Locally Markovian Object Nets

Dietmar Wikarski
Fraunhofer-Einrichtung für Software- und Systemtechnik (ISST)
Kurstraße 33, O-1086 Berlin, Germany

Abstract

An object net (ON) can be derived from a (Place-Transition) Petri net (PN) by enhancing the latter with modularization information. The resulting components, called for different reasons objects, have the property that any pair of transitions of the PN which are in a static conflict belong to the same object in any ON derived from that PN.

Suppose that there are given probability distributions for the solution of dynamic conflicts in favor to the different sets of concurrently enabled transitions which depend at most upon the current markings of the net but not upon past ones.

Assuming this property for a PN, usually called memoryless or Markovian, it will be natural to suppose this dependency only on the marking of those places directly connected with the transitions in conflict. According to the definition of objects, the probability distributions for conflict solution will then depend only upon the local markings of the object which the conflict belongs to. For this reason, an ON with this property together with the local probability distributions will be called locally Markovian object net (LMON).

Based on the local introduction of probabilities, a natural introduction of local "time" becomes possible. In particular, this local time may be qualitatively and quantitatively characterized by Phase-Type probability distributions.
Consequently, it may be used for an appropriate characterization of the behaviour of the subnets encapsulated by the objects and thus for a qualified abstraction and refinement in "quantified nets".

As an important potential application, LMONs promise to be well suited for a distributed probabilistic simulation of models of distributed systems. In particular, performance measures can be obtained for distributed programs and protocols based on a natural (with respect to the maintenance of the locality principle) "quantitative" extension of a pure "qualitative" model.

1.0 Introduction

The starting for the approach presented in this research note was the observation that the qualitative properties (boundedness, liveness etc.) of a Petri net are not preserved in the general case when global time is introduced as usual in timed and stochastic Petri nets. This fact is closely related to the modification of the description power of the arising classes of models which either to that of Turing machines or to that of Markov chains. In some cases, previously decidable properties become undecidable.

The deeper reason for these effects is the principal modification of the transition rule: Instead of using a local transition rule as in Place/Transition nets, in timed and stochastic nets maximal sets of concurrently enabled transitions have to be chosen and to be fired concurrently.

Such a firing strategy requires a permanent global view on the marking of a Petri net and contradicts the general idea of modelling distributed systems with Petri nets: This idea is to predict properties of the globally **potential behaviour** of a distributed system, whereby the construction of the model is based on local observations of its real behaviour or on local specifications of its potential behaviour. The global behaviour of the model is defined by the MULT (steps of single transitions) or by the ARB (steps of arbitrary sets of concurrent transitions) semantics ([SM81]) and is thus basically locally determined.

The approach described in the given paper extends this general approach of Petri nets to the prediction of the **probable behaviour** of a distributed system. Thereby, the potential behaviour of a distributed system is assumed to be fixed by an underlying PN. Prediction of the probable behaviour is based on local observations of the frequencies of conflict solutions within certain subnets of the underlying PN called **objects**.

Objects (cf. [Wika90]) may be seen as a type of locality in nets next higher to transitions in the following sense: To know for a transition whether it is enabled to fire in the next step, it is sufficient to know the markings of its “surrounding” places (i.e. preplaces in the case of infinite, and pre- and postplaces in the case of finite capacities of places). To know the probability for a set of concurrent transitions to fire in the next step, the knowledge of the local marking of the object which this set belongs to is assumed to be sufficient.

Such a fundamental property of objects (as parts of a net model whose originals are repeatedly observable by local observers) is due to the following basic assumptions:

Firstly, an object is such a subnet of a PN that the surroundings of all transitions being in a static conflict in this subnet are part of this subnet, too (see chapter 2).

Secondly, the probability of the solution of a dynamic conflict in favor to a certain set of concurrent transitions may depend only upon the current marking but not upon past behaviour. For stochastic processes, such a property is usually called **Markovian** one. Due to the natural restriction that this dependency will be assumed only on the marking of those places which are causally connected to the transitions involved in the conflict, this property will be called **local** Markovian one.

2.0 Canonical Modularization in Petri Nets: Object Nets

2.1 Formal Definitions and Properties

2.1.1 Petri nets

Be N = [S, T, F] a net, i.e. the finite sets S (of places) and T (of transitions) and the relation F ("flow relation") fulfil the conditions

(1) $S \cup T \neq \emptyset$, $S \cap T = \emptyset$,

(2) $F = C \cup E$ with $C \subseteq S \times T$ and $E \subseteq T \times S$

(3) $\mathrm{dom}(F) \cup \mathrm{cod}(F) = S \cup T$ with $\mathrm{dom}(F) = \{x \in S \cup T\colon\ y \in S \cup T,\ (x,y) \in F\}$

and $\mathrm{cod}(F) = \{y \in S \cup T\colon\ x \in S \cup T,\ (x,y) \in F\}$

A **Petri net** is usually defined as a net with weighted arcs together with an initial marking and a well-defined transition rule, i.e. PN = [S, T, F, W, Mo] , where

(4) [S, T, F] is a net,

(5) $W\colon F \mapsto N^+$ is a mapping (the **weights** of the arcs)

(6) $M_0\colon S \mapsto N$ is a mapping (the **initial Marking** of PN)

The usual ("weak") **transition rule** defines a transition $t \in T$ as **enabled**, iff all of its preplaces contain a number of equally or more tokens than the weight of the arc connecting this place with this transition, i.e. for all s of S holds:

If $(s,t) \in C$ then $M(s) \leq W(s,t)$.

Hereby the mapping $M\colon S \mapsto N$ defines the **current marking** of the net, i.e. M(s) is the number of tokens currently on the place s. (By M we will denote also the vector of the numbers of tokens currently on the places of the net.).

An enabled transition can **fire**, i.e. take the numbers of tokens equal to the weights of the incoming arcs from its preplaces and put the numbers equal to the weights of the outgoing arcs on its postplaces.

A set of transitions U is called **concurrent for given a marking M**, if the number of tokens available in the preplaces of the transitions of U is sufficient for these transitions to be fired simultaneously according to the transition rule given above. The set U is called to be **in a** (dynamic) **conflict for a given marking**, if all of its transitions are enabled, but not concurrent for this marking.

Given a net N, a pair of transitions is called to be **in a static** (or structural) **conflict**, if they share a common preplace.

2.1.2 Object Nets

The tuple ON = $[O, E_O]$ is called **object net**, iff

(7) $O = \{O_i, i \in I\} = \{[S_i, T_i, F_i, W_i, M_{0i}], i \in I\}$

is a finite set of disjoint Petri nets and

(8) $E_O \subseteq T \times S$

is a relation. The elements $s \in S_i$ and $t \in T_i$, for which there is an $x \in X$ with $(x,s) \in E_O$ or with $(t,x) \in E_O$ are called **border elements** (more precisely: **input places** or **output transitions,** respectively) **of the (Petri) object** O_i.

A Petri net $O = [S, T, F, W, M_0]$ is called **(Petri) object**, if for it a non-empty set $B_O = S_O \cup T_O$ of border elements is defined (with S_O and T_O denoting subsets of S and T, respectively). Thus, the Petri nets O_i configuring in (7) may be called objects due to the definition of their border elements given by (8).

A subnet N1 = [S1, T1, F1] of a net N = [S, T, F] with $F = C \cup E$ as in (2) is called

- object of N, if it holds

(9) $F \cap ((X1 \times (X \setminus X1) \cup (X \setminus X1) \times X)) \subseteq E$, with $X1 = S1 \cup T1$ and $X = S \cup T$,

- **basic object of N**, if the graph of F1 is a maximally connected component of the graph of the relation C,

- **component** or **complete object of N**, if N1 is an object of N and moreover a subnet of N, i.e. additionally to (9) it holds

(10) $F1 = F \cap ((S1 \times T1) \cup (T1 \times S1))$,

- **basic component** or **complete basic object of N**, if $[S1, T1, F1 \cap C]$ is a basic object of N and it holds

(11) $(T1 \times S1) \cap F \subseteq F1$.

2.1.3 Elementary Properties of Object Nets, Objects and Basic Objects

The following assertions can be proven straightforwardly from the definitions given above

(P1) An object net is a Petri net together with a given decomposition into components of the underlying net (i.e. into subnets fulfilling conditions (9) and (10)).

(P2) The basic objects of a net are uniquely determined as the transitive hull of the (static) conflict relation. Consequently, any pair of transitions which might be in a (dynamic) conflict in a Petri net belong to the same basic object of the underlying net.

2.2 Motivation for and Interpretation of Objects

As mentioned in the introduction, the main goal of the proposed approach is to get able to predict properties of the global probable behaviour of a distributed system based on local observations or on local specification of its behaviour.

When considering the basic model of net theory, Condition/Event systems, it becomes clear that the “locality” which has to be observed or specified for modelling the potential behaviour of a distributed system, is the “surrounding” of an **event**: If a simultaneous change of the states of (the real world prototypes of) a certain set of conditions occurs, this occurrence is taken for “causality” between the involved conditions via the common event. This well-defined (by its surrounding) event may occur in each circumstance when the same situation holds again and thus defines potential local behaviour.

Petri nets (more precisely: Place/Transition systems), which can be considered as an aggregated and abstracted net class of Condition/Event systems, take the same concept: **Transitions** together with their surroundings can be interpreted as (minimal) locality in the behaviour of a Petri net[1].

As the next higher level of locality in nets, we consider the **basic objects**, which were introduced in [Wika90] (cf. previous paragraph; another, suitable name due to W. Reisig is “conflict clusters”) and **objects**, representing commonly observable subnets.

In some sense, basic objects and objects may be considered as the canonical localities for modelling the probable behaviour of nets corresponding to events and transitions as the canonical localities for modelling its potential behaviour: As sketched above, the latter are sufficient for a non-deterministic distributed simulation of the corresponding model, because the knowledge of the current marking of their surrounding is sufficient to decide whether an event resp. transition can fire or not. But: In the case of arising conflicts no information is available about how to solve them. Therefore, we will assume that this information can be obtained by repeated observation of the same (local) conflict situation: The observed relative occurrence frequencies of the different sets of concurrent transitions will be taken as the corresponding probabilities in a distributed local simulation (cf. next chapter).

The relation between basic objects and objects is the following: Basic objects are uniquely determined as minimal connected subnets of an (unmarked) net containing only arcs from places to transitions. Objects of a net are arbitrarily composed of them using arcs from transitions to places of this net. (The object net corresponding to an underlying Petri net contains **all** of its arcs from transitions to places.) Petri objects are marked objects of a net.

---

1. Strictly speaking, we have to distinguish between Petri nets with finite place capacities and those with infinite ones. In the first case, “surrounding” of a transition means, as above, the set of all pre- and postplaces of a transition. Within the case of infinite place capacities, which is equivalent to the application of the weak transition rule, it is sufficient to consider the set of all preplaces of a transition because the markings of the postplaces of a transition do not influence its enabling. Due to the fact that Petri nets with place capacities can be locally simulated by those with infinite ones (cf. [Star90]), for the goal pursued here it is sufficient to consider only Petri nets with the weak transition rule.

As in the case of events and transitions, both basic objects and objects have a meaningful semantics only together with its surroundings (i.e. with the incoming arcs and transitions).

As one of the most important points, objects may and should be chosen in such a way that they represent a commonly observable part of the real world system to be modelled or a commonly controllable part of a system to be specified. Such a commonly observable/ controllable part, in particular allows the observation/specification of simultaneous changes corresponding to concurrent transitions.

Thus, in connection with their properties with respect to conflicts, objects are a natural locality in Petri nets allowing a local introduction of probability information for the local solution of conflicts.

3.0 Locally Markovian Object Nets

3.1 Formal Definition

Given the notations of (1) to (8), the tuple MN = [ON, P] is called **locally Markovian object net** (**LMON** for short) if

(12) $P = \{P^{ik}, i \in I, k \in K_i\}$

is a family of probability distributions, defined over the sets $U^{ik}{}_l$, $l \in L^{ik}$, of concurrent transitions for each reachable marking M_{ik}, $i \in K_i$, of each object O_i, of the underlying object net. Thereby, the K_i denote the index sets of the reachable markings of the objects of ON.

For convenience, let us consider the sets of potentially concurrent transitions of an object in the following as "object transitions" and denote them generally by lower case "u", with appropriate indexes if necessary. Furthermore, we will denote by U_i the set of object transitions of the object O_i and by $U = \cup U_i$ the set of all object transitions of a given object net.

The set of enabled object-transitions of a reachable marking M_{ik} will be denoted by U_{ik} and the individual probabilities of the corresponding distribution P^{ik} will be denoted by $p^{ik}{}_u$, $u \in U_{ik}$.

The enabling of transitions be governed by the so-called **local probabilistic transition rule:** Given that the marking of object O_i is M_{ik}, the object transition u is enabled with probability $p^{ik}{}_u$. For the global semantics of an LMON, we allow at least the following two cases which will be considered in some detail:

Firstly, an enabled object transition u **may** fire, but needs not to do so. Thus, by a firing of an object transition of another object, this object transition might become disabled or another marking of the object might arise resulting in another set of enabled object transitions. In accordance with the usual MULT-semantics of Petri nets we will assume that only one object transition may be fired in a (global) "time point". Such a semantics seems to be a reasonable extension of the MULT- semantics (as extension, it includes the concurrent firing of concurrent sets of transitions) and a restriction of the ARB-semantics of Petri nets (it excludes the concurrent firing of transitions of different objects). It might be interpreted as a network of "asynchronously coupled" stochastic automatons and leads to a description power differing from that of Turing machines. Further investigations should show its usefulness for defining and proving properties of distributed systems allowing local statistical observations.

Secondly, we may assume that an enabled object transition **must** fire. In this case it becomes useful to speak about local (and discrete) time scales associated with the objects and, assuming some additional restrictions, to characterize structure and marking (and thereby local behaviour) of an object by some quantifiable delay function. Such a characterization can be used for abstraction from subnets, thereby not loosing qualitative (net structure) and quantitative (markings and probabilities) information of these subnets. This case will be considered shortly in the next paragraph.

3.2 Local Time in Nets

For the object transitions of an LMON, assume the object-local must-firing rule as described above. Suppose furthermore that the objects are conservative with respect to the numbers of firings to input-places and of output-transitions (i.e. to each incoming token corresponds just one firing of an output-object-transition - such a property is e.g. fulfilled for the trivial case of an object with the net structure of a state-machine). The behaviour of each object is then characterized by a Markov chain derived from its net structure, initial marking and transition probabilities of the object transitions. More precisely, the sojourn time of a token in an object (defined as the number of entrances into internal places) can be characterized by a so-called discrete Phase-type distribution ([Neut81]) in the case of at least one stochastic branch or as a deterministic finite number in the other case.

4.0 Concluding Remark

As it is seen by the states of elaboration and application of the presented conceptual model, much work of investigation is left to be done. Nevertheless, it might prove to be useful to present the results in this stage, in order to have a preliminary but fixed version of the concept which allows further investigation and application by others.

I would like to thank MONika Heiner for her interest in this topic and the stimulation to write this note.

5.0 References

[Moll85] M.K. Molloy, Discrete Time Stochastic Petri Nets, IEEE Transactions on Software Engineering, Vol. SE-11(1985)4, pp. 417-423

[Neut81] M. F. Neuts, Matrix-Geometric Solutions in Stochastic Models, John Hopkins University Press, Baltimore 1981

[SM81] A. Salwicki, T. Müldner, On the Algorithmic Properties of Concurrent Programs, LNCS 125, Springer, Berlin, 1981, pp.169-197

[Star90] P.H. Starke, Analyse von Petri-Netz-Modellen, B.G. Teubner, Stuttgart, 1990

[Wika90] D. Wikarski, Object Nets - A Canonical Class of Models for Behaviour Simulation and Structure Synthesis of Distributed Systems?, Informatik - informationen, reporte, 6(1990)12, pp. 91-100

[Wika91] D. Wikarski, Ein neuer Ansatz zur Integration von Petri-Netz- und Warteschlangennetz-Modellen, ASIM-Mitteilungen 26 (1991)

[Wika91a] D. Wikarski, Zur Modellierung des Verhaltens verteilter Systeme auf der Basis von Netzen, Forschungsbericht, IIR Berlin, 1991

Revised version of Locally Markovian Object Nets - the Conceptual Model, which was published in: Proceedings of the International Seminar on Concurrency, Programming and Specification (CS&P92),Humboldt-University, Berlin, November 1992, pp. 182-189