Communicating Petri Nets

G. Degli Antoni[1]

Abstract

In this present paper basic Petri Nets models are re-viewed. Their difficulties in modelling some real systems (cognitive and epistemological difficulties) are analyzed. In order to overcome these difficulties, the basic models (Condition Event/Elementary Net) are extended. The proposed extension implies a basic mechanism of communication, whereby the holding of a transmitting condition lets the receiving events to fire, whenever their have concession in the classical sense. The corresponding token rule is suggested, leading itself to a natural extension of the notion of Petri Nets, where there is a communication between different Nets (components) modelling parts of a single system.
Simple examples are given in order to illustrate the easiness in modelling the proposed extension of Petri Nets. It is observed that the proposed extensions fit with a general movement towards object orientation in modelling complex concurrent systems.

1. Introduction

Petri Nets are one of the most interesting formalism to model systems. A large number of people has considered them as an important subject of theoretical research. Certainly the nets ability to nicely represent the notions related to causality and concurrency makes Petri Nets a field where fundamental research could lead to a deeper understanding of processes and systems.

[1] Dipartimento di Scienze dell'Informazione, Università degli Studi di Milano
Via Giuseppe Colombo 46, 20133 Milano, e.mail degliantoni@hermes.unimi.it

The research has been developed in the frame of a cooperation project (Personal University Project) between Milano University (Italy) and Catholic University of Asunciòn (Paraguay), sponsored by the Minister of Foreign Affairs jointly with the Minister of University and Scientific Research of the Italian Governement.
In the PU project the the Communicating Petri Nets are related to an hypermedia modelling system which is under developement by S. Aranda, M. Bondi, G. Grossi.

Nevertheless, those that have used Petri Nets for practical purposes realize difficulties in representing real life systems and processes. These difficulties are well predicted in Petri's conferences on Nets. In fact, frequently Petri claims that in every field the right language has to be used. Petri's position is extremely honest and certainly correct, but it is difficult to avoid a positive thinking that suggests to make research in order to introduce small modifications in Nets to improve their expressiveness, and their usability.

That research would be of a purely empirical nature as it happens exactly any other really scientific discipline. Only human non mathematical judgements can justify sentences of any theory axiomatizing some notions (sets, numbers, spaces, space time...). Obviously, given any axiomatization, its consequences can be subjected to the mathematical investigation, just to prove theorems or to enlight the empirical nature (models, both mathematical or cognitive ones) of the concept captured by the axioms.

In the case of Petri Nets, the basic definitions collect fundamental ideas: their capability of representing extremely wide classes of processes makes them an interesting tool for practical or for epistemological research, i. e. for making science through modelling activity.

It follows that the choice of good models for Petri Nets is an important matter.

Perhaps a research channel has to be opened, whereby comparison of modelling power of net definitions have to be performed in order to eliminate all the doubts that some practitioners maintain regarding the usability of Petri Nets.

The loss of literature and the extreme differences in formalism used to reason about Petri Nets, together with mathematical difficulties of many papers, is transforming the field of Petri Nets in a purely academic subject in which it is honestly difficult to have clear answers to simple problems.

The state of affairs suggests that a non completly mature basic definition of Petri Nets has been reached and that we have to look again at the first papers with curiosity and criticism, even considering the many researches that have been developed since the early days of Petri Nets.

Following this esprit the present paper suggests a new class of Nets derived from Petri's ideas in order to overcome some of the difficulties encountered in using Nets. The results seem interesting for various reasons, even related to the nature of nets coming out from our analysis. The resulting nets, that we will call *Communicating Petri Nets* (CPN), certainly can be classified as Object Oriented Petri Nets. It is interesting to note that our result is coming out not as a consequence of the actual pressure towards object orientation, but rather from the aim of understanding net behaviour in some critical situations (1).

2. A concise re-view on net definitions

A concise review on nets and particularly on their token games is here introduced for sake of synthesis and to build up with the reader a common view on nets. Generally, in the many models of net behaviour specification a token game is considered, eventually described through some mathematical definition that gives a better insight on the meaning of net behaviour. Token games bring a lot of intuition on problems related to the net behaviour through the physical nature of the token game itself.

Generally the various models of token games used in literature make transition firing (event occurrence) change the tokens pattern in the places directly related to the considered transitions. The set of holding conditions (case) changes as due to transition firings (processes). Firing is subjected to some limitation that are generally called concession conditions. Concession holds for sets of transitions. Concessed transitions can occur concurrently if conflicts are not present. A concessed transition is in conflict with another one if its firing changes the token pattern in such a way that it is impossible for the second one to fire. Generally a conflict is a symmetric relation. The interaction between conflicts and concurrency give rise to interesting confusion situations where conflicts avoidance is an observer related property (subjectivity). Class of nets depends heavily on the choice of concession and firing rules and on the nature of the places: these can be made holding one or more tokens. Extensions have been obtained considering structured tokens. Concession and firing rules will be related to those data structures associated to the token types.

Properties of class of nets have been studied extensively in order to relate local behaviour to the general one and static properties to dynamical ones. In this latter category for our purposes we want to recall safety: this property tries to identify dangerous situations in real systems, if there are contacts in the corresponding net model.

In order to simplify the burden of so many possibilities, Petri has introduced Net Morphisms, suggesting that all net models should be reduced - through net morphism - to some elementary net model (2). The preferred Petri's choice has been the Condition Event (CE) nets model. Elementary Net Model is sometimes preferred (3). In these models places hold a single token and are called conditions. Transitions are called events and have concession if all their input conditions (places) and no output conditions hold. The occurrence of an event changes all its input and output conditions.

The concise view given here can be easily translated into a more compact mathematical notation with the danger of loosing some underlying idea. We prefer to leave it as informal and as concise as possible, preferring possible ambiguities (if any) to the difficulty in reading formal definitions. These are certainly needed when properties have to be studied, not now in our definitional approach.

3. A cognitive difficulty with Petri Nets

In all models of Petri Nets, occurrence of events, that is firing of transitions, is essentially a change of cases, that is a change on the state of the token =holding places (conditions). Some condition will cease to hold while others will begin. A fundamental idea is indeed captured: no event occurs anywhere without change on some condition (extensionality).

The unfortunate state of affairs with current definitions is that some extremely simple system can not be represented. To exemplify, let's consider the flowing of current after a switch has been closed in an electrical circuit or any other analog system. The way to represent that system is to consider the concurrent changes of both switch and current as events (Fig. 1).

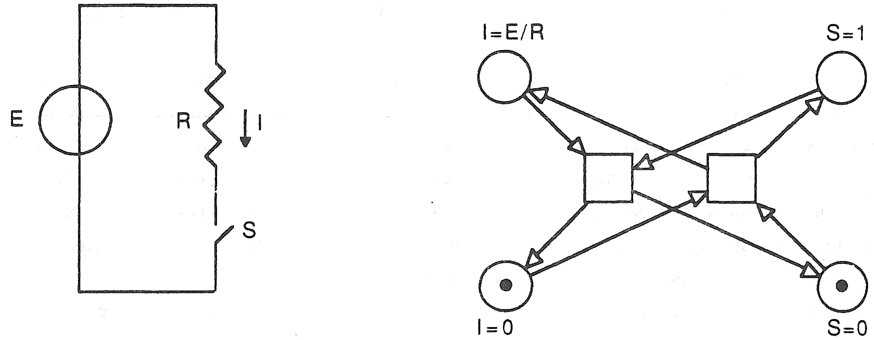


Fig. 1

It is acceptable that there must exist some form of concurrency between the changing of the state of the switch and the change of current flowing through the switch itself. But the intuitive causal dependence of the current flow consequent tp the switch closure has been lost. But it is completely unacceptable to let the transition of the switch be dependent on the state of the current flow. In some sense the net brings some information on the phenomenon: infact it is quite likely that at some level of description the state of the switch is related to the current flowing on it. So the net models only a very limited state of affairs, perhaps acceptable in some extremely deep sense. But the simple intuitive meaning of the circuit fully understandable by any human mind is lost, indicating some fundamental difficulty. The fenomenology implied in the example shown before is perhaps related to some deep discrepancy between composition rules in nets and composition rules in circuits. If these considerations are correct, it will be probably difficult to understand a better compositionality in nets working inside the net theory as actually developed. Unfortunately, the example shown has to do with all

systems where changes in components modify other components. The state of affairs related to the selected example has consequences in all real situations. If the discussion is correct, then it is not difficult to understand why nets are used and then abandoned after their strong maieutical power has been depicted to analyze the system at hands.
With respect to the latter assertion, it is certainly likely that nets owe their maieutical power to many reasons, certainly related to the double role of places. Infact places in nets represent an indication act on the world (4), that is some entity and its model as assumed by the reader (or modeler). Moreover, the effort that modelling activity requires, is probably responsible for the judgement on the maieutical power of nets. Maieutical power and simplicity of the models have made Petri Nets the most known cognitive tool for the representation of real systems behaviour with concurrence. The difficulties reported in this chapter do not weaken the Petri Nets as a modelling tool. Viceversa, we think that all criticism will improve a better understanding of nets and their applications to real life problems.

4. An epistemological difficulty

The last chapter has discussed some difficulties on modelling systems with Petri Nets in their actual model. The nature of the difficulties discussed are related to real systems and their models, a topic rarely analyzed in the theoretical Petri Nets literature. We call those difficulties cognitive ones. There are others difficulties with Petri Nets when their interaction with agents (for example the reader using them for simulation purposes it is taken into account).
The problem can perhaps be classified as an epistemological one and it is related to the nature of the observation on nets. They are done just as if the agent using nets does not influence the net behaviour. The agent can use information on nets as a system with complete information on everything is happening on the net. For example, he can decide to make event occur (transition firing) taking the information from its own state, without discussing on how this happens.
This implies some form of communication from the net to the observer and from the observer to the net. For example, if simulation is made for comparing a model A with a model B, an event on A is made to occur from the analyst just considering some state in B in order to compare consequences of changes. More precisely, if some state of affairs is present in one of the system, an event will be made to occur in the other one.
It is a custom of all sciences to consider abstractions. In our example we have to abstract from the existence of the observer but we have to take into account what happens.

What happens can be understood using net concepts if we consider that some event can occur if and only if there is an *external condition of occurrency,* as it happens when by hand simulation we decide to fire a transition. That decision does not necessarily modify the state of the affairs that has led to the firing decision.
A similar state of affairs will hold in communication from one net model to other net models.

5. Extending the net model

Extending the actual models of net behaviour can help solving the epistemological and the cognitive difficulties discussed above.
The discussions above imply that those difficulties are not present if some form of state is associated to the events and that state depends on what happens. For example, the current can flow through a switch only if this has been closed. In order to take into account the suggestion provided by the present discussion, an extension to the net re-view presented above is convenient. We will call the resulting nets Communicating Petri Nets.
Communicating Petri Nets are a set of Condition Event (or Elementary) Nets that will be called Components. Each of these Components behave exactly as usually and as described in our re-view for elementary net models.
A Communication Structure is superimposed to the components. The Communication structure is given as a set of pairs *(t, r)* called elementary channels. The first element of each pair is a condition of some Component, the second element is an event of a different Component. So a pair *(t,r)* indicates an elementary channel whereby the holding of a transmitting condition *t* sends some signal to the corresponding receiving event *r*. A component A communicates with a component B $(A \neq B)$ in a Communicating Petri Net if there is a set of elementary channels with transmitters in A and the corresponding receivers in B.
We will call the events that are receivers of some condition R-Events (in word: committed events). Transmitting conditions will be indicated with T-Conditions. In a pair *(t,r)* we will call *r* committed by *t*. It is to be noted that a transmitter *t* may commits different receivers, and that a receiver can receive commitments from distinct transmitters.
To complete the model, concession and firing rules must be extended. There are many possible ways to do it. We suggest here one possibility that agrees fully with Petri's attitude regarding timing details, i.e. timing are fully excluded from definitions. Signal reception by an r-event is modelled putting a token in it . The sending of messages by a t-condition *p*, in our preferred choice, happens when conditions begin to hold, or equivalently when an event (committed or not) has placed a token on *p*.

A choice has still to be done: to decide how a c-event stores the received message (a token) from some t-condition. Our choice is the following: if a c-event has concession in the spirit of the standard concession, then the received message will be stored in it just to the firing of the event itself.
Now we can completely formulate our new c-concession and c-firing rules. A standard event (one that is not an R-Event) has c-concession if it has concession in the usual well known sense. An r-event has c-concession if it has concession in the usual sense and it holds a token (commitment) in it. Events occur (transitions c-fire) only if they have c-concession c-firing of an event is standard if that event is standard. If the event *e* is an r-event, then the token game proceeds as usual but, a token (commitment) is placed in every uncommitted (that is a non holding token) standardly concessed receiver event associated to the transmitting output conditions of the considered event *e*. It is to be noted that in these definitions a situation similar to the contact is not considered.
As any other net, our Communicating Nets require an initial marking. The marking can be associated to both conditions and events. It is easy to accept as initial marking definition one that requires only initial marking on places. Non receiving events are just those events that model some autonomy in our model. It will be their responsibility to send tokens to the transmitting conditions, making possible a message sending from t-conditions to r-events sites. Side effects of the event occurrence is the reception of commitments by receiving events concessed in the usual sense.

6. The notation for the extended model

The basic elements of the notation are shown in fig. 2. The notation is suitable for further extensions of the model in order to consider (for example) data types associated with conditions and the nature of messages. In our simple model, the letters T in places and R in transitions are redundant. Elementary communicating channels are described by the correspondence between symbols indicated at the t-conditions and the corresponding r-events, for example using capital letter in the t-condiitions and the lower case in the corresponding r-events.

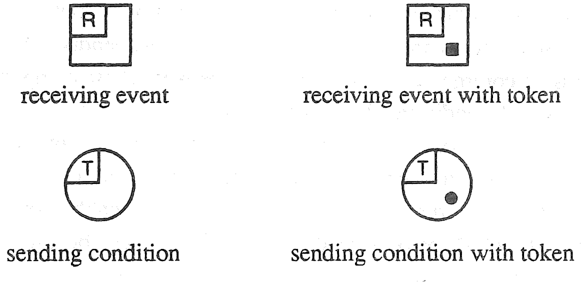


Fig. 2

7. Discussion of the extended net model

The proposed net model can be considered from many points of view. But perhaps the most striking aspect is the introduction of tokens in committed events. This introduction has some pleasant consequence: messages became side effects, simple causality relations are naturally modelled. Moreover, some form of independence has been modelled between the behaviour of transmitter and receiver systems as well as a form of dependence among them. Our choice makes the sending of a message possible at any time even if the preceeding ones have not been consumed on the same elementary channels. The presence of tokens in some sense only indicated that the communication is used instrumentally and has necessarily some form of memory (messages travelling).

8. System analysis, design and implementation with Communicating Petri Nets

Analyzing systems with our model is a rather simple matter. Components have to be identified. These have to be modelled as CE o EN nets, or with any similar formalism. Initial marking has to be given. The Communication Structure has then to be identified by listing the elementary channels. Simulation or other knowledge elicitation means has now to be used in order to verify the correctness of the model, for example comparing his mathematical properties with abstractions of the real system.
The identification of the components in an analytical work could be a matter of taste. Where the system is build of separately described objects, then the best policy it is probably to consider these as parts. In these cases choices are less arbitrary and results more reliable. It is to be noted nevertheless that a good analysis of a systems does not introduce arbitrary choices, but possibly classes of behaviour.
For example an elevator is made of a cabin, doors, buttons with various purposes , lights, a platform to detect presence of people in the elevator, a control unit and some users. Each of these objects can be described with nets (Appendix). And the overall behaviour depends on communication (interactions) between these components.
The cabine has a set of conditions describing its state (i. e., where it is). Transitions from the various states are possible and are due to elementary messages coming from a centralized control unit or from decentralized components entering in suitable states. Doors (one for each state of the elevator) will be blocked, openable, open. There will be buttons to call the elevator outside the doors. Others will be used to select the destination place. Buttons are a two-state entity. Light are used for various purposes and are displaced inside and outside the elevator. Outside the elevator the light will be located at each starting and destination place. The state of the platform indicates if people is inside the cabine. People will require services identified by the starting and

destination places. The control-unit will eventually centralize requests from the starting places and will send messages to the components in order to make them change their state according a good policy. The elevator will be correct if it will correspond to the real system modelled. In the case of a design the elevator model will be correct if it will model the requested services: these can be described by a couple of locations: the starting and arrival location of all the users acting concurrently.
Having obtained a good specification of a concurrent system the problem of *implementing* it is related to the nature of components used to implement the system considered. In fig. 3 a Set-Reset Flip Flop is modelled. In fig. 3.1 and 3.2 transitions due to an input signal are shown. In fig. 3.4 the flip obtained using standard switching synthesis procedures is shown. It is outside the objective of this paper to suggest general implementation methodologies starting from a net description. Nevertheless, something can be said quite easily. In the case some component is naturally implemented (the cabin....) nothing has to be done. The control unit must be synthesized, for example using digital circuits with the approaches implied in fig. 2.. It is a matter of activity partially feasible by automated processing the problem of building some circuits or software realization of the control unit itself starting from the full specification of the entire system with CP Nets The reader can exercize itself implementing a lift as modeled in fig. 4 and fig. 5.

9. Object orientation in Communicating Petri Nets

Real system are frequently made by components that can be described in more general common terms. The term class applies naturally to these cases. Reasoning with our nets, the classes are nets (communicating Petri Nets or other nets). Actual instantiation of classes gives rise to actual objects that in our terms correspond, for example, to components. Actual object behave according to their methods (the net description). Messages are sent through objects.
The inheritance relation among classes and actual objects requires a bit of attention in the case of nets: infact such a relation can be modelled in terms of morphisms!
What we have seen in the present paper strongly supports the evidence that object orientation is particularly suitable to improve the expressiveness of Petri Nets as well as for improving the possibility of using Petri Net theoretical concept for a deeper understanding of object orientation in presence of concurrency.
We hope that this paper has given a first insight in such an interesting matter.

Acknowledgements

Sincere thanks are due to Angelo Beltramini, Anatol Holt, Giorgio De Michelis and Astasia Pagnoni for our many discussions about Petri Nets.

References

1. A. Beltramini, G. Degli Antoni, P. Perego,
   "Petri nets and Switching Circuits, In Petri Nets and Related System Models", Newsletter 18, October 1984. Published by Gesellschaft fur Informatik, Bonn.

2. C.A. Petri
   "Concurrency as a Basis of Systems Thinking. In Finn V. Jensen; Brian H. Mayoh, Karen K. Moller (Eds): Proceedings from the 5th Scandinavian Logic Symposium, Aalborg, January 1979.

3 P.S. Thiagarajan,
   "Elementary net systems". In W. Brauer, W. Reisig and G. Rozenberg (Eds.): Petri Nets: Central Models and Their Properties, Advanced in Petri Nets 1986 Part I, Lecture Notes in Computer Scienze (Vol. 254). Berlin: Springer Verlag, 1987.

4 Spencer Brown
   "The Laws of Form",
   Pinguin Book

Note added in Proof

The author is aware of efforts directed to introduce object orientation in Petri Ne modelling by product of this note. Some of them is reported here.

- Y. Chongyi Towards a general solution of system automation: A net specification of a N_Lift System. GMD, St. Augustin 1, West Germany, January 1990.

- C. Ghezzi, D. Mandrioli On Eclecticism in Specification: A Case Study Centered Around Petri Nets, Techical report 87 - 008, Politecnico Milano.

- D. Buchs, N. Guelfi CO-OPN: A Concurrent Object Oriented Petri Nets Approach for System Specification, internal report LRI n°616, UPS Orsay, 1990.

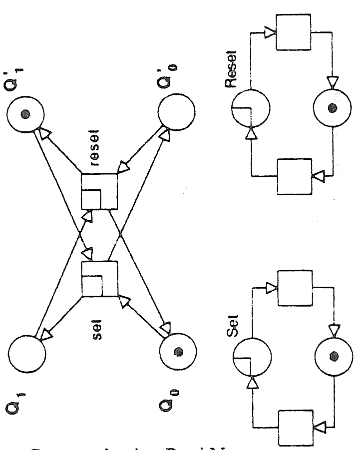


Fig. 3.1

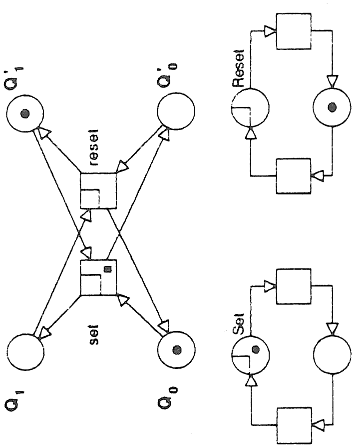


Fig. 3.2

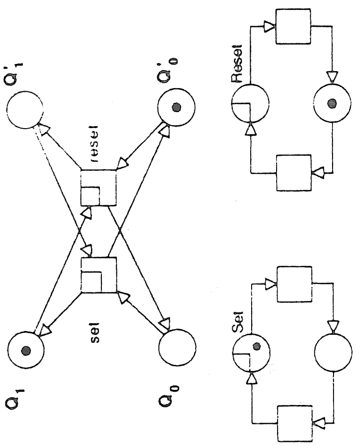


Fig. 3.3

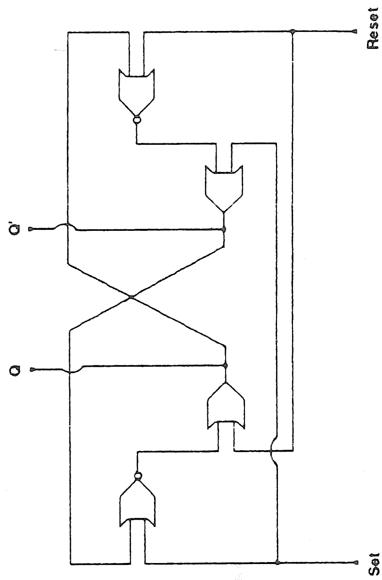


Fig. 3.4

Fig. 3

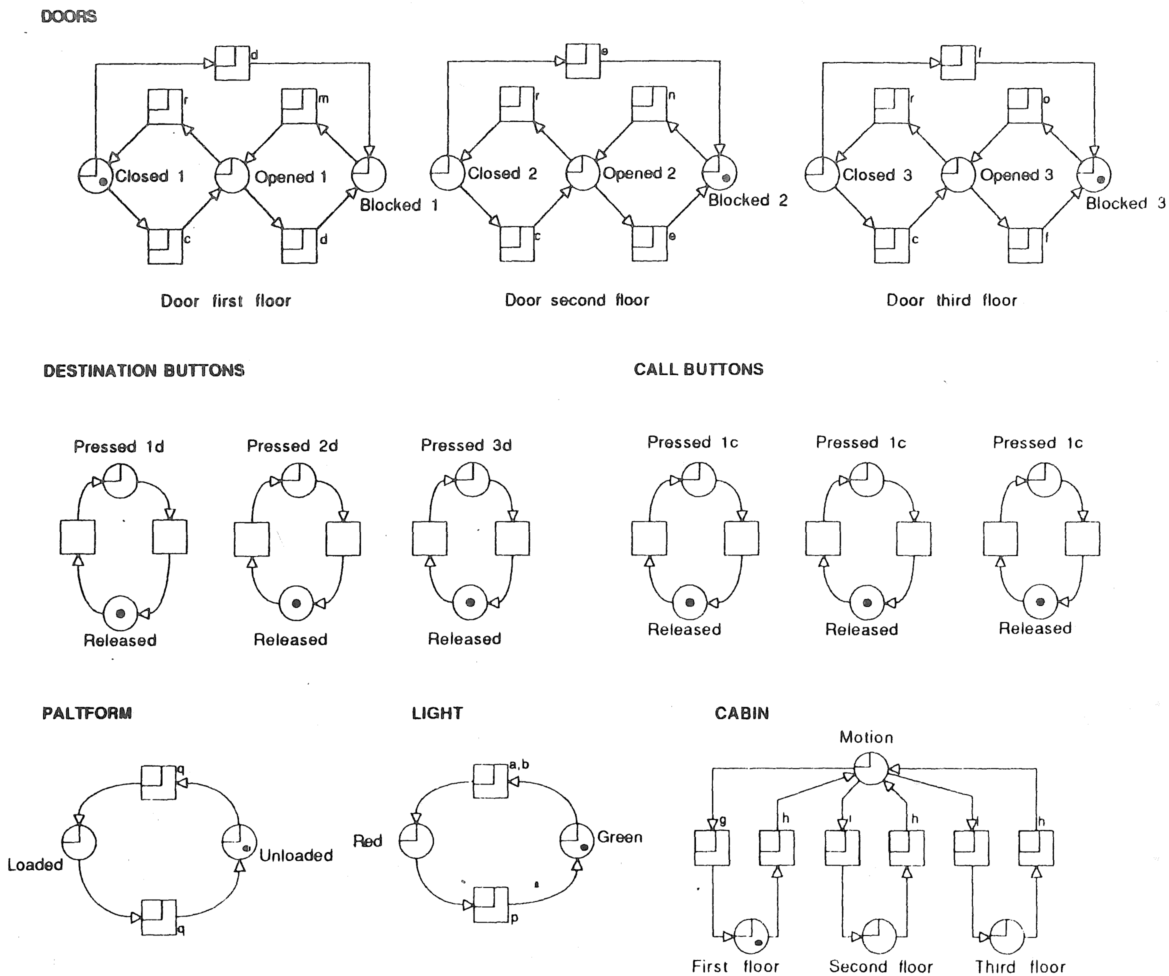
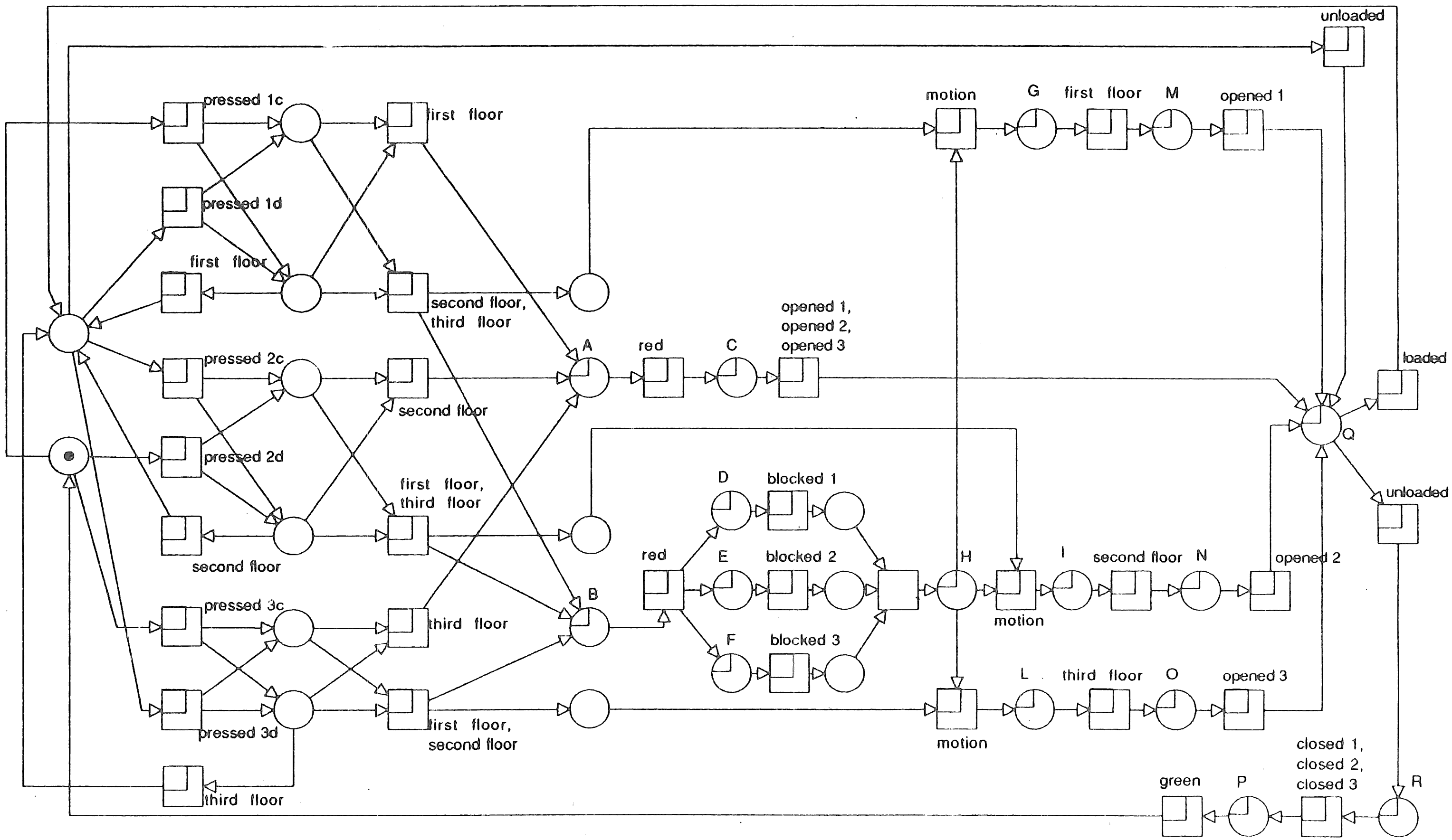


Fig 4

A CPN model for the lift

CONTROL UNIT



CPN model for the Control Unit of the lift

Fig. 5