expresses linearly that initially marked traps remain marked. In order to add this condition to the state equation, it must be stated in terms of existence of solution, what can be done using the alternatives theorem. Doing so, we obtain that initially marked traps are marked under $\mathbf{m}$ iff there is a solution to:

$$\mathbf{C}_\Theta \cdot \mathbf{x}' + x'\mathbf{m_0} - \alpha\mathbf{m} \leq \mathbf{0} \wedge \mathbf{x}' \geq \mathbf{0} \wedge x' > 0 \wedge \alpha \geq 0 \tag{59}$$

If a solution with $\alpha = 0$ exists, then so does a solution with $\alpha > 0$. Dividing (59) by α we get:

$$\mathbf{C}_\Theta \cdot \mathbf{x} + x\mathbf{m_0} - \mathbf{m} \leq \mathbf{0} \wedge \mathbf{x} \geq \mathbf{0} \wedge x > 0 \tag{60}$$

This condition can be interpreted as a generator of trap invariants. It can be added to the state equation leading to the following *improved state equation* (integrality of $\boldsymbol{\sigma}$ can be required for better accuracy, while $\mathbf{x}$ and x are rational because they originate in the transformation of (18) by the alternatives theorem):

Theorem 42. *Let $\mathcal{S}$ be a P/T system. If $\mathbf{m} \in \mathrm{RS}(\mathcal{S})$ then it is a solution to:*

$$\begin{aligned} \mathbf{m} - \mathbf{C} \cdot \boldsymbol{\sigma} &= \mathbf{m_0} \\ \mathbf{m} - x\mathbf{m_0} - \mathbf{C}_\Theta \cdot \mathbf{x} &\geq \mathbf{0} \\ \mathbf{m}, \boldsymbol{\sigma}, \mathbf{x} &\geq \mathbf{0} \\ x &> 0 \end{aligned} \tag{61}$$

where $\mathbf{C}_\Theta$ is as defined in Theorem 13.

Remark 43. Since in live, bounded, and reversible free choice systems the reachable markings are the vectors in $\mathrm{LRS}^{\mathrm{Psf}}$ that mark every trap [30], Theorem 42 characterises reachability. □

This method has the advantage that no previous computation of traps is required. (It can be said that the method based on implicit places is "compiled", in the sense that trap invariants obtained in some previous or off-line computation are incorporated or coded as part of the — transformed — net. Following the same analogy, the method based on a generator of trap invariants is "interpreted", in the sense that markings that are solution to the state equation but violate the trap condition are eliminated on-line.)

Nevertheless, the gain in efficiency of this method is paid by the loss in accuracy compared to the method based on implicit places: only markings where an initially marked trap is unmarked are cut. For instance, the marking $p_2 + p_3 + p_4$ in the net system of Figure 14 with $q = 1$ and $r = 2$ is still spurious with respect to the improved state equation because no trap is unmarked.

The advantages of both methods can be combined by (iteratively) adding cutting implicit places to a given net up to a certain point, and then using the improved state equation (i.e., incorporating the generator of trap invariants) instead of the plain state equation for the analysis of properties.

6 Structural Liveness and the Rank Theorems

We have encountered that the state equation method is best suited to the analysis of properties formulated in terms of existence or non-existence of markings (and firing vectors), such as marking bounds, mutual exclusion, or even deadlock-freeness, for which it provides at least necessary or sufficient conditions. For other properties the statement of which combines existential and universal quantifiers, e.g., liveness: *for every* reachable marking *there exists* a reachable successor that enables t, the direct application of the state equation method does not allow to reach any conclusion.

Of course, the sufficient condition for deadlock-freeness is useful for liveness analysis (deadlock-freeness is necessary for liveness; even in some cases, e.g., bounded strongly connected equal conflict systems, it is sufficient). But this is not the only way in which the incidence matrix of a net can be exploited to facilitate the analysis of liveness. In this section we describe in some detail efficient tests — polynomial time — that give necessary conditions for the existence of an initial marking that makes a given net live (and bounded). Such necessary conditions are also sufficient in some net subclasses.

6.1 The Rank Theorem: A General Necessary Condition for Liveness and Boundedness

A well-known polynomial time necessary condition for liveness and boundedness of a net system, based solely on purely structural properties, is strong connectedness [77] and consistency (Proposition 10) of the net. (As stated by Theorem 20, for structural boundedness and structural liveness, conservativeness and consistency are necessary.)

These conditions are very useful to discard models that are not correct before undertaking a more costly analysis. Unfortunately they are only necessary: there are strongly connected and consistent nets that cannot be lively and boundedly marked (see Figure 17).

We present here an improved — and still polynomial time — necessary condition, that incorporates an upper bound for the rank of the incidence matrix, namely that it must be less than the number of equal conflict sets of the net. We make use of *circuit arbiters*, which are a particular class of the *regulation nets* of [47] that we use to regulate non-trivial equal conflicts. For an equal conflict set $e \in \mathrm{SEQS}$, its circuit arbiter is defined as follows:

Definition 44. Let $\mathcal{N}$ be a P/T net, and let $e \in \mathrm{SEQS}$ such that $|e| > 1$. A net $\mathcal{A}_e = \langle P_e, e, \mathbf{Pre}_e, \mathbf{Post}_e \rangle$ is an *(ordinary) circuit arbiter* for the equal conflict set e iff $\mathcal{A}_e$ is an ordinary net such that $P_e \cap P = \emptyset$ and its underlying graph is an elementary circuit.

Some straightforward properties of these arbiters are: being circuits, they have the same number of places and transitions, i.e., $|e|$; the set of places of a

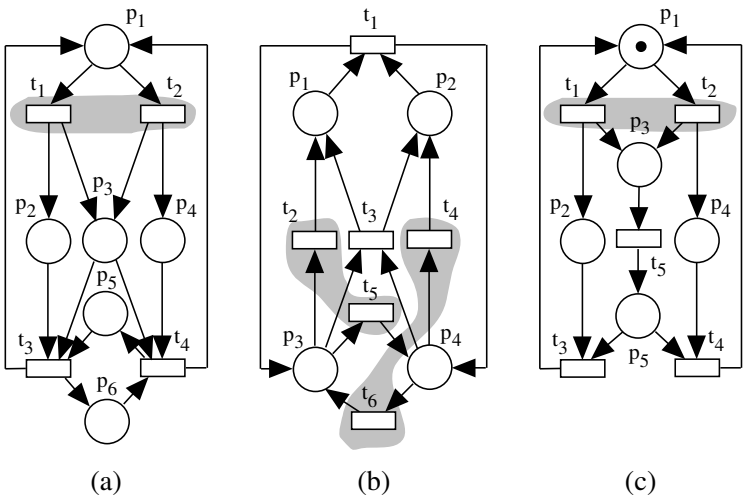

Fig. 17. Three conservative and consistent nets with $\text{rank}(\mathbf{C}) = 3$. Their (non-trivial) equal conflict sets are shaded. The nets (a) and (b) cannot be lively and boundedly marked, while (c) is live and bounded with the marking shown.

circuit arbiter in a net is the support of a minimal P-semiflow; with every non-empty initial marking, a circuit arbiter is live, bounded, and reversible. Figure 18 represents a circuit arbiter (shaded places) merged on an equal conflict set.

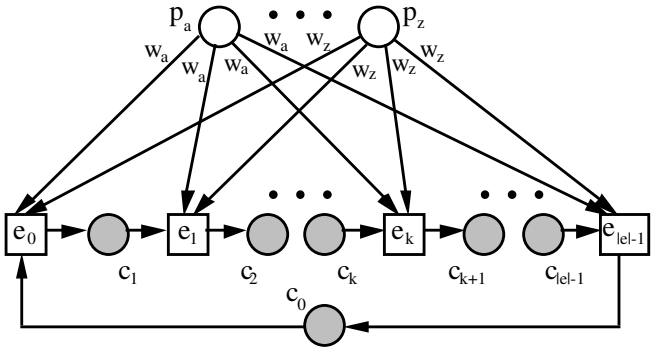

Fig. 18. A circuit arbiter (shaded places) merged on an equal conflict set.

Theorem 45 (The rank theorem). *If $\mathcal{S}$ is a live and bounded P/T system, then $\mathcal{N}$ is strongly connected, consistent, and* $\text{rank}(\mathbf{C}) < |\text{SEQS}|$.

A weaker but more "symmetric" statement, that clearly shows the extension of Theorem 20, is:

Corollary 46. *If $\mathcal{N}$ is a structurally live and structurally bounded P/T net, then $\mathcal{N}$ is conservative, consistent, and* $\text{rank}(\mathbf{C}) < |\text{SEQS}|$.

Lemma 47. *Let $\mathcal{S}$ be a P/T net, and let $e \in \text{SEQS}$ such that $|e| > 1$. Let $\mathcal{A}_e$ be a circuit arbiter for e, and let $\mathcal{N}'$ be the net $\mathcal{N}$ merged with the circuit arbiter $\mathcal{A}_e$ sharing the transitions in e. If $\mathcal{S}$ is live and bounded then*

1. *A marking $\mathbf{m_0}'$ with $\mathbf{m_0}'[P] = \mathbf{m_0}$ such that $\mathcal{S}' = \langle \mathcal{N}', \mathbf{m_0}' \rangle$ is live and bounded exists.*
2. $\text{rank}(\mathbf{C}') = \text{rank}(\mathbf{C}) + |e| - 1$

Proof (of Lemma 47). For Part 1, boundedness of $\mathcal{S}$ and conservativeness of $\mathcal{A}_e$ guarantee boundedness of $\mathcal{S}'$ for every $\mathbf{m_0}'$ with $\mathbf{m_0}'[P] = \mathbf{m_0}$. Since $\mathcal{S}$ is live and bounded, then the number $r_e = \max\{\min\{\#(e, \sigma) \mid \sigma t \in \text{L}(\mathcal{N}, \mathbf{m})\} \mid t \in T \wedge \mathbf{m} \in \text{RS}(\mathcal{S})\}$ is well-defined. This is a bound for the number of firings of transitions in e that are *required* to enable an arbitrary transition from an arbitrary reachable marking. We put r_e tokens in each place in P_e, what completes the definition of $\mathbf{m_0}'$ and now we prove that $\mathcal{S}'$ is live. Let $\mathbf{m}' \in \text{RS}(\mathcal{S}')$ and $t \in T$. We shall prove that t can ultimately be enabled from $\mathbf{m}'$. We claim that there exists a marking $\mathbf{m}'' \in \text{RS}(\mathcal{N}', \mathbf{m}')$ such that $\mathbf{m}''[P_e] = \mathbf{m_0}'[P_e]$. In that case, since (1) $\mathcal{S}$ is live, (2) $\mathbf{m}''[P] \in \text{RS}(\mathcal{S})$, and (3) $\mathbf{m_0}'[P_e]$ has been defined in a way that it does not interfere when firing a sequence to enable an arbitrary t from an arbitrary reachable marking, then we can fire in $\langle \mathcal{N}', \mathbf{m}'' \rangle$ the same sequence that we could fire in $\langle \mathcal{N}, \mathbf{m}''[P] \rangle$ in order to enable t. To prove the claim, let $\sigma_e = e_{i_1} e_{i_2} \cdots e_{i_k} \in \text{L}(\mathcal{A}_e, \mathbf{m}'[P_e])$ be such that $\mathbf{m}'[P_e] \xrightarrow{\sigma_e} \mathbf{m_0}'[P_e]$, i.e., a sequence in the circuit arbiter returning to the initial marking. It is easy to see that a sequence such that its projection on e is σ_e can be fired in $\langle \mathcal{N}', \mathbf{m}' \rangle$. The idea is firing transitions not in e, which does not affect the marking of places in P_e, until e are P-enabled (their input places in P have enough tokens, no matter how many tokens are there in other places), which will eventually happen thanks to liveness of $\langle \mathcal{N}, \mathbf{m}'[P] \rangle$, then firing e_{i_1} which is also P_e-enabled according to our definition of σ_e, then firing more transitions not in e until e are P-enabled again, then firing e_{i_2} which is also P_e-enabled, etc.

To prove Part 2, for $\text{rank}(\mathbf{C}') = \text{rank}(\mathbf{C}) + |e| - 1$, we shall prove that $|e| - 1$ out of the $|e|$ rows corresponding to the places of the circuit arbiter are linearly independent. Let us fix a notation for the equal conflict set and the circuit arbiter (see Figure 18):

- $e = \{e_0, e_1, \ldots, e_k, \ldots, e_{|e|-1}\}$
- $P_e = \{c_0, c_1, \ldots, c_k, \ldots, c_{|e|-1}\}$
- $\mathbf{Pre}_e[c_i, e_i] = 1$ and $\mathbf{Post}_e[c_{i \oplus 1}, e_i] = 1$ (otherwise zeros), where $\oplus$ represents the sum modulo $|e|$.

It is clear that there is one row being a linear combination of the rest, for instance $\mathbf{C}[c_0, T] = -\sum_{p \in P_e \setminus \{c_0\}} \mathbf{C}[p, T]$, so we remove it and then we prove

that the rows corresponding to places in $P_e \backslash \{c_0\}$ are all linearly independent. Assume, on the contrary, that c_k, where $1 \leq k \leq |e| - 1$, is a linear combination of the other places (let *the other places* be denoted by $OP = P \cup P_e \backslash \{c_0, c_k\}$):

$$\mathbf{C}[c_k, T] = \sum_{p \in OP} \boldsymbol{\lambda}[p] \cdot \mathbf{C}[p, T] = \boldsymbol{\lambda} \cdot \mathbf{C}[OP, T] \tag{62}$$

Thus, the marking increment produced by a sequence σ should also be a linear combination of the marking increment of the other places:

$$\Delta\mathbf{m}[c_k] = \mathbf{C}[c_k, T] \cdot \boldsymbol{\sigma} \stackrel{by\ (62)}{=} \boldsymbol{\lambda} \cdot \mathbf{C}[OP, T] \cdot \boldsymbol{\sigma} = \boldsymbol{\lambda} \cdot \Delta\mathbf{m}[OP] \tag{63}$$

Clearly, it is possible to fire in $\mathcal{S}$ a sequence σ such that $\#(e_i, \sigma) =$ **if** $i <$ k **then** ω **else** 0, where ω is arbitrarily large. In that case $\Delta\mathbf{m}[c_k] = \mathbf{C}[c_k, T] \cdot$ $\boldsymbol{\sigma} = \omega$ is arbitrarily large, while all the entries in $\Delta\mathbf{m}[OP]$ are finite, what contradicts (63). □

Proof (of Theorem 45). Only the rank condition needs to be proven. Let $\mathcal{N}'$ be the net $\mathcal{N}$ together with circuit arbiters merged to *every* non-trivial equal conflict set. Applying Lemma 47.2 repeatedly after each circuit arbiter is merged, what can be done thanks to Lemma 47.1, it follows that:

$$|T| - 1 \geq \text{rank}(\mathbf{C}') = \text{rank}(\mathbf{C}) + \sum_{e \in \text{SEQS}} (|e| - 1)$$

Rearranging the above inequality we obtain a bound for the rank:

$$\text{rank}(\mathbf{C}) \leq |T| - \sum_{e \in \text{SEQS}} (|e| - 1) - 1$$

Since $\sum_{e \in \text{SEQS}} |e| = |T|$, this bound is $|\text{SEQS}| - 1$, so the result follows. □

In the example of Figure 17 (a), Theorem 45 allows to decide that the net cannot be lively and boundedly marked. Both nets in Figure 17 (b) and (c) "pass" the test of Theorem 45, although only the latter can be lively and boundedly marked, e.g., with the marking shown. In summary, the rank condition in Theorem 45 allows indeed to discriminate some cases but, unfortunately, not all of them, that is, it is not sufficient.

In the next subsection we seek for cases where the sufficiency holds, but before let us briefly discuss on the kind of situations leading to non structural liveness that the rank theorem detects. Intuitively, from the proof of Theorem 45, it becomes apparent that the rank condition fails when some individual "choices between alternatives" are not independent from the rest of the system (synchronisations, other choices, etc.). Apart from "flow problems" (i.e., absence of consistency), absence of such independence indicates that a wrong decision taken in an individual choice may affect the rest of the system (to the point of "killing" it).

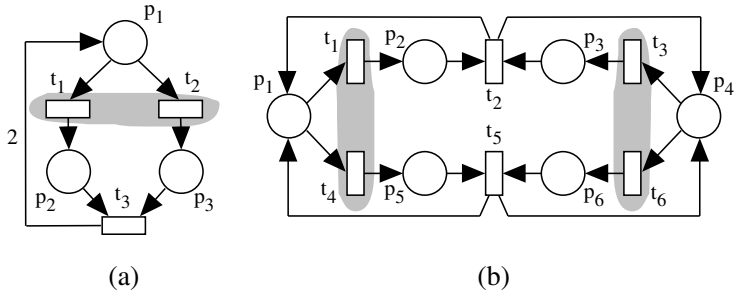

Fig. 19. Two conservative and consistent nets where the rank theorem detects non structural liveness. The (non-trivial) equal conflicts are shaded.

Let us illustrate these ideas with a couple of simple examples where the rank theorem detects non structural liveness. In the net of Figure 19 (a), the only minimal T-semiflow is $\mathbf{1}$, and it is also a basis of T-flows, so $\text{rank}(\mathbf{C}) = 2 = |\text{SEQS}|$. Notice that the fact that t_1 and t_2 are together in every T-semiflow — what is due to the synchronisation or join transition t_3 — means that in every infinite sequence they should be fired in a fixed proportion (one to one in this case). Nevertheless, since the choice between t_1 and t_2 is free, the net does not prevent that this proportion is violated. This mismatch between conflicts and synchronisations is what the rank theorem detects. Observe that if we merge a circuit arbiter on t_1 and t_2, say c_0 from t_2 to t_1 and c_1 from t_1 to t_2, the rank is not increased: one place is clearly a linear combination of the other, say $\mathbf{C}[c_0, T] = -\mathbf{C}[c_1, T]$; but also c_1 is a linear combination of other places, namely $\mathbf{C}[c_1, T] = \mathbf{C}[p_1, T] + 2\mathbf{C}[p_2, T]$, what reveals the problem. In terms of implicit places, both places of the circuit arbiter are structurally implicit, so they can be made implicit with a large enough initial marking. Implicitness of the arbiter reveals that the choice is not free. (Remarkably, for $\mathbf{m_0} = 2p_1$, when the arbiter is marked with only one token the system with the arbiter is live. Notice that in such case the arbiter places are *not* implicit, actually they affect the behaviour avoiding the deadlocks. Increasing the marking of the arbiter places eventually makes them implicit, while it destroys liveness — liveness is not monotonic.)

In the net of Figure 19 (b), the minimal T-semiflows are $t_1 + t_2 + t_3$ and $t_4 + t_5 + t_6$, and they are also a basis of T-flows, so $\text{rank}(\mathbf{C}) = 4 = |\text{SEQS}|$. Now the synchronisations (t_2 and t_5) do not impose a given resolution of each conflict to allow infinite activity (the outcomes of each conflict are in different minimal T-semiflows), but they impose that each conflict is solved according to the other, what is again not guaranteed by the net structure where the choices are free. The rank theorem detects also this mismatch. If we merge a circuit arbiter, say on t_1 and t_4, it increases the rank. Now the net with the arbiter has a unique minimal T-semiflow, $\mathbf{1}$, and the second circuit arbiter does not increase the rank, in the same way as in the previous example (after merging an arbiter

on one conflict, a proportion between the outcomes of the other conflict has been fixed).

6.2 The Rank Theorem for Some Subclasses

For certain subclasses, the general necessary condition for structural liveness (Theorem 45) has been proven to be sufficient too. Loosely speaking, these subclasses have in common that their syntactical constraints leave only conflicts that are essentially "choices between alternatives" (equal conflicts), so representing competition or resource sharing is very limited when not forbidden. This is particularly the case of equal conflict systems (it is also trivially the case of state machines or marked graphs):

Theorem 48. *Let $\mathcal{N}$ be an equal conflict net.*

A marking $\mathbf{m_0}$ such that $\mathcal{S}$ is live and bounded exists iff $\mathcal{N}$ is strongly connected, consistent, and $\mathrm{rank}(\mathbf{C}) = |\mathrm{SEQS}| - 1$.

Proof (Sketch). The necessity part is the general rank theorem, where $\mathrm{rank}(\mathbf{C}) < |\mathrm{SEQS}|$ reduces to $\mathrm{rank}(\mathbf{C}) = |\mathrm{SEQS}| - 1$ because a live and bounded equal conflict net where circuit arbiters have been merged to every equal conflict set has a unique minimal T-semiflow. For existence of this T-semiflow, notice that the "arbitered" net must be consistent. For unicity, it suffices to show that the support of every T-semiflow is the whole T. Let $\mathbf{x}$ be a T-semiflow of the arbitered net, and let $t \in \|\mathbf{x}\|$. All the transitions in $\mathrm{CCS}(t)$ are also in $\|\mathbf{x}\|$ because the places in the circuit arbiters have only one output transition. Every output place of the transitions in $\mathrm{CCS}(t)$ must have at least one output transition in $\|\mathbf{x}\|$, so we can apply repeatedly the same argument and, by strong connectedness, all the transitions are shown to be in $\|\mathbf{x}\|$.

The sufficiency part requires a closer investigation of the structural properties of the net that is out of the scope of this work, but we outline here the proof, referring to [92]. Consistency and $\mathrm{rank}(\mathbf{C}) = |\mathrm{SEQS}| - 1$ imply *P-allocatability* (see [92, Theorem 20]). Strong connectedness and P-allocatability imply conservativeness (see [92, Theorem 24]), hence boundedness for every initial marking. Moreover, strong connectedness and P-allocatability imply that liveness of the whole net is guaranteed by liveness of every *P-component* (see [92, Theorem 27.2]). A marking $\mathbf{m_0}$ such that $\mathbf{m_0}[p] = \mathbf{Pre}[p, t]$ for every p, where $t \in p^\bullet$, is sufficient to make every P-component live. □

From the proof of the sufficiency part, it follows in particular that live and bounded equal conflict systems are structurally bounded, thus conservative taking consistency into account:

Corollary 49. *If $\mathcal{S} = \langle \mathcal{N}, \mathbf{m_0} \rangle$ is a live equal conflict system, then $\mathcal{S}$ is bounded iff $\mathcal{N}$ is conservative (hence structurally bounded).*

In the case of free choice nets [39], the P-components are strongly connected state machine P-subnets. Strongly connected state machines are live iff they

are marked, so in the ordinary case the above statement can be made stronger, particularly showing the polynomial complexity of the liveness and boundedness problem for free choice systems:

Corollary 50. *A free choice system $\mathcal{S}$ is live and bounded iff $\mathcal{N}$ is strongly connected, consistent, and* $\text{rank}(\mathbf{C}) = |\text{SEQS}| - 1$*, and no P-semiflow* $\mathbf{y}$ *such that* $\mathbf{y} \cdot \mathbf{m_0} = 0$ *exists.*

Remark 51. Some results from the classical free choice theory can be deduced easily from the above rank theorem, particularly the *duality theorem* [39], which states that a free choice net is structurally live and structurally bounded iff its reverse-dual net (which is also free choice) is. □

The rank based characterisation of structural liveness and boundedness has been extended to larger subclasses of nets. In particular, for the class of *deterministically synchronised sequential processes (DSSP)* [71,74], that is intended for the modular modelling of sequential agents that cooperate through buffers, the corresponding result is proven in [71].

A DSSP is a net system formed by a collection of *sequential agents* interconnected in a restricted way through *buffers*. The sequential agents are live and safe state machines. The buffers are places whose outputs are in one sequential agent (i.e., buffers are *destination private*) and that do not condition the resolution of the conflicts of their destination (i.e., all the outcomes of a conflict in a sequential agent have the same precondition). Under interleaving semantics, DSSP are a *strict generalisation* of equal conflict systems. In other words, provided that only sequential observations are relevant, equal conflict systems can be *simulated* by DSSP. The construction is simple (see Figure 20): add self-loop

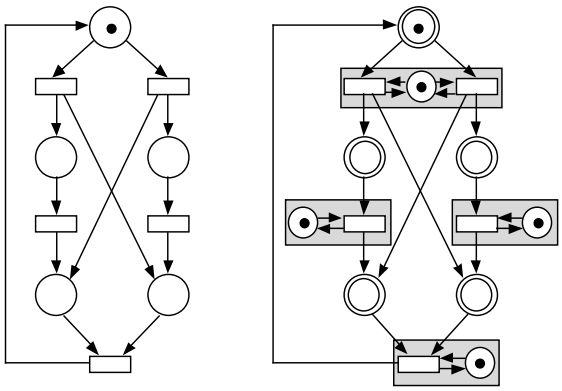

Fig. 20. Simulation of equal conflict systems by DSSP.

places marked with one token around each equal conflict set of a given equal conflict system. These self-loop places (with their adjacent transitions) are the

sequential agents, and the original places of the equal conflict system play the role of buffers.

Other results for equal conflict systems have been extended to DSSP (see [74]), including the equivalence of liveness and deadlock-freeness (under boundedness and strong connectedness), the existence of home states (under liveness and boundedness), or the absence of spurious deadlocks (under liveness and consistency). Therefore, in particular, it is possible to verify liveness using integer programming, the same as in equal conflict systems (the deadlock-freeness condition can be written as a single system of linear inequalities preserving the class membership also in this case using a particular transformation rule).

Extending the DSSP definition recursively, the class of {SC}*EQS is defined, for which also a rank based characterisation of structural liveness and boundedness holds [72].

6.3 Application of the Rank Theorems for Subclasses to General Nets

Given a P/T net for which the available rank theorems do not allow to decide on its structural liveness, e.g., the nets in Figure 17 (b,c), it is sometimes possible, using certain net *transformation rules* (e.g., removal of bypass transitions or implicit places, other classical reduction rules, equalisation, release, etc.) to obtain another net where the corresponding property preservation of the rules together with the available results allow to decide. This topic deserves a closer investigation that is out of the scope of this work (see [73]), but we illustrate it with some examples.

In the net of Figure 17 (b), transition t_3 is a linear combination of t_2 and t_4: $\mathbf{C}[P, t_3] = \mathbf{C}[P, t_2] + \mathbf{C}[P, t_4]$, so the effect of firing t_3 is the same as the effect of firing t_2 and t_4. Moreover, ${}^\bullet\{t_2, t_4\} \cap \{t_2, t_4\}^\bullet = \emptyset$, so t_3 can only be fired when both t_2 and t_4 can be fired in one step (t_3 is a particular case of *bypass* transition, its occurrence "bypasses" the occurrence of the step $t_2 + t_4$ or any of its interleavings). Clearly, by removing t_3 from Figure 17 (b) we could not destroy liveness, i.e., liveness with t_3 ensures liveness without it. But in the net that we obtain after removing t_3, $\text{rank}(\mathbf{C}) = 3$ and $\text{SEQS} = \{\{t_1\}, \{t_2, t_5\}, \{t_4, t_6\}\}$, so Theorem 45 shows that it cannot be lively and boundedly marked, hence the net of Figure 17 (b) is proven not to be structurally live.

In the net of Figure 17 (c), the path $p_3 \to t_5 \to p_5$ can be substituted by a (macro)place p_{35}. This place is implicit, hence it can be removed. Since the resulting net can be proven (structurally) live, so it is the original one — actually, in this case we need not apply the rank theorem, since the net is simply a state machine.

Similarly, in the conservative and consistent net of Figure 3, where $\text{rank}(\mathbf{C}) = 9 = |\text{SEQS}| - 1$ (i.e., the necessary condition for structural liveness holds), place R is structurally implicit. Since the removal of R leads to a net where structural liveness can be proven, we are done — again in this case we need not apply the rank theorem, since the net without R is simply a marked graph.

A particular transformation, called *equalisation*, allows to obtain a general sufficient condition for structural liveness and boundedness. *Total equalisation* of a net consists in adding arcs or increasing weights as needed to make every coupled conflict set equal without changing $\mathbf{C}$ (if we add an arc from p to t we must add another from t to p). Figure 21 shows a net and the net obtained

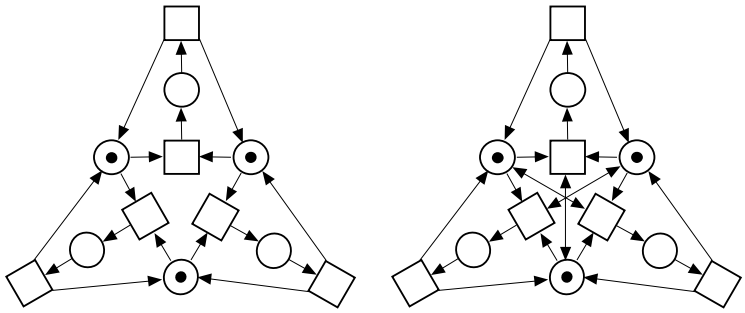

Fig. 21. Total equalisation of a net.

by total equalisation. The resulting net is equal conflict; if it can be proven structurally live and bounded (using Theorem 48), then the original net is proven structurally live and bounded too:

Theorem 52. *Let $\mathcal{N}$ be a P/T net.*

If $\mathcal{N}$ is strongly connected, consistent, and $\mathrm{rank}(\mathbf{C}) = |\mathrm{SCCS}| - 1$*, then* $\mathbf{m_0}$ *such that $\mathcal{S}$ is live and bounded exists.*

Proof. After total equalisation we obtain an equal conflict net $\mathcal{N}'$ such that $\mathbf{C}' = \mathbf{C}$ and $\mathrm{SEQS}' = \mathrm{SCCS}' = \mathrm{SCCS}$. It follows that $\mathcal{N}'$ is strongly connected, consistent, and $\mathrm{rank}(\mathbf{C}') = |\mathrm{SEQS}'| - 1$, so, by Theorem 48, a marking $\mathbf{m_0}$ exists such that $\langle \mathcal{N}', \mathbf{m_0} \rangle$ is live and bounded. Since live and bounded equal conflict systems are conservative (see Corollary 49), $\mathcal{N}'$ is conservative too, hence so it is $\mathcal{N}$, and then $\mathcal{S}$ is bounded. Assume $\mathcal{S}$ non live. Then $t \in T$ and $\mathbf{m}_t \in \mathrm{RS}(\mathcal{S})$ exist such that t cannot be fired from any $\mathbf{m} \in \mathrm{RS}(\mathcal{N}, \mathbf{m}_t)$. Clearly, $\mathbf{m}_t \in \mathrm{LRS}^{\mathrm{SE}}(\mathcal{N}', \mathbf{m_0})$, thus a marking $\mathbf{m}_1$ exists such that $\mathbf{m}_1 \in \mathrm{RS}(\mathcal{N}', \mathbf{m}_t) \cap \mathrm{RS}(\mathcal{N}', \mathbf{m_0})$ (see Subsection 3.4). Since $\langle \mathcal{N}', \mathbf{m_0} \rangle$ is live, $\mathbf{m}_2 \in \mathrm{RS}(\mathcal{N}', \mathbf{m}_1)$ exists such that t is enabled. Contradiction, since $\mathbf{m}_2 \in \mathrm{RS}(\mathcal{N}, \mathbf{m}_t)$. □

In the case of ordinary nets, basically the same argument allows to make use of Corollary 50, leading to:

Corollary 53. *Let $\mathcal{S}$ be an ordinary P/T system.*

If $\mathcal{N}$ is strongly connected, consistent, and $\mathrm{rank}(\mathbf{C}) = |\mathrm{SCCS}| - 1$*, and no P-semiflow* $\mathbf{y}$ *such that* $\mathbf{y} \cdot \mathbf{m_0} = 0$ *exists, then $\mathcal{S}$ is live and bounded.*

The above result(s) allow, for instance, to prove (structural) liveness and boundedness of the system (net) in Figure 21.

In summary, given an arbitrary P/T net $\mathcal{N}$ that is strongly connected and consistent (otherwise it cannot be lively and boundedly marked), only when $|\text{SEQS}| - 1 \geq \text{rank}(\mathbf{C}) \geq |\text{SCCS}| - 1$ a marking $\mathbf{m_0}$ such that $\mathcal{S}$ is live and bounded exists, what is guaranteed in case $\text{rank}(\mathbf{C}) = |\text{SCCS}| - 1$. (The inequality $\text{rank}(\mathbf{C}) \geq |\text{SCCS}| - 1$ can be deduced as follows: given $\mathcal{N}$ strongly connected and consistent, by total equalisation we obtain an equal conflict net $\mathcal{N}'$ that is strongly connected and consistent too. Merging arbiters in every equal conflict set as in the proof of Theorem 48, the resulting net $\mathcal{N}''$ may have either one minimal T-semiflow or none. Being consistent, a basis of T-flows can be made up of T-semiflows only — dual of Proposition 12.1 — so the dimension of the space of right annullers of $\mathbf{C}''$ is at most one.)

7 Bibliographical Remarks

Linear algebra has been used in net theory at different net levels (e.g., P/T, or high level) and with different purposes (e.g., logical analysis, performance evaluation, controller design, or net synthesis). In logical/correctness analysis — which is the topic of this paper — other properties not considered here have also been studied (e.g., home states [44], or fairness [84]). For *performance evaluation and optimization*, linear techniques have been applied for the computation of performance bounds [78,69,17], for approximation techniques [14], or for initial marking optimization [15]. For the design of logic controllers for a plant modelled with P/T, linear techniques have been applied within the so called *supervisory control* theory [68] (e.g., [38,53,54,40]). The synthesis of P/T systems from an automata using the *theory of regions* applies also linear algebraic techniques [2]. In the case of high level PN, linear algebraic techniques have been developed mainly for the computation of P- (and T-) invariants (most relevant works are collected in [43]).

The use of integer linear algebra for the correctness analysis of PN dates back to the seventies [52], where the invariant method is introduced. Other pioneering works, using linear algebra in the real domain, introduce the notions of consistency and conservativeness [70,55]. The alternatives theorem (or Farkas lemma) is applied in [59,79] to provide dual perspectives of structural boundedness and repetitiveness, laying a first bridge between net theory and convex geometry. Taking into account that in live marked graphs reachability is linearly characterised, and that the incidence matrix of a marked graph is unimodular, linear programming can be used for the analysis, as it was firstly done in [36]. After realising that the evolution equation of a net system is a state equation in control theory sense, [63] tries to lay a bridge between nets and classical linear control theories. Despite the great conceptual interest of this bridge, integrality and non-negativity constraints, and the existence of spurious solutions, limit its strength.

The topic was surveyed in both previous Advanced Courses on PN: In [60] emphasis is given to boundedness, repetitiveness, and duality, [51] is an introductory tutorial on linear algebraic techniques for P/T nets, and [61] mainly overviews the invariant method for high level nets.

The basic idea in early works was the intensive exploitation of the P- (and T-) invariants — the so called invariant method — focusing on minimality, decomposition, and applicability to prove properties. In essence, the idea is to find the appropriate invariants to (dis)prove the interesting properties, for what the computation of the fundamental set is important. A need for this kind of computations has been encountered — and solutions re-discovered — quite often in several disciplines (dating back to Fourier! see [22] for more details). Within the PN field, [57] gives a first algorithm to compute the fundamental set of semiflows, taking advantage from a rank based property to remove non minimal semiflows before their computation is completed. In order to reduce the computational complexity of the algorithm, some heuristics have been proposed (e.g., [57,1]). In [22] the interpretation of semiflows as directions of a cone is explicited, and the existing algorithms are reviewed, improved, and their performance is analysed. Taking into account integrality constraints reveals that $\mathbf{y} \cdot \mathbf{C} = \mathbf{0}$ is an homogeneous linear Diophantine system, and different solutions are investigated in [48]. Actually, in principle, the state equation comprises integrality constraints. If only integer solutions are to be considered to prove non-reachability of a given marking, then we should solve it using integer linear algebraic methods, e.g., by means of the Smith normal form [45,76]. Integrality constraints in linear equation systems can be treated from a modulo-arithmetic perspective. By applying this approach to the state equation, modulo-invariants are obtained in [32], generalising the notion of P-flows.

Traps and siphons, which lead to different marking invariants, have been extensively used in the structure theory of (mainly ordinary) P/T, particularly in the case of some net subclasses (e.g., [39,41,10,4,31,5]). Regarding the computation of traps and siphons using linear algebraic techniques, the initial attempts try to translate the logic conditions defining the corresponding objects into a set of linear inequalities (e.g. [1,81]). A new line of thinking was opened in [50], where the computation of traps and siphons is carried out through the computation of semiflows in a transformed net. This approach was used with slight improvements in [33]. A similar approach, where only the weighting of the problem net is possibly transformed, was introduced in [35]. (A particular instance of this method has been presented here.)

In summary, the classical method consists in computing some structural objects (semiflows, traps, etc.) and then using the corresponding invariants in order to prove properties. The point of view adopted here considers directly the state equation, possibly improved, e.g., by taking into account the information provided by traps. This method was introduced in [83], where the analysis of several synchronic properties of general P/T systems is carried out through linear programming problems based on the state equation, and it was developed in detail in [19]. The idea of using implicit places to cut spurious solutions and

their relation with trap invariants was presented in [23]. In [87] the method is implemented, and it is observed that the iterative application can remove more spurious solutions because the added implicit places lead to new traps. The improvement of the linear description is not only helpful for correctness analysis, but also for performance evaluation [16]. Monitor places have been used in order to forbid reaching some markings within supervisory control theory [38,93]. The observation that they could be used also to remove given spurious solutions, and that this method allows to remove all of them in structurally safe systems appears in [21]. The idea of incorporating a generator of trap invariants into the state equation appears in [58]. Also in [23] a totally different improvement method is introduced, consisting in removing spurious solutions without predecessors, for what partial enumeration is required (by the way, this method removes the spurious solutions of the example system in Figure 14 with $q = r = 2$).

Implicit places were introduced in [6]. Actually, in this seminal work, only implicit places the marking equation of which is redundant were considered, i.e., $\mu \geq 0$, and they were called redundant places. It was observed in [81] that redundancy (in a convex geometry sense) is not necessary for the place to be implicit (in the sense that it does not affect the behaviour), i.e., allowing $\mu < 0$. Implicit places were revisited in [23], where the structural ones were derived, using duality theory, from the linear inequalities expressing the condition that the behaviour is not affected. Moreover, a sufficient condition for a place to be implicit in terms of a linear programming problem was introduced. Besides their interest for reduction techniques and improvement of the state equation, implicit places play an important role in implementation techniques. On one hand, the addition of implicit places increases the Hamming distance of the code defined by marking vectors, what is interesting for fault-tolerant (error detecting and correcting) implementations [80,85,86]. On the other hand, since new semiflows appear after the addition of implicit places, the possibilities for decomposition are increased, what is useful for distributed implementations [81,24]. The new possibilities for decomposition are also interesting for approximate performance evaluation techniques [14,66] and exact performance evaluation [18].

Non-existence of solution to the $|T|$ linear systems of equations for t dead of the form (40) is essentially the necessary condition for liveness presented in [60,49]. In this sense, non-existence of solution to (37) is a greatly more accurate necessary condition for liveness. Some techniques to improve the performance of verifying non-existence of solution to (37) were presented in [89], and they have been recalled and improved here.

It was early realised that the state equation (or the invariants that can be deduced from it) is in general insufficient to analyse liveness or similar properties. Quoting from [51]: "Token counting in P-semiflows is by far not subtle enough to solve general liveness problems." One way of approaching the problem from structure theory is to investigate the conditions under which the net structure allows a live marking, i.e., structural liveness. Presently, the best linear conditions are given by the so called rank theorems. The rank theorem for free choice systems was conceived from the problem of computability of visit ratios

in stochastic free choice nets in [13]. A proof, based on the Commoner's theorem [39], so limited to the free choice case, was published in [28]. In order to extend the applicability of the result, the necessity part was developed for general P/T nets [20], and the sufficiency part was developed for equal conflict systems [91], independently of the classical free choice theory. (Both results, that have been recalled here with minor modifications, can be found in [92].) This allowed to obtain rank based characterisations of structural liveness and boundedness in larger subclasses, namely DSSP [71] and {SC}*EQS [72]. The idea of using the rank theorem for free choice — applying equalisation — to obtain a general sufficient condition for liveness and boundedness in ordinary P/T systems (actually, to define a subclass, the *regular marked nets*, that are always live and bounded) appears in [29]. The extension to general P/T systems is contained in [71]. Generalising this approach by means of other transformation rules increases the decision power of rank theorems [73].

Acknowledgements

The authors wish to thank the careful reading and valuable suggestions of E. Badouel, J. Ezpeleta, C. Girault, K. Lautenbach, T. Murata, L. Ojala, L. Pomello, and L. Recalde.

References

1. H. Alaiwan and J. M. Toudic. Recherche des semi-flots, des verrous et des trappes dans les réseaux de Petri. *Technique et Science Informatiques*, 4(1):103–112, 1985.
2. E. Badouel and P. Darondeau. A survey on net synthesis. In Borne et al. [9], pages 309–316.
3. Z. A. Banaszak and B. H. Krogh. Deadlock avoidance in flexible manufacturing systems with concurrently competing process flows. *IEEE Trans. on Robotics and Automation*, 6(6):724–734, 1990.
4. K. Barkaoui and M. Minoux. A polynomial-time graph algorithm to decide liveness of some basic classes of bounded Petri nets. In Jensen [42], pages 62–75.
5. K. Barkaoui and J. F. Pradat-Peyre. On liveness and controlled siphons in Petri nets. In Billington and Reisig [8], pages 57–72.
6. G. Berthelot and G. Roucairol. Reduction of Petri netss. In *Procs. of the Symposium on MFCS '76*, volume 45 of *Lecture Notes in Computer Science*, pages 202–209. Springer, 1976.
7. E. Best, J. Desel, and J. Esparza. Traps characterize home states in free choice systems. *Theoretical Computer Science*, 101:161–176, 1993.
8. J. Billington and W. Reisig, editors. *Application and Theory of Petri Nets 1996*, volume 1091 of *Lecture Notes in Computer Science*. Springer, 1996.
9. P. Borne, J. C. Gentina, E. Craye, and S. El Khattabi, editors. *Symposium on Discrete Events and Manufacturing Systems. CESA '96 IMACS Multiconference*, Lille, France, July 1996.
10. G. W. BRAMS. *Réseaux de Petri: Théorie et Pratique*. Masson, 1983.
11. W. Brauer, editor. *Net Theory and Applications*, volume 84 of *Lecture Notes in Computer Science*. Springer, 1980.

12. W. Brauer, W. Reisig, and G. Rozenberg, editors. *Petri Nets: Central Models and their Properties. Advances in Petri Nets 1986, Part I*, volume 254 of *Lecture Notes in Computer Science*. Springer, 1987.
13. J. Campos, G. Chiola, and M. Silva. Properties and performance bounds for closed free choice synchronized monoclass queueing networks. *IEEE Trans. on Automatic Control*, 36(12):1368–1382, 1991.
14. J. Campos, J. M. Colom, H. Jungnitz, and M. Silva. Approximate throughput computation of stochastic marked graphs. *IEEE Trans. on Software Engineering*, 20(7):526–535, 1994.
15. J. Campos, J. M. Colom, and M. Silva. Performance evaluation of repetitive automated manufacturing systems. In *Procs.* 2^{nd} *Int. Conf. on Computer Integrated Manufacturing and Automation Technology (CIMAT '90)*, pages 74–81. IEEE-Computer Society Press, 1990.
16. J. Campos, J. M. Colom, and M. Silva. Improving throughput upper bounds for net based models. In S. G. Tzafestas and J. C. Gentina, editors, *Robotics and Flexible Manufacturing Systems*, pages 281–294. Elsevier, 1992.
17. J. Campos and M. Silva. Structural techniques and performance bounds of stochastic Petri net models. In G. Rozenberg, editor, *Advances in Petri Nets 1992*, volume 609 of *Lecture Notes in Computer Science*, pages 352–391. Springer, 1992.
18. J. Campos, M. Silva, and S. Donatelli. Structured solution of stochastic DSSP systems. In *Procs. of the* 7^{th} *Int. Workshop on Petri Nets and Performance Models (PNPM97)*. IEEE Computer Society Press, 1997.
19. J. M. Colom. *Análisis Estructural de Redes de Petri. Programación Lineal y Geometría Convexa*. PhD thesis, DIEI. Univ. Zaragoza, June 1989.
20. J. M. Colom, J. Campos, and M. Silva. On liveness analysis through linear algebraic techniques. In *Procs. of the AGM of Esprit BRA 3148 (DEMON)*, 1990.
21. J. M. Colom et al. Linear algebraic characterisation of structurally safe P/T systems. Technical report, DIIS. Univ. Zaragoza, 1997. In preparation.
22. J. M. Colom and M. Silva. Convex geometry and semiflows in P/T nets. A comparative study of algorithms for computation of minimal P-semiflows. In Rozenberg [75], pages 79–112.
23. J. M. Colom and M. Silva. Improving the linearly based characterization of P/T nets. In Rozenberg [75], pages 113–145.
24. J. M. Colom, M. Silva, and J. L. Villarroel. On software implementation of Petri nets and colored Petri nets using high-level concurrent languages. In *Proc.* 7^{th} *European Workshop on Application and Theory of Petri Nets*, pages 207–241, Oxford, England, July 1986.
25. F. Commoner, A. W. Holt, S. Even, and A. Pnueli. Marked directed graphs. *Journal on Computer Systems Science*, 5:72–79, 1971.
26. R. G. Coyle. *Management System Dynamics*. Wiley, 1977.
27. R. David and H. Alla. *Petri Nets and Grafcet*. Prentice-Hall, 1992.
28. J. Desel. A proof of the rank theorem for extended free choice nets. In Jensen [42], pages 134–153.
29. J. Desel. Regular marked Petri nets. In J. Leeuwen, editor, *WG' 93:* 19^{th} *Int. Workshop on Graph-Theoretic Concepts in Computer Science*, volume 790 of *Lecture Notes in Computer Science*, pages 264–275. Springer, 1993.
30. J. Desel and J. Esparza. Reachability in cyclic extended free choice systems. *Theoretical Computer Science*, 114:93–118, 1993.
31. J. Desel and J. Esparza. *Free Choice Petri Nets*, volume 40 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, 1995.

32. J. Desel, K. P. Neuendorf, and M. D. Radola. Proving nonreachability by modulo-invariants. *Theoretical Computer Science*, 153(1-2):49–64, 1996.
33. J. Esparza and M. Silva. A polynomial time algorithm to decide liveness of bounded free choice nets. *Theoretical Computer Science*, 102:185–205, 1992.
34. J. Ezpeleta, J. M. Colom, and J. Martínez. A Petri net based deadlock prevention policy for flexible manufacturing systems. *IEEE Trans. on Robotics and Automation*, 11(2):173–184, 1995.
35. J. Ezpeleta, J. M. Couvreur, and M. Silva. A new technique for finding a generating family of siphons, traps and ST-components. application to coloured Petri nets. In G. Rozenberg, editor, *Advances in Petri Nets 1993*, volume 674 of *Lecture Notes in Computer Science*, pages 126–147. Springer, 1993.
36. H. J. Genrich and K. Lautenbach. Synchronisationgraphen. *Acta Informatica*, 2:143–161, 1973.
37. H. J. Genrich, K. Lautenbach, and P. S. Thiagarajan. Elements of general net theory. In Brauer [11], pages 21–163.
38. A. Giua, F. DiCesare, and M. Silva. Generalized mutual exclusion constraints on nets with uncontrollable transitions. In *IEEE Int. Conf. on Systems, Man, and Cybernetics*, Chicago, IL, USA, October 1992.
39. M. H. T. Hack. Analysis of production schemata by Petri nets. Master's thesis, M.I.T., Cambridge, MA, USA, 1972. (Corrections in *Computation Structures Note* 17, 1974).
40. L. E. Holloway, B. H. Krogh, and A. Giua. Petri nets for the control of discrete event systems: A tutorial survey. In *Supervisory Control of Discrete Event Systems*. Laboratoire d'Automatique de Grenoble, INPG, September 1995.
41. M. Jantzen and R. Valk. Formal properties of Place/Transition nets. In Brauer [11], pages 165–212.
42. K. Jensen, editor. *Application and Theory of Petri Nets 1992*, volume 616 of *Lecture Notes in Computer Science*. Springer, 1992.
43. K. Jensen and G. Rozenberg, editors. *High-level Petri Nets*. Springer, 1991.
44. C. Johnen. Algorithmic verification of home spaces in P/T systems. In *Procs. IMACS 1988, 12^{th} World Congress on Scientific Computation*, pages 491–493, 1988.
45. R. Kannan and A. Bachem. Polynomial algorithms for computing the Smith and Hermite normal forms of an integer matrix. *SIAM Journal on Computing*, 8:499–507, 1979.
46. P. Kemper. Linear time algorithm to find a minimal deadlock in a strongly connected free-choice net. In M. Ajmone Marsan, editor, *Application and Theory of Petri Nets 1993*, volume 691 of *Lecture Notes in Computer Science*, pages 319–338. Springer, 1993.
47. W. E. Kluge and K. Lautenbach. The orderly resolution of memory access conflicts among competing channel processes. *IEEE Trans. on Computers*, 31(3):194–207, 1982.
48. F. Krückeberg and M. Jaxy. Mathematical methods for calculating invariants in Petri nets. In G. Rozenberg, editor, *Advances in Petri Nets 1987*, volume 266 of *Lecture Notes in Computer Science*, pages 104–131. Springer, 1987.
49. J. B. Lasserre and P. Mahey. Using linear programming in Petri net analysis. *Operations Research*, 26(1):43–50, 1989.
50. K. Lautenbach. Linear algebraic calculation of deadlocks and traps. In K. Voss et al., editors, *Concurrency and Nets*, pages 315–336. Springer, 1987.
51. K. Lautenbach. Linear algebraic techniques for Place/Transition nets. In Brauer et al. [12], pages 142–167.

52. K. Lautenbach and H. A. Schmid. Use of Petri nets for proving correctness of concurrent process systems. In *Procs. IFIP Congress 74*, pages 187–191. North-Holland, 1974.
53. Y. Li and W. M. Wonham. Control of vector discrete event systems I - the base model. *IEEE Trans. on Automatic Control*, 38(8):1214–1227, 1993.
54. Y. Li and W. M. Wonham. Control of vector discrete event systems II - controller synthesis. *IEEE Trans. on Automatic Control*, 39(3):512–531, 1994.
55. Y. E. Lien. Termination properties of generalized Petri nets. *SIAM Journal on Computing*, 5(2):251–265, 1976.
56. D. G. Luenberger. *Introduction to Linear and Non Linear Programming.* Addison Wesley, 1972.
57. J. Martínez and M. Silva. A simple and fast algorithm to obtain all invariants of a generalized Petri net. In C. Girault and W. Reisig, editors, *Application and Theory of Petri Nets*, pages 301–310. Springer, 1982.
58. S. Melzer and J. Esparza. Checking system properties via integer programming. In H. R. Nielsen, editor, *Proceedings of ESOP '96*, volume 1058 of *Lecture Notes in Computer Science*, pages 250–265. Springer, 1996.
59. G. Memmi. Applications of the semiflow notion to the boundedness and liveness problems in Petri net theory. In *Conf. on Information Sciences and Systems.* John Hopkins University, 1978.
60. G. Memmi and G. Roucairol. Linear algebra in net theory. In Brauer [11], pages 213–223.
61. G. Memmi and J. Vautherin. Analysing nets by the invariant method. In Brauer et al. [12], pages 300–336. Collected in [43].
62. T. Murata. Circuit theoretic analysis and synthesis of marked graphs. *IEEE Trans. on Circuits and Systems*, 24(7):400–405, 1977.
63. T. Murata. State equation, controllability, and maximal matchings of Petri nets. *IEEE Trans. on Automatic Control*, 22(3):412–416, 1977.
64. T. Murata, B. Shenker, and S. M. Shatz. Detection of Ada static deadlocks using Petri net invariants. *IEEE Trans. on Software Engineering*, 15(3):314–326, 1989.
65. K. G. Murty. *Linear Programming.* Wiley and Sons, 1983.
66. C. J. Pérez, J. Campos, and M. Silva. On approximate performace evaluation of manufacturing systems modelled with weighted T-systems. In Borne et al. [9], pages 201–207.
67. J. L. Peterson. *Petri Net Theory and the Modeling of Systems.* Prentice-Hall, 1981.
68. P. J. G. Ramadge and W. M. Wonham. The control of discrete event systems. *Proceedings of the IEEE*, 77(1):81–98, 1989.
69. C. V. Ramamoorthy and G. S. Ho. Performance evaluation of asynchronous concurrent systems using Petri nets. *IEEE Trans. on Software Engineering*, 6(5):440–449, 1980.
70. C. Ramchandani. Analysis of asynchronous concurrent systems by Petri nets. Technical Report Project MAC, TR-120, M.I.T., Cambridge, MA, USA, 1974.
71. L. Recalde, E. Teruel, and M. Silva. On well-formedness analysis: The case of deterministic systems of sequential processes. In J. Desel, editor, *Proc. of the Int. Workshop on Structures in Concurrency Theory (STRICT)*, Workshops in Computing, pages 279–293. Springer, 1995.
72. L. Recalde, E. Teruel, and M. Silva. {SC}*ECS: A class of modular and hierarchical cooperating systems. In Billington and Reisig [8], pages 440–459.
73. L. Recalde, E. Teruel, and M. Silva. Improving the decision power of rank theorems. In *IEEE Int. Conf. on Systems, Man, and Cybernetics*, Orlando, Florida, USA, October 1997. To appear.

74. L. Recalde, E. Teruel, and M. Silva. Modeling and analysis of sequential processes that cooperate through buffers. Technical report, DIIS. Univ. Zaragoza, 1997. Submitted paper.
75. G. Rozenberg, editor. *Advances in Petri Nets 1990*, volume 483 of *Lecture Notes in Computer Science*. Springer, 1991.
76. A. Schrijver. *Theory of Linear and Integer Programming*. Wiley, 1986.
77. M. W. Shields. *An Introduction to Automata Theory*. Blackwell Scientific Publications, 1987.
78. J. Sifakis. Uses of Petri nets for performance evaluation. In *Measuring, Modelling, and Evaluating Computer Systems*, pages 75–93. North-Holland, 1977.
79. J. Sifakis. Structural properties of Petri nets. In J. Winkowski, editor, *Mathematical Foundations of Computer Science 1978*, pages 474–483. Springer, 1978.
80. J. Sifakis. Realization of fault-tolerant systems by coding Petri nets. *Design Automation and Fault-Tolerant Computing*, 3(2):93–107, 1979.
81. M. Silva. *Las Redes de Petri: en la Automática y la Informática*. AC, 1985.
82. M. Silva. Introducing Petri nets. In *Practice of Petri Nets in Manufacturing*, pages 1–62. Chapman & Hall, 1993.
83. M. Silva and J. M. Colom. On the computation of structural synchronic invariants in P/T nets. In G. Rozenberg, editor, *Advances in Petri Nets 1988*, volume 340 of *Lecture Notes in Computer Science*, pages 387–417. Springer, 1988.
84. M. Silva and T. Murata. B-fairness and structural b-fairness in Petri net models of concurrent systems. *Journal of Computer and System Sciences*, 44(3):447–477, 1992.
85. M. Silva and S. Velilla. Detección y corrección de errores mediante la codificación de redes de Petri. In *II Simposium Nacional IFAC: Automática en la Industria*, pages 491–500, Zaragoza, Spain, November 1984.
86. M. Silva and S. Velilla. Error detection and correction on Petri net models of discrete event control systems. In *Proc. ISCAS 85*, pages 921–924, 1985.
87. E. Teruel. Programa para la mejora de la descripción lineal de redes de Petri mediante adición de lugares implícitos secantes. Master's thesis, DIEI. Univ. Zaragoza, 1990.
88. E. Teruel, P. Chrzastowski, J. M. Colom, and M. Silva. On weighted T-systems. In Jensen [42], pages 348–367.
89. E. Teruel, J. M. Colom, and M. Silva. Linear analysis of deadlock-freeness of Petri net models. In *Procs. of the* 2^{nd} *European Control Conference*, volume 2, pages 513–518. North-Holland, 1993.
90. E. Teruel, J. M. Colom, and M. Silva. Choice-free Petri nets: A model for deterministic concurrent systems with bulk services and arrivals. *IEEE Trans. on Systems, Man, and Cybernetics*, 27(1):73–83, 1997.
91. E. Teruel and M. Silva. Well-formedness of equal conflict systems. In R. Valette, editor, *Application and Theory of Petri Nets 1994*, volume 815 of *Lecture Notes in Computer Science*, pages 491–510. Springer, 1994.
92. E. Teruel and M. Silva. Structure theory of equal conflict systems. *Theoretical Computer Science*, 153(1-2):271–300, 1996.
93. F. Tricas and J. Martínez. An extension of the liveness theory for concurrent sequential processes competing for shared resources. In *IEEE Int. Conf. on Systems, Man, and Cybernetics*, pages 4119–4124, Vancouver, Canada, October 1995.
94. N. Viswanadham, Y. Narahari, and T. L. Johnson. Deadlock prevention and deadlock avoidance in flexible manufacturing systems using Petri net models. *IEEE Trans. on Robotics and Automation*, 6(6):713–723, 1990.

95. M. C. Zhou, F. DiCesare, and A. A. Desrochers. A hybrid methodology for synthesis of Petri nets for manufacturing systems. *IEEE Trans. on Robotics and Automation*, 8(3):350–361, 1992.

Elements of Linear Programming and Duality Theory

Many text books cover linear programming (see, for instance, [56,65]). Here we recall a few definitions and results that are used in the paper.

Any *linear programming problem* can be written in standard form (possibly requiring the incorporation of *slack variables* to transform inequalities into equations) as:

$$z = \max\{\mathbf{c} \cdot \mathbf{x} \mid \ \mathbf{A} \cdot \mathbf{x} = \mathbf{b} \wedge \mathbf{x} \geq \mathbf{0}\} \tag{64}$$

where $\mathbf{x}$ are the (real valued) variables, $\mathbf{c} \cdot \mathbf{x}$ is the *cost function* to optimise, and $\mathbf{A} \cdot \mathbf{x} = \mathbf{b} \wedge \mathbf{x} \geq \mathbf{0}$ are the *linear constraints*. The computational complexity of linear programming problems is polynomial time. They are usually solved using the *simplex* algorithm, which, among other advantages compared to polynomial algorithms, performs most often in linear time in spite of its worst case exponential complexity.

Depending on the existence of solutions to the linear constraints and the value of the objective function, a linear programming problem can be:

- *Non feasible*: No solution to the linear constraints.
- *Unbounded*: The value of the cost function can be increased arbitrarily.
- *Bounded*: There are *optimal solutions* $\mathbf{x}$ that maximise the value of the cost function.

The *dual* of the *primal* problem (64) is:

$$z' = \min\{\mathbf{b} \cdot \mathbf{y} \mid \ \mathbf{y} \cdot \mathbf{A} \geq \mathbf{c}\} \tag{65}$$

Note that the dual of the primal problem (65) is (64).

The *weak duality* theorem states that, if $\mathbf{x}$ and $\mathbf{y}$ are feasible solutions to (64) and (65), respectively, then $\mathbf{c} \cdot \mathbf{x} \leq \mathbf{b} \cdot \mathbf{y}$.

The *duality* theorem states that, if both (64) and (65) are feasible, then both are bounded and $z = z'$.

The *unboundedness* theorem states that, if only one of (64) or (65) is feasible, then it is unbounded.

These theorems allow to prove the *alternatives* theorem (for homogeneous or non-homogeneous systems). Two out of the many formulations of this theorem are the following:

- One and only one of the following systems is feasible:

$$\mathbf{A} \cdot \mathbf{x} \geq \mathbf{b} \tag{66}$$

$$\mathbf{y} \geq \mathbf{0} \wedge \mathbf{y} \cdot \mathbf{b} > 0 \wedge \mathbf{y} \cdot \mathbf{A} = \mathbf{0} \tag{67}$$

– One and only one of the following systems is feasible:

$$\mathbf{A} \cdot \mathbf{x} \geq \mathbf{0} \wedge \mathbf{x} > \mathbf{0} \tag{68}$$

$$\mathbf{y} \cdot \mathbf{A} \lneq \mathbf{0} \wedge \mathbf{y} \geq \mathbf{0} \tag{69}$$

If $\mathbf{x}$ is restricted to be integer in (64), then it is an *integer programming problem*, which is NP-complete. (A typical algorithm is *branch and bound* based on linear programming.) In the particular case that $\mathbf{A} = [\mathbf{I}\ \mathbf{A}']$, a property that we use is that boundedness of the integer programming problem is equivalent to boundedness of the corresponding linear programming problem where integrality is disregarded, although the optimal value may not coincide.