Analysis of Autonomous Petri Nets with Bulk Services and Arrivals *

Manuel Silva and Enrique Teruel

Dpt. Ing. Eléctrica e Informática. CPS, Univ. Zaragoza
María de Luna 3, E-50015 Zaragoza, Spain
Fax: + 34 76 512932 e-mail: silva@cc.unizar.es

Abstract. The basic modelling features of nets and diverse aspects of the functional analysis of uninterpreted net models are surveyed in this paper. Emphasis is given to structural techniques and the study of restricted non ordinary subclasses, which allow the modelling of systems with bulk services and arrivals.
Keywords: Petri nets, functional analysis, structure theory.

1 Introduction

The mathematical description of reality is one of the basis of the impressive developments of technology. In the case of concurrent systems, a formalism able to adequately mathematize the behaviour of a collection of agents possibly interacting, cooperating and competing in performing their own tasks is crucial. *Petri nets* are a mathematical formalism to deal with concurrent systems, in the same way that differential equations deal with continuous dynamical systems. More precisely, Petri nets are a graph based formalism specially well suited to model and analyze *discrete event dynamic systems (DEDS)* which exhibit parallel evolutions and whose behaviours present synchronization and resource sharing phenomena. They have been used in a wide range of fields, including communication networks [21], computer systems [2, 55] and discrete part manufacturing systems [22, 56].

Although they can represent very complex behaviours, they consist of a few simple objects, relations and rules. A net model consists of two parts: (1) A *net structure*, an inscribed bipartite directed graph, that represents the static part of the system; and (2) a *marking*, representing a distributed state on the structure. The places correspond to the state variables of the system and the transitions to their transformers, or actions. The inscriptions may be very different, leading to different families of nets. If the inscriptions are simply natural numbers, named weights, *Place/Transition (P/T) nets* are obtained. In this case, the weights permit the modelling of bulk services and arrivals, while the marking represents the values of the state variables. A historically and conceptually interesting subclass of P/T nets is obtained when all weights are one. The nets are said to be *ordinary*, and they represent a straightforward but important generalization of

* This work is supported by the projects CICYT ROB-91-0949, and Esprit BRA Project 7269 (QMIPS) and W.G. 6067 (CALIBAN).

automata models. Although ordinary and weighted nets have the same theoretical expressive power, weighted ones produce more compact and natural models of many systems. More elaborate inscriptions on the net structure and marking lead to the so called *High Level Petri Net (HLPN)* formalisms [31]. For finite state space (i.e. bounded) net systems, high level descriptions allow more compact models, but again the theoretical modelling power is the same of ordinary P/T nets. For the purpose of this work, which focuses on basic concepts, the P/T level is considered as the reference level.

The intricate behaviours that can be described by P/T net systems make crucial the consideration of well founded *analysis* and *synthesis* techniques. The efficient treatment of P/T models may be very difficult, but for a few subclasses it is reasonably well understood and efficiently solved. Subclasses can be defined either restricting the behaviour or the structure. A possible way of obtaining *structural subclasses* is restricting the inscriptions (e.g. ordinary nets are a subclass of P/T nets) or the topology, usually aiming at limiting the interplay of conflicts and synchronizations. Historically, subclasses of ordinary nets have received special attention. Powerful results were early obtained for them [17, 28] originating an active branch of net theory [5, 6, 20, 25, 35, 63].

Graphs allow different *interpretations*. Among many others, the following are classic graph interpretations: state diagrams and algorithmic state machines representing sequential switching systems, state transition rate diagrams representing continuous time homogeneous Markov Chains, resource allocation graphs, PERT (Program Evaluation and Review Technique) graphs, etc. An interpretation precises the semantics of objects and their behaviour, eventually making explicit the connection of the model to the external world. As these classic graph models, Petri nets can be interpreted, leading to models describing different perspectives of a given system along its life-cycle. Nevertheless, for the purpose of this work, we mainly consider *autonomous* (i.e. uninterpreted) net models.

This survey is organized as follows: The formal definitions together with the description of some modelling features are contained in Sect. 2. Basic analysis methods are overviewed in Sect. 3. Among them, several structural techniques are further detailed in Sect. 4.

2 Modelling of Systems Using Nets

Formally, a P/T net is an inscribed bipartite directed graph: $\mathcal{N} = (P, T, F, W)$, where P and T are the disjoint sets of places and transitions, $F \subseteq (P \times T) \cup (T \times P)$ defines the support of the flow relation, and $W : F \mapsto \mathbb{N}$ are the inscriptions. The functions $Pre, Post : P \times T \mapsto \mathbb{N}$ can be defined as $Pre(p,t) = \textbf{if } (p,t) \in F \textbf{ then } W(p,t) \textbf{ else } 0$, and $Post(p,t) = \textbf{if } (t,p) \in F \textbf{ then } W(t,p) \textbf{ else } 0$. Thus, a P/T net can be alternatively defined as $\mathcal{N} = (P, T, Pre, Post)$. The incidence functions are usually represented by matrices. If, without loss of generality, $\forall p \in P, t \in T : Pre[p,t] \cdot Post[p,t] = 0$, then the incidence matrix, $C = Post - Pre$, contains all the flow information. A P/T system is a pair $(\mathcal{N}, M_0)$ where $\mathcal{N}$ is a P/T net and $M_0 : P \mapsto \mathbb{N}$ is the initial marking, that can be represented in vector form. Transition t is enabled at marking M iff $M \geq Pre[\cdot, t]$. Being enabled it may fire, yielding a reached marking $M' = M + Post[\cdot, t] - Pre[\cdot, t]$. Firing a sequence σ of transitions from M yields $M' = M + C \cdot \vec{\sigma}$ (this is known as the *state equation* of the system), where $\vec{\sigma}$ is the firing count vector of sequence σ, i.e. $\forall t \in T : \vec{\sigma}[t] = \#(t, \sigma)$. The set of all reachable markings (states) from the initial one is denoted by $R(\mathcal{N}, M_0)$. (For more detailed presentations see [7, 40, 42, 49, 53].)

Net systems, like automata, are able to model sequential systems: transitions c and d (see the leftmost net in Fig. 1) fire in *sequence*, while a *conflicting* situation appears between transitions c and e. Combining sequences and decisions *iterations* are obtained: fire e and then f until firing c. But net systems also permit modelling *true concurrency*: transitions b and c may fire in any order or even "simultaneously" (stronger concurrency notion that interleaving). Also the notion of self-concurrency can be represented: after firing a, the two tokens in place 2 permit two concurrent firings of transition b. The basic *synchronization* construct is a transition having more than one input place, such as d: the transition cannot fire until enough tokens have arrived to all the input places. But also arcs from a place to a transition with a weight other than one are a sort of synchronization, because several tokens must assemble in order to enable the transition. These net basic primitives allow to conveniently model typical synchronization schemas, like rendez-vous, semaphores, symmetric and asymmetric rendez-vous implemented by semaphores, fork-join, subprogram flow control, shared resources, guards, etc. [42, 53].

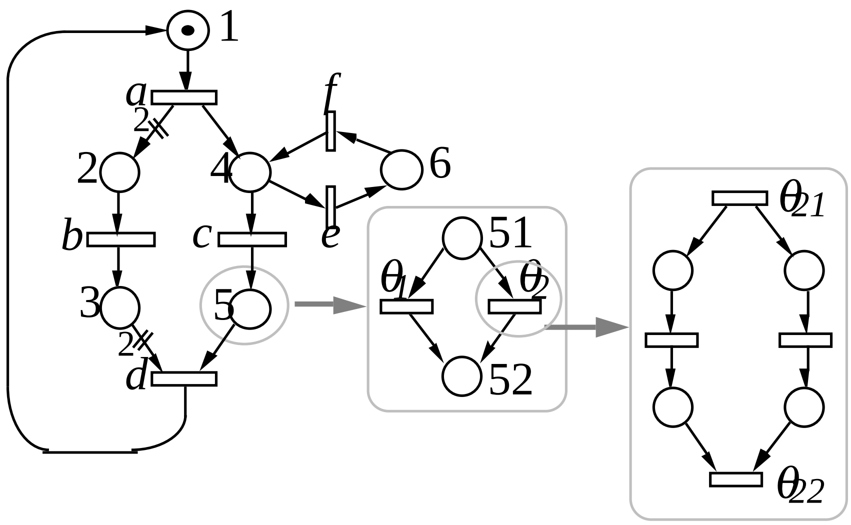

Fig. 1. A P/T net system. Illustration of state and action refinements.

Thanks to the bipartite nature of nets, state variables (places) and their local transformers (transitions) are considered on equal footing. This facilitates the consideration of diverse top-down and bottom-up design methodologies. Figure 1 shows how both state variables, like 5, and actions, like θ_2, can be *refined.* On the other hand, the leftmost net in Fig. 1 could have been built in a *compositional* way. For instance, it can be seen as the synchronization of two components: one defined by places $\{1, 2, 3\}$ together with their adjacent transitions and the other by $\{1, 4, 5, 6\}$ with theirs, merging the common transitions a and d. Or it can be seen as the fusion of two nets defined by their transitions: $\{a, b, c, d\}$ and $\{e, f\}$, merging the common place 4. The design of systems by refinements and compositions is surveyed in [3, 9], and is frequently used in practice [22, 68].

Despite their simplicity, net systems may model extremely complex and intricate behaviours. A classical approach to the solution of complex problems is to study restricted versions of them. This is the aim of defining subclasses. The typical restrictions which are imposed aim at limiting the interplay between synchronizations and conflicts. On one hand, these restrictions facilitate the analysis (we shall discuss some results for a few subclasses in Sect. 4). On the other, some modelling capabilities are lost. 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 subclasses.

The relationship of the system modelled by a net to the external world is usually explicited by the interpretation, that gives a meaning to the abstract objects which form the net. A *state diagram* like interpretation leads to *marking diagrams*, which are well suited to model concurrent switching automatisms. For instance, a *Moore* marking diagram can be constructed associating events to transitions (the transition fires if it is enabled and the corresponding event is true) and unconditional actions to places (the action occurs whenever the associated place is marked). Another interesting family of interpretations is obtained when time and probabilities are associated to the model. For instance, some transitions may have a duration, and a probabilistic conflict resolution conflict may be specified. Timed and stochastic models are frequently used to evaluate performance issues. They can be seen as simple and systematic extensions of *Queueing Networks* (e.g. Queueing Networks with Fork/Join, with passive resources, etc.). This variety of interpretations yields a major advantage of modelling with nets: Net systems provided with appropriate interpretations can be used along the design and operation of systems, using a *single family of formalisms*, and basic concepts and results are shared (can be reused) by the different interpretations, leading to some economy in the analysis and synthesis of models.

In the sequel, only autonomous net systems will be considered.

3 Correctness Verification. An Overview

A major goal of the mathematization of systems is the *verification of their (functional) correctness.* Different concepts of correctness exist. Basically, *a system is said to be correct when two models, namely the* specification *and the* implementation*, are* equivalent [9, 43, 66]*, or when the system exhibits a set of* desirable properties*, either expressed as formulas of a* logic [13, 37]*, or selected from a given* kit *(e.g. boundedness, liveness, etc. See below).* Henceforth, a possible criterion to classify analysis methods is the underlying correctness concept.

Another important criterion to classify analysis methods, specially in the case of concurrent systems, is whether the reasoning is performed directly on the description of the behaviour or not. *Behavioural* techniques consider the reachable states, firable sequences, etc. For bounded systems, this typically allows to decide correctness, but leads to extremely inefficient algorithms due to the *state space explosion problem* [29, 33, 42]. On the other hand *structural* techniques try to obtain information about the behaviour by reasoning on the underlying structure and the initial configuration of resources. This approach typically produces more efficient algorithms but does not always permit deciding correctness, but for some particular subclasses and properties.

Among the properties that are usually part of the specification of (reactive) systems, we want to highlight a few, putting them in terms of P/T net systems. A P/T system is *bounded* when every reachable marking is finite, and it is *live* when every transition can ultimately occur from every reachable marking. Boundedness is necessary whenever the system is to be implemented, while liveness is often required to preclude the existence of actions that can become permanently disabled, or dead, even if there is still some activity in the system. They are so important that the name *well-behaved* has been coined for live and bounded systems. A net $\mathcal{N}$ is *structurally (str.) bounded* when $(\mathcal{N}, M_0)$ is bounded for *every* M_0 and it is *str. live* when *there exists* an M_0 such that $(\mathcal{N}, M_0)$ is live. Consequently, if a net $\mathcal{N}$ is str. bounded and str. live there exists some marking M_0 such that $(\mathcal{N}, M_0)$ is well-behaved. In such case, non well-behavedness is exclusively imputable to the marking, and we say that the net is *well-formed.*

Other interesting properties follow. A weaker property related to liveness is *deadlock-freeness*, i.e. existence of at least one enabled transition at every reachable marking. A reachable marking M is a *home state* in $(\mathcal{N}, M_0)$ iff it is reachable from every reachable marking, and $(\mathcal{N}, M_0)$ is *reversible* iff M_0 is a home state. Finally, transition firing dependencies can be expressed and even measured by a series of *synchronic concepts* [50, 54].

Conventionally, analysis methods of autonomous Petri net models are classified as follows [7, 40, 49]:

Enumeration Techniques. The first step is to compute the *reachability graph*: nodes represent the markings while there is an arc labelled t from M_1 to M_2 whenever firing a transition labelled t which is enabled at marking M_1 yields the marking M_2. If the system is bounded, the construction of the reachability graph can be regarded as an exhaustive simulation of the system, which can be used as the *computational model* for a *proof system*, which verifies formulas from a given (temporal) logic, or for *decision procedures* and *tools for automatic verification* [13, 37]. A major problem of this approach is the size of the state space of a concurrent system. To palliate the state space explosion problem, techniques to disregard markings which are not interesting for a given analysis [64, 65] or to take advantage from symmetries [48] have been proposed, obtaining significant practical improvements on the performance of these methods. It is also possible to deal with unbounded systems by using the *coverability graph* [27, 34], but this construction does not allow to decide important properties such as liveness and reachability [42].

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 eventual efficiency of the algorithms. Two intimately related families of techniques have extensively been used:

Graph Theory. Objects like paths, circuits, handles, bridges, siphons, traps, etc. and their relationships are investigated. Typically only ordinary nets are considered, and the major results are obtained for particular properties, mainly boundedness, liveness and reversibility, and particular subclasses [5, 17, 24, 28, 63].

Linear algebra and convex geometry. Let $(\mathcal{N}, M_0)$ be a P/T system with incidence matrix C. The state equation gives a *necessary* condition for a marking to be reachable: a vector $M \in \mathbb{N}^{|P|}$ such that $\exists \vec{\sigma} \in \mathbb{N}^{|T|} : M = M_0 + C \cdot \vec{\sigma}$ is said to be *potentially reachable*, denoted by $M \in PR(\mathcal{N}, M_0) \supseteq R(\mathcal{N}, M_0)$. The *potential reachability graph* is defined like the reachability graph, but the set of nodes is now $PR(\mathcal{N}, M_0)$ instead of $R(\mathcal{N}, M_0)$. From the state equation it is easy to see that annullers of the incidence matrix lead to linear marking invariants and potential cyclic behaviours (firing invariants), which are useful for reasoning on the behaviour of the system. As an example, a well-known polynomial necessary condition for well-formedness, based on purely structural properties, is *conservativeness* and *consistency*, i.e. existence of positive left and right annullers of the incidence matrix. (For convenience sake, a conservative and consistent net will be called *well-structured*.) These invariants can be used to prove properties like boundedness, mutual exclusion, liveness, etc. More generally, the state equation can be used as a basic description of the system in order to prove or disprove the existence of markings or firing sequences fulfilling some given conditions, eventually expressed as logic formulas [14, 23]. Typically results for general P/T net systems are obtained [15, 36, 39, 59], some of which become specially powerful when restricted to particular subclasses combining graph theory based arguments [11, 25, 58, 60, 61, 62].

Transformation techniques. To facilitate the analysis of a large and complex system it can be transformed (typically reduced) preserving the properties to be analyzed [4, 16, 49]. Transformation rules somehow preserve the behaviour while they are often supported by structural arguments as simple, and efficient, sufficient conditions.

We have outlined the *analysis approach* to the design of correct systems, consisting on verifying correctness by different techniques. It encompasses a "trial and error" methodology. We want to mention to conclude this section that an interesting alternative is the *synthesis approach*, generating correct models by construction (either implementing a given specification or satisfying some desirable properties) by the application of *refinement* and *composition* transformations [38]. Behaviour-based equivalence notions are important to apply these principles [9, 43, 66], but also structural techniques can be used. For instance, a top-down methodology is easily derived reversing the application of reduction rules, and bottom-up methodologies can be obtained by careful study of structural properties. Again restricted subclasses of systems permit the best results [25, 26, 68].

4 Structural Analysis of P/T Net Systems

We concentrate on the structural analysis of selected properties, but we also consider semidecision of properties expressed as formulas of a first order logic.

In order to be able to deal with non ordinary P/T net systems, the use of linear algebra based techniques appears to be more convenient than the use of conventional graph theoretic tools because the matrix representation is exactly the same for ordinary and non ordinary nets, while important graph objects, like siphons and traps, are meaningful only in the ordinary case. Nevertheless the intimate relationship of both families of techniques is often used through the technical development of results, and some objects that were originally conceived from the graph theoretic perspective for ordinary nets can be generalized and are useful for weighted nets [62]. Conversely, results originally conceived from a linear algebraic perspective can be re-interpreted in graph theoretic terms, perhaps leading to more efficient algorithms for particular classes of ordinary nets.

The aim of this section is to overview some of the results that have been recently obtained for non ordinary P/T net systems, either by the use of linear algebra and convex geometry based arguments alone, or by the use of linear algebra combined with graph arguments. The material in this section is organized in two subsections: the first one deals with the analysis of general P/T net systems, and the second is concerned with the analysis of some subclasses.

4.1 Structural Analysis of General P/T Net Systems

The key idea leading to a structurally based verification of properties of a general P/T system is simple, and it has been already commented in the previous section: Let $(\mathcal{N}, M_0)$ be a P/T system with incidence matrix C. If M is reachable from M_0 by firing sequence σ, then $M = M_0 + C \cdot \vec{\sigma}$. Therefore the set of natural solutions, $(M, \vec{\sigma})$, of the state equation $M = M_0 + C \cdot \vec{\sigma}$ defines $PR(\mathcal{N}, M_0)$. This set can be used to analyze properties like marking and submarking reachability and coverability, firing concurrency, conflict situations, deadlock-freeness, mutual exclusion, k-boundedness, existence of frozen tokens, synchronic relations, etc. To do so, the properties are expressed as formulas of a first order logic having linear inequalities as atoms, where the reachability or firability conditions are relaxed by satisfiability of the state equation. This formulas are verified checking existence of solutions to systems of linear equations that are automatically obtained from them [14]. For instance, if $\forall M \in PR(\mathcal{N}, M_0) : M[p] = 0 \vee M[p'] = 0$ then places p and p' are in mutual exclusion. This is verified checking absence of (natural) solutions to $\{M = M_0 + C \cdot \vec{\sigma} \wedge M[p] > 0 \wedge M[p'] > 0\}$. *Integer Linear Programming Problems* [41] where the state equation is included in the set of constraints can be posed to express optimization problems, like the computation of marking bounds, synchronic measures, etc. [14, 54]. For instance, an upper bound for the marking of place p is $M[p] \leq \max\{M[p] \in PR(\mathcal{N}, M_0)\}$. This approach is a generalization of the classical reasoning using linear invariants [36, 39], and it deeply bridges the domains of net theory and convex geometry resulting in a unified framework to understand and enhance structural techniques [14].

Unfortunately, it usually leads to only semidecision algorithms because, in general, $R(\mathcal{N}, M_0) \subset PR(\mathcal{N}, M_0)$. The undesirable solutions are named *spurious*, and they may be removed by the application of certain structural techniques, consequently improving the quality of the linear description of the system [16]. Anyway the algorithms do decide in many situations, and they are relatively efficient, specially if the integrality of variables is disregarded. (This further relaxation may spoil the quality, although in many cases it does not [20, 54].)

In some cases the formulas are complex, like in deadlock-freeness analysis. The formula to express that a marking is a deadlock consists of a condition for every transition expressing that it is disabled at such marking. This condition consists of several inequalities, one per input place of the transition (expressing that the marking of such place is less than the corresponding weight) linked by the "$\vee$" connective (because lack of tokens in a single input place disables the transition). This can be verified checking absence of solutions to a certain (often large) set of linear systems. Nevertheless it is possible to perform simple net transformations preserving the set of solutions to the state equation and reducing the number of systems to solve, basically by obtaining an equivalent wrt. deadlock-freeness net system where the enabling conditions of transitions can be expressed linearly [59]. As a result, deadlock-freeness of a wide variety of net systems can be proven by verifying absence of solutions to a single system of linear inequalities. Even more, in some subclasses it is known that there are no spurious solutions being deadlocks, so the method decides on deadlock-freeness [61].

In temporal logic terms, the above outlined approach is well suited for *safety* properties ("some bad thing never happens"), but not so much for *liveness* properties ("some good thing will eventually happen"). For instance, the formula expressing reversibility would be $\forall M \in PR(\mathcal{N}, M_0) : \exists \vec{\sigma}' \gneq 0 : M_0 = M + C \cdot \vec{\sigma}'$, but this is neither necessary nor sufficient for reversibility. The general approach to linearly verify these liveness properties is based on the verification of safety properties that are necessary for them to hold, together with some inductive reasoning [32]. For instance, deadlock-freeness is necessary for transition liveness, and the existence of some *decreasing potential function* proves reversibility [51].

Another important contribution of linear techniques to liveness analysis has been the derivation of *ad hoc* simple and efficient semidecision conditions. We have already mentioned that conservativeness and consistency are necessary for well-formedness. This condition has been significantly improved by adding a rank upper bound, which was originally conceived when computing the *visit ratios* in certain subclasses of net models [10]. To state this important result, some conflict relations between transitions must be defined first. Two transitions, t and t', are in *Choice relation* iff $t = t'$ or they share some input place. This relation is not transitive. Its transitive closure is named *Coupled Conflict relation*, and the equivalence classes in T are *Coupled Conflict sets*. Another stronger equivalence relation than this is the *Equal Conflict relation*: $t \sim t'$ iff $t = t'$ or $Pre[P,t] = Pre[P,t'] \neq 0$. Note that whenever a transition t is enabled, all the transitions in the equivalence class (or *Equal Conflict set*) of t are also enabled. A slightly modified statement of the result in [15] is: *if a* P/T *net* $\mathcal{N}$ *is well-formed, then* $\mathcal{N}$ *is well-structured and* $\text{rank}(C) < |\mathcal{E}|$, *where* $\mathcal{E}$ *is the set of all Equal Conflict sets of* $\mathcal{N}$. This condition has been proven to be also sufficient for some subclasses of nets [18, 62]. The proof in [18], for (ordinary) Free Choice nets, uses elements of the classical theory (Commoner's theorem and the decomposition theorem), so it was of little help to work out non ordinary subclasses, but remarkably it lead to a general *sufficient* condition for ordinary P/T nets [19]: *if an ordinary* P/T *net* $\mathcal{N}$ *is well-structured and* $\text{rank}(C) < |\mathcal{C}|$, *where* $\mathcal{C}$ *is the set of all Coupled Conflict sets of* $\mathcal{N}$, *then* $\mathcal{N}$ *is well-formed.* Observe that, even for ordinary nets, we do not have a complete characterization of well-formedness. Since $|\mathcal{C}| \leq |\mathcal{E}|$, there is still a range of nets which satisfy neither the necessary nor the sufficient condition for well-formedness!

4.2 Structural Analysis of Subclasses of P/T Net Systems

The development of the structure theory for subclasses of (weighted) P/T net systems is somehow parallel to that for the ordinary case. The syntactical restrictions to facilitate the analysis aim at limiting the interplay between synchronizations and conflicts. The first results were obtained for subclasses where conflicts are structurally forbidden, namely *Weighted T-systems (WTS)* [58] and, more generally, *Choice-free (CF) systems* [60], which are nets where every place has at most one output transition. (The *reverse-dual* subclass, interchanging places and transitions and reversing arcs, forbids synchronizations, and its purely structural results can be obtained automatically transforming those for CF nets.) The next step was allowing both synchronizations and conflicts in a restricted fashion, such that choices are autonomous or free, thus generalizing (Extended) Free Choice net systems. The formal definition of *Equal Conflict (EC) systems* [61] requires that all Coupled Conflict sets are Equal Conflict sets, i.e. $\mathcal{E} = \mathcal{C}$. Basically we are interested in simple and efficient structural characterizations of well-formedness and well-behavedness, that is, correctness of the net and the system respectively. We also consider home states and other properties.

Choice-free Systems [60]**.** These nets strictly generalize the ordinary *Marked Graphs (MG)* or *T-systems* [5, 17], where every place has one input and one output transition, in two ways: weights and attributions are allowed (hence both restrictions on the inscriptions and the topology are relaxed). They still enjoy a crucial property indicating the absence of undeterminism: they are *persistent* [35] for any initial marking. For str. bounded nets, determinism is reflected at the structural level by *the existence of a unique minimal T-semiflow in strongly connected CF nets*. Well-formedness is characterized easily: *a CF net is well-formed iff it is strongly connected and consistent*, while MG are well-formed iff they are strongly connected, because they are always consistent.

The characterization of liveness of a MG in terms of the circuits which compose the net has its analogue in the case of CF systems, where more complex subnets than circuits have to be considered, but it is not so efficient. Nevertheless *it suffices for well-behavedness (and to find a home state) to fire a sequence greater than or equal to the minimal T-semiflow*.

There are many other results for this subclass. Some of them are technical developments to be used in larger subclasses that can be seen as composed by CF subnets. Some others are interesting on their own. For instance, it is proven that all spurious solutions of a well-behaved and reversible CF system can be removed by adding certain linear constraints to the state equation, so *the reachability set can be described by a system of linear inequalities over the naturals*. This result provides a unified framework for verifying properties of reversible and well-behaved CF systems using the approach outlined in Subsect. 4.1, which in this case *decides* because no spurious solutions are left. A slight adaptation of this result leads to a method to compute optimal initial markings making the system well-behaved and reversible, plus other custom (linear) constraints. This result, together with liveness monotonicity on the marking, provides a systematic way to construct well-behaved systems from well-formed nets.

Historically, a more restricted subclass than CF systems was considered first, namely WTS [58], where only the inscriptions are generalized wrt. MG. In that case some further results hold. In particular, every well-behaved WTS is reversible, what extremely simplifies the linear characterization of the state space because it leads to absence of spurious solutions to the state equation of a well-behaved WTS.

Equal Conflict Systems [61, 62]**.** These nets strictly generalize the ordinary *Extended Free Choice (EFC) systems* [5, 28], where every two transitions sharing some input place have exactly the same pre-incidence function, by allowing arbitrary weights except for the output arcs from choice places, whose weights must be equal to keep choice freeness.

Well-formedness is characterized in polynomial time by well-structuredness and the rank condition $\text{rank}(C) = |\mathcal{E}| - 1$. Assuming well-formedness, well-behavedness is characterized in terms of certain subnets which are the analogue of P-components [62]. Unfortunately this condition cannot be checked as efficiently as in the EFC case, where it reduces to prove that there is no minimal

P-semiflow whose support is unmarked by the initial marking. Anyway, liveness of a strongly connected and bounded EC system is equivalent to deadlock-freeness [61], so it can be decided using the approach outlined in Subsect. 4.1 [59].

Other important results for this subclass are: Existence of home states for well-behaved EC systems [61], which generalizes the result in [6] using state equation based arguments. Decomposition and duality theorems for well-behaved systems [62], which generalize the results in [28, 63] after the generalization of the concepts of *components* and *allocations*.

Extensions: Modular Subclasses. Once some basic subclasses are reasonably well understood, their properties can be used for the analysis and synthesis of other subclasses which can be considered as *essentially* similar to them. Among such subclasses, specially challenging and appealing candidates are those which are defined in a *modular* way, i.e. as the composition of smaller models, because this is a most desirable feature from a system design perspective [3]. With respect to the expressiveness versus manageability compromise that was mentioned in Sect. 2, the first has often been the motivation for the definition of important modular subclasses (see [3, 68]). At present, some classes of models that have been considered, and for which there exists a rather rich structure theory, are: EC systems, after their decomposition results [62]. We hope that a synthesis theory analogue to that for EFC [25] will be developed soon. (Increasingly larger versions of) *Deterministic Systems of Sequential Processes* [44, 57, 11], where sequential processes cooperate by restricted message passing. *Systems of Simple Sequential Processes with Resources* [26], where restricted sequential processes compete for shared resources represented by monitor places.

5 Conclusion

Our intention was to briefly survey a rich area of net theory, namely the analysis of autonomous models, where the variety of approaches is perhaps the most salient feature. We concentrated on structural methods, which are a peculiar approach originated within the Petri nets field.

Generally speaking, the more restricted the systems the stronger the results. This principle motivates the study of subclasses, where expressiveness is sacrificed to obtain powerful tools. The study of subclasses had been restricted mainly to ordinary systems, where the representation of bulk services and arrivals is cumbersome. We described several extensions to weighted systems, which have larger expressive power than their ordinary counterparts.

The analysis of autonomous models is interesting on its own, but also it is important for the study of interpreted models. In fact, this nice feature of net models that they can be used with different interpretations has led to mutual beneficial influence between functional and performance analysis [10, 11, 12, 52], and it promises to keep on doing so.

References

1. M. Ajmone Marsan, editor. *Application and Theory of Petri Nets 1993*, volume 691 of *Lecture Notes in Computer Science*. Springer Verlag, 1993.
2. M. Ajmone-Marsan, G. Balbo, and G. Conte. *Performance Models of Multiprocessor Systems*. MIT Press, 1986.
3. L. Bernardinello and F. DeCindio. A survey of basic net models and modular net classes. In [46], pages 304–351.
4. G. Berthelot. Checking properties of nets using transformations. In G. Rozenberg, editor, *Advances in Petri Nets 1985*, volume 222 of *Lecture Notes in Computer Science*, pages 19–40. Springer Verlag, 1986.
5. E. Best and P. S. Thiagarajan. Some classes of live and safe Petri nets. In [67], pages 71–94.
6. E. Best and K. Voss. Free Choice systems have home states. *Acta Informatica*, 21:89–100, 1984.
7. G. W. BRAMS. *Réseaux de Petri: Théorie et Pratique*. Masson, 1983.
8. W. Brauer, editor. *Net Theory and Applications*, volume 84 of *Lecture Notes in Computer Science*. Springer Verlag, 1979.
9. W. Brauer, R. Gold, and W. Vogler. A survey of behaviour and equivalence preserving refinements of Petri nets. In [45], pages 1–46.
10. 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.
11. J. Campos, J. M. Colom, M. Silva, and E. Teruel. Functional and performance analysis of cooperating sequential processes. Research Report GISI-RR-93-20, DIEI. Univ. Zaragoza, Sept. 1993.
12. J. Campos and M. Silva. Steady-state performance evaluation of totally open systems of Markovian sequential processes. In M. Cosnard and C. Girault, editors, *Decentralized Systems*, pages 427–438. North-Holland, 1990.
13. E. M. Clarke, E. A. Emerson, and A. P. Sistla. Automatic verification of finite state concurrent systems using temporal logic specifications. *ACM Trans. Prog. Lang. Syst.*, 8:244–263, 1986.
14. J. M. Colom. *Análisis Estructural de Redes de Petri. Programación Lineal y Geometría Convexa.* PhD thesis, DIEI. Univ. Zaragoza, June 1989.
15. 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.
16. J. M. Colom and M. Silva. Improving the linearly based characterization of P/T nets. In [45], pages 113–145.
17. F. Commoner, A. W. Holt, S. Even, and A. Pnueli. Marked directed graphs. *Jnl. Comput. System Sci.*, 5:72–79, 1971.
18. J. Desel. A proof of the rank theorem for Extended Free Choice nets. In [30], pages 134–153.
19. J. Desel. Regular marked Petri nets. Presented at "Graph Theoretical Concepts in Computer Science", Utrecht, 1993.
20. J. Desel and J. Esparza. Reachability in cyclic Extended Free Choice systems. *Theoretical Computer Science*, 114:93–118, 1993.
21. M. Diaz. Petri net based models in the specification and verification of protocols. In W. Brauer et al., editors, *Petri Nets: Applications and Relationships to Other Models of Concurrency. Advances in Petri Nets 1986, Part II*, volume 255 of *Lecture Notes in Computer Science*, pages 135–170. Springer Verlag, 1987.

22. F. DiCesare, G. Harhalakis, J. M. Proth, M. Silva, and F. B. Vernadat. *Practice of Petri Nets in Manufacturing*. Chapman & Hall, 1993.
23. J. Esparza. Model checking based on branching processes. In volume 668 of *Lecture Notes in Computer Science*, pages 613–628. Springer Verlag, 1993.
24. J. Esparza and M. Silva. Circuits, handles, bridges and nets. In [45], pages 210–242.
25. J. Esparza and M. Silva. On the analysis and synthesis of Free Choice systems. In [45], pages 243–286.
26. J. Ezpeleta and J. Martinez. Synthesis of live models for a class of FMS. In *Procs. of the IEEE Int. Conf. on Robotics and Automation*, pages 557–563, 1993.
27. A. Finkel. The minimal coverability graph for Petri nets. In [47], pages 210–243.
28. M. H. T. Hack. Analysis of production schemata by Petri nets. Master's thesis, MIT, 1972. (Corrections in *Computation Structures Note* 17, 1974).
29. M. Jantzen and R. Valk. Formal properties of Place/Transition nets. In [8], pages 165–212.
30. K. Jensen, editor. *Application and Theory of Petri Nets 1992*, volume 616 of *Lecture Notes in Computer Science*. Springer Verlag, 1992.
31. K. Jensen and G. Rozenberg, editors. *High Level Petri Nets*. Springer Verlag, 1991.
32. C. Johnen. Algorithmic verification of home spaces in P/T systems. In *Procs. IMACS 1988, 12th World Congress on Scientific Computation*, pages 491–493, 1988.
33. N. D. Jones, L. H. Landweber, and Y. E. Lien. Complexity of some problems in Petri nets. *Theoretical Computer Science*, 4:277–299, 1977.
34. R. M. Karp and R. E. Miller. Parallel program schemata. *Jnl. Comput. System Sci.*, 3:147–195, 1969.
35. L. H. Landweber and E. L. Robertson. Properties of conflict-free and persistent Petri nets. *Jnl. of the ACM*, 25(3):352–364, 1978.
36. K. Lautenbach. Linear algebraic techniques for Place/Transition nets. In W. Brauer et al., editors, *Petri Nets: Central Models and their Properties. Advances in Petri Nets 1986, Part I*, volume 254 of *Lecture Notes in Computer Science*, pages 142–167. Springer Verlag, 1987.
37. Z. Manna and A. Pnueli. The anchored version of the temporal framework. In J. W. de Bakker et al., editors, *Linear Time, Branching Time and Partial Order in Logics and Models of Concurrency*, volume 354 of *Lecture Notes in Computer Science*, pages 201–284. Springer Verlag, 1989.
38. K. M.Chandy and J. Misra. *Parallel Program Design*. Addison-Wesley, 1988.
39. G. Memmi and G. Roucairol. Linear algebra in net theory. In [8], pages 213–223.
40. T. Murata. Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, 77(4):541–580, 1989.
41. G. L. Nemhauser and L. A. Wolsey. *Integer and Combinatorial Optimization*. Wiley, 1988.
42. J. L. Peterson. *Petri Net Theory and the Modeling of Systems*. Prentice-Hall, 1981.
43. L. Pomello, G. Rozenberg, and C. Simone. A survey of equivalence notions for net based systems. In [46], pages 410–472.
44. W. Reisig. Deterministic buffer synchronization of sequential processes. *Acta Informatica*, 18:117–134, 1982.
45. G. Rozenberg, editor. *Advances in Petri Nets 1990*, volume 483 of *Lecture Notes in Computer Science*. Springer Verlag, 1991.

46. G. Rozenberg, editor. *Advances in Petri Nets 1992*, volume 609 of *Lecture Notes in Computer Science*. Springer Verlag, 1992.
47. G. Rozenberg, editor. *Advances in Petri Nets 1993*, volume 674 of *Lecture Notes in Computer Science*. Springer Verlag, 1993.
48. K. Schmidt. Symmetries of Petri nets. *Petri Net Newsletter*, 43:9–25, 1993.
49. M. Silva. *Las Redes de Petri: en la Automática y la Informática*. AC, 1985.
50. M. Silva. Towards a synchrony theory for P/T nets. In [67], pages 435–460.
51. M. Silva. Logical controllers. In *IFAC Symposium on Low Cost Automation*, pages 157–166. 1989.
52. M. Silva. Interleaving functional and performance structural analysis of net models. In [1], pages 17–23.
53. M. Silva. Introducing Petri nets. In [22], pages 1–62.
54. 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 Verlag, 1988.
55. M. Silva and J. M. Colom. Petri nets applied to the modelling and analysis of computer architecture problems. *Microprocessing and Microprogramming. The EUROMICRO Jnl.*, 38(1-5):1–11, 1993.
56. M. Silva and R. Valette. Petri nets and flexible manufacturing. In G. Rozenberg, editor, *Advances in Petri Nets 1989*, volume 424 of *Lecture Notes in Computer Science*, pages 374–417. Springer Verlag, 1988.
57. Y. Souissi and N. Beldiceanu. Deterministic systems of sequential processes: Theory and tools. In *Concurrency 88*, volume 335 of *Lecture Notes in Computer Science*, pages 380–400. Springer Verlag, 1988.
58. E. Teruel, P. Chrzastowski, J. M. Colom, and M. Silva. On Weighted T-systems. In [30], pages 348–367.
59. E. Teruel, J. M. Colom, and M. Silva. Linear analysis of deadlock-freeness of Petri net models. In *Procs. of the 2nd European Control Conference*, volume 2, pages 513–518. North-Holland, 1993.
60. E. Teruel, J. M. Colom, and M. Silva. Modelling and analysis of deteministic concurrent systems with bulk services and arrivals. In *Procs. of the ICDDS '93*, pages 93–104. North-Holland, 1993.
61. E. Teruel and M. Silva. Liveness and home states in Equal Conflict systems. In [1], pages 415–432.
62. E. Teruel and M. Silva. Well-formedness and decomposability of Equal Conflict systems. Research Report GISI-RR-93-22, DIEI. Univ. Zaragoza, Nov. 1993.
63. P. S. Thiagarajan and K. Voss. A fresh look at Free Choice nets. *Information and Control*, 61(2):85–113, 1984.
64. A. Valmari. Stubborn sets for reduced state space generation. In [45], pages 491–515.
65. A. Valmari. Compositional state space generation. In [47], pages 427–457.
66. K. Voss. Interface as a basic concept for systems specification and verification. In [67], pages 585–604.
67. K. Voss et al., editors. *Concurrency and Nets*. Springer Verlag, 1987.
68. M. C. Zhou and F. DiCesare. *Petri Net Synthesis for Discrete Event Control of Manufacturing Systems*. Kluwer Academic Publishers, 1993.