DECISION PROCESS PETRI NET ANALYSIS

Fred D.J. Bowden
Land Operations Division
Defence Science and Technology Organisation
PO Box 1500, Salisbury, SA 5113, AUSTRALIA
+61 8 8259 5205
fred.bowden@dsto.defence.gov.au
and
Charles E.M. Pearce
Applied Mathematics Department
The University of Adelaide
Adelaide SA 5005, AUSTRALIA
+61 8 8303 5412
cpearce@maths.adelaide.edu.au

ABSTRACT

When modelling human (or non-human) decisions there are often a number of different outcomes that may be reached for the same initial state. Although extensive work has been done on determining the probability of reaching a given state there does not appear to have been any work done on calculating the mean reward of reaching a given state, when there is no guarantee to reach it in finite time. Here, a special class of Petri nets called decision process Petri nets is analysed to determine the mean absorption rewards of decisions, given the decision is reached. This analysis is applicable not only to decision processes, but to any system where there is an interest in the mean reward gained reaching a given state where there is no certainty that the state will be reached.

1. Introduction

In the modelling of any system a certain amount of abstraction is required, often necessitating stochastic ingredients. This is particularly true in the case of modelling human (or non-human) decision processes as such systems are too complex to represent completely. Also, realistic modelling of decision processes requires uncertainty in the way a model reacts to a given situation. Thus for an initial state there could be a number of possible final decisions or absorbing states that may be reached. This paper considers the calculation of the probability of reaching a given absorbing state and the mean reward (cost) until absorption in the given state. The modelling and analysis approach outlined in this paper is to be applied in two areas of command and control studies: course of action analysis; and decision model analysis and validation.

In considering possible courses of action it is important that the overall cost of each action is considered. That is, the *mean conditional reward*. Thus the theories defined in this paper can be used to determine the “best” course of action. The use of rewards means that any measure can be used to determine what “best” means for given situations. For example, the rewards may relate to

the time taken to conduct a course of action, the resources used in a course of action, the possible risk to personnel to carry out a course of action or any combination of these.

When developing models of decision processes, either for inclusion in a larger model or to be directly analysed, two important measures are the probability of reaching a given decision and the mean time to reach that decision. These values are particularly useful to any subject matter adviser who might be validating a model as the values can easily be related back to the real system without any knowledge of the mathematical detail of the model. They are also valuable measures in analysing any decision process directly. The use of these types of measures to validate decision processes is a further extension of the validation methods outlined in [Bowden *et al.*, 1995].

Although extensive work has been performed on determining the probability of reaching a given state, there does not appear to have been any work performed on calculating the mean reward of reaching a given state when one is not guaranteed to reach it. In this paper a special class of Petri nets (PN) called decision process Petri nets are analysed to determine the mean rewards of decisions given the decision is reached. This analysis is applicable not only to decision processes, but also to any system where there is an interest in the mean reward to reach a given state where there is no certainty that the state of interest will be reached.

This paper introduces a basic building block that can be used to construct complex decision process models. It uses an example to illustrate how this building block can be can be used to model a decision process. Then it analyses the decision process to determine the probability and mean cost (in terms of time taken) to reach each of the possible final states. Finally some areas of future work are outlined.

2. Petri Net Background

PNs were originally developed by Carl Adam Petri in his doctoral thesis titled “Communication with Automata” [Petri, 1962]. A PN is a bipartite directed graph with two types of nodes: *places* and *transitions*. Pictorially, places are indicated by circles and represent entities such as conditions, buffers, servers, resources and queues. Transitions are displayed on the graph as rectangles and represent concepts in a real system such as processors, algorithms and events. Arcs can go only from a place to a transition or *vice versa*. In this paper an arc from a place to a transition is referred to as an *input arc* and the places with an input arc to a given transition are its *input places*. An arc from a transition to a place is called an *output arc* and the places with an output arc from a particular transition are referred to as the transition’s *output places*.

Tokens make up the final element in a PN. Tokens are represented graphically by identical dots and can only occur in places. The movement of tokens between places is controlled by the transitions of the PN. In a model, the position of the tokens defines the state of the system, defining situations such as satisfied conditions, items in a buffer, the number of free servers, availability of resources and entities in a queue. The distribution of tokens through the PN is defined as the PN’s marking. It is the PN’s marking that defines the current state of the system being modelled.

The dynamics of the model are represented by the movement of tokens, which is described by the transition firing rules. A transition is said to be *enabled* by a given marking if all its input places have at least one token for each input arc from the place to the transition. An enabled transition can *fire* (occur). In firing a transition removes one token from its input places for each input arc from that place and creates one token in each of its output places for each output arc to that place. A more detailed overview of PNs can be found in [Peterson, 1981] and [Reisig, 1982].

There are two ways of representing time in PNs; holding times ([Ramchandani, 1973]) and enabling times ([Merlin, 1974]). With holding times, tokens are reserved for the firing of a given transition. Such reserved tokens, called unavailable tokens (as opposed to available tokens), cannot be used to enable other transitions. The time that a given token remains unavailable can depend on the transition ([Ramchandani, 1973]), the output arc that created the tokens ([Murata, 1996]) or the place at which the token was created ([Sifakis, 1977]). With enabling times the transitions compete for the tokens is a way reminiscent of a birth and death process. The transition that fires first gets to use the tokens disabling any transitions that are in conflict with the firing transition. This is called the race conflict resolution policy or just the race policy. Enabling times have been assigned to both input arcs ([Nowicki and Wilczkowiak, 1989]) and transitions ([Merlin, 1974]). A more detailed discussion of the different ways of representing time in PNs can be found in [Bowden, 1996].

These representations came from different requirements. Holding times are used in systems where resources are held for a given period of time before being free for use by other processes, while enabling times allow the modelling of such interruptions as time outs.

To allow for the current representations of time, a new time PN called the super-class timed PN (STPN), was presented in [Bowden, 2000]. This representation assigns enabling times to input arcs and holding times to output arcs, allowing for the representation of both interruptions and the holding of resources in a flexible way. It is this representation that will be used in this paper.

Figure 1 shows a simple STPN in which deterministic delays have been used. In this STPN transitions t1 and t2 are enabled with the marking shown. Transition t1 must be enabled for three time units before it can fire as this is the maximum enabling time on its input arcs, given by the arc from place p1. Transition t2 must be enabled for four time units before it can fire. This delay is determined by its only input arc. Thus transition t1 fires at time three, removing one token from places p1 and p2 and creating an unavailable token in places p4 and p5. The token created in p4 is unavailable for seven time units, while the one in place p5 is unavailable for two time units. Thus the next event is the firing of transition t2 when the clock reaches four. This creates one unavailable token in places p5 and p6. The token in place p5 is unavailable for two time units and the token in place p6 for six time units. Thus the final events occur when the unavailable tokens become available. The first to become available are the tokens in place p5 at times five and six. Then the unavailable tokens in places p4 and p6 become available at time 10. This example is a very simple STPN with deterministic delays. The STPNs considered later have arbitrary delay functions.

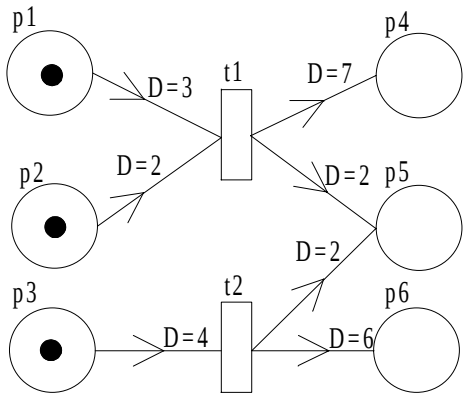


Figure 1: Simple super-class timed Petri net

A further extension of this idea is to include rewards. This is done here much as presented in [Ciardo *et al.*, 1993]. In the analysis considered in this paper reward functions are assigned to each arc similarly to the way time was assigned to arcs.

In the past PNs have been used to carry out many different types of performance analysis. In some papers, such as [Ajmone Marsan *et al.* 1995], [German, 1995], [Haas and Shedler, 1986] and [Lindemann and Shedler, 1996], PN models with Markovian state spaces are constructed. This allows the use of Markovian techniques to analyse the PN's state space. Other papers, such as [Berthomieu and Diaz, 1991] and [van der Aalst and Odijk, 1995], employ interval timings and use the state space to generate performance bounds. Still others, [Baccelli *et al.* 1992], have developed analysis techniques that operate directly on sub-classes of PNs. The advantage of working directly with the PN model rather than using the state space for analysis is that the problem of state space explosion can be avoided. Since generally the PN models of interest are of a given restricted class, it is possible to develop analysis techniques that can be applied directly to this type of PNs. This is made possible by the Markovian nature of the PN building blocks used to develop the decision process models. Thus the technique presented in used here falls into the latter type.

3. Decision Process Definition

A decision process is defined in this paper as a series of decisions and actions that lead a decision maker (of human or other form) to a final decision or situation. The aim of this section is to develop a basic building block that can be used to build complex decision processes. The building block also needs to be able to be directly analysed to determine the probability of reaching a given decision and the conditional mean reward to reach that state. The building block approach is based on that presented in [Bowden and Davies, 1997], although the structures presented in this paper are more restrictive to allow direct analysis to the building blocks.

3.1 Example

Consider the case of a military unit which is given the mission of finding and destroying a particular target. Initially the unit seeks to find the designated target. There are four possible outcomes of this searching: the friendly unit is found and neutralised by the enemy, the target is

located, the desired target is not found, and the desired target is not found and the unit decides to rest before continuing its search. Once the target is located a battle begins. There are three possible outcomes of such a battle: the friendly unit is neutralised, the target is destroyed or no resolution is reached and the battle continues. Figure 2 shows a PN model of this decision process. Each of the eight component processes is represented by a transition in the PN. Table 1 indicates which transition corresponds to which process. With the initial marking shown in Figure 2 there are three possible absorbing states (a token in p2, p5 or p6). These relate to the three possible outcomes of the system: the friendly unit is neutralised before it locates its target, the friendly unit is neutralised by the target during the battle, or the mission is successfully completed.

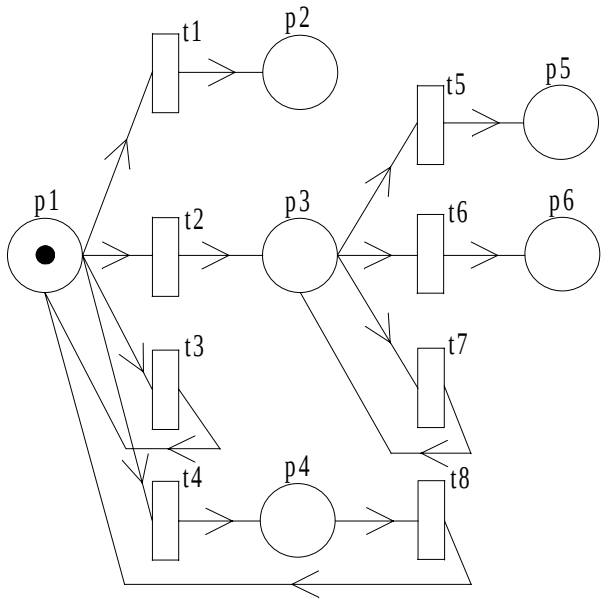


Figure 2: Sample decision process

The PN in Figure 2 is an example of the type of model that will be considered in this paper. This example will be used to illustrate the analysis techniques that are developed.

3.2 The Decision Process Building Block

The basic building block that underlies the decision processes is made up of two parts. One comprises a number of decisions and actions that take the system from its initial state back to the initial state. In the example above such a sequence of events is that where the friendly unit is unable to find the target so decides to rest before continuing to search. The second part of the decision process involves a set of possible decisions (outcomes) and/or actions, that move the system one step closer to a final decision (an absorbing marking in PN terms). The location of the target by the friendly unit is an example of an action that shifts the systems towards an absorbing marking.

Table 1: Descriptions of transitions in Figure 2

Transition	Description
t1	The friendly unit is located and neutralised by the enemy.
t2	The target is located and the friendly unit prepares to attack.
t3	The desired target is not located and the friendly unit continues to search.
t4	The desired target is not located so the unit decides to rest before continuing its search.
t5	The friendly unit is neutralised during the battle.
t6	The enemy unit is neutralised during the battle.
t7	No resolution is reach and the battle continues.
t8	The unit rests and then continues the search.

In terms of PNs the decision process building block is made up of a series of T-invariants[1] and a number of possible absorbing states. Here only legal firing sequences that return the PN to its independent marking are considered T-invariants. Figure 3 shows the decision process building block. The system comprises N T-invariants each starting with a transition Q_i, where i = 1 to N. The T-invariants are referred to as the Q part of the system. The T-invariant which begins with Q_i is called the ith T-invariant. In addition to this there are K other transitions referred to as absorbing transitions. Each absorbing transition leads to an absorbing state of the decision process building block, so there are K possible absorbing states. The jth absorbing state is reached when a token is in place A_j, which occurs when the transition R_j fires, where j = 1 to K. The marking with one token in place A_j is referred to as state j. Each of the absorbing states relates to an outcome that can be reached by the decision process. The absorbing part of the decision process will be referred to as the R part of the system. The place A_j will be referred to at times as absorbing place j. The initial state of this decision process is where a token is in place I. This will be referred to as state I and is considered part of the Q system.

The example given in Figure 2 has two building blocks. These are shown separately in Figure 4. The initial building block, shown in Figure 4(a), has N = K = 2. Since K = 2 there are two absorbing markings, one when a token is in place p2 and the other when a token is in place p3. That is, two outcomes can be reached; the friendly unit is neutralised or the target is located. The second building block is shown in Figure 4(b) and has N = 1 and K = 2. Once more there are two absorbing states; one when a token is in place p5, representing the case when the friendly unit is neutralised during the battle, and the other when a token is in place p6, representing the case when the mission is successfully completed.

[1] A T-invariant is a sequence of transition firings that returns the PN to its starting marking[van der Aalst and Obijk, 1995].

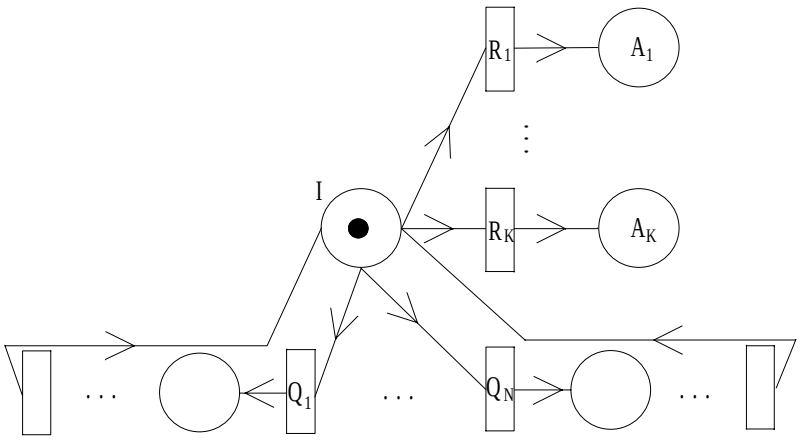
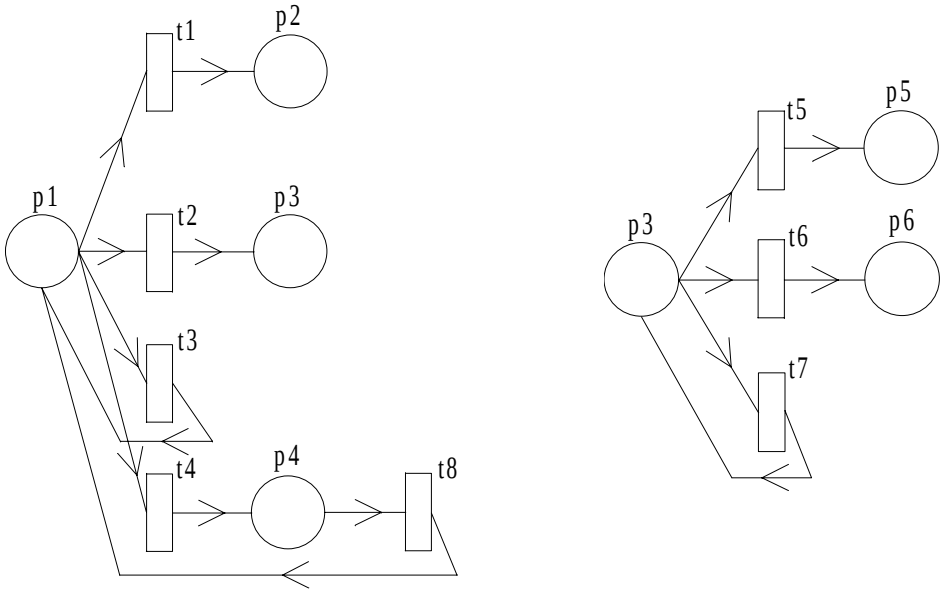


Figure 3: The decision process building block



(a) Initial building block (b) Final building block

Figure 4: Decision process building blocks for Figure 2

Developing an analysis technique for the decision process basic building block makes it possible to produce an iterative method of calculating the absorption probability and mean conditional absorption time for each of the absorbing states of a total system. Appendix I derives techniques developed to calculate the probability that the final marking has one token in place A_j as well as the mean reward gained in reaching this state, given it is reached. These results can then be used iteratively to analyse the whole system as shown in the next section.

4. Application of the Theory

The example decision process shown in Figure 2 will now be used to illustrate how such a system can be analysed. The theories used to carry out this analysis are given in Appendix I. The

values associated with the transitions are given in Table 2. The enabling times of transitions t1 to t4 relate to the time spent by the friendly unit searching for its target. In the case of t4 more time is spent searching as once the decision to rest is made there will be no further searching for a relatively long period of time. The holding time of transition t1 reflects the time it takes for the friendly unit to be neutralised by the enemy. The holding time of transition t2 relates to the time the unit spends preparing to attack the target. The holding time of t3 is the time taken to determine where next to search for the enemy. The holding time of transition t4 is the time spent moving into a defensive position. The enabling times of transitions t5 to t7 are the times spent in battle between the friendly unit and the target. If no outcome is reached then the holding time of t7 is the time between direct exchanges. If the friendly unit is neutralised then the holding time of transition t5 is the time taken to confirm the neutralisation of the friendly unit and vice versa in the case of transition t6. The enabling and holding times of transition t8 relate to the time spent resting and moving out of their defensive position.

Table 2: Sample mean enabling and holding delays for the STPN in Figure 2

Transition	Enabling time		Mean holding time (τ_i)
	Distribution	Parameters	
t1	Exponential	$\mu_1 = 1/5$	$\phi_1 = 10$
t2	Exponential	$\mu_2 = 1/5$	$\phi_2 = 10$
t3	Exponential	$\lambda_3 = 1/5$	$\tau_3 = 5$
t4	Exponential	$\lambda_4 = 1/50$	$\tau_4 = 50$
t5	Exponential	$\mu_5 = 1/10$	$\phi_5 = 10$
t6	Exponential	$\mu_6 = 1/10$	$\phi_6 = 10$
t7	Gamma	$\alpha_7 = 2, \lambda_7 = 1/10$	$\tau_7 = 20$
t8	-	$\lambda_8 = 1/50$	$\tau_8 = 100$

The results for the decision process building block shown in Figure 4(a) are given in Table 3. The results for the second decision process building block shown in Figure 4(b) are given in Table 4. For these tables the values for P(j) are calculated using Corollary 2[2] while those for T_I are calculated using Corollary 3[2].

Table 3: Results for STPN in Figure 4(a) for the values in Table 2

Decision (state j is reached)	Probability of reaching decision (P(j))	Mean conditional time to reach decision (T_{p1})
The friendly unit was neutralised before it located the target (state 2).	½	$T_{p1} = 7$
The target located (state 3).	½	$T_{p1}(3) = 7$

[2] See Appendix I.

Table 4: Results for STPN in Figure 4(b) for the values in Table 2

Decision (state j is reached)	Probability of reaching decision (P(j))	Mean conditional time to reach decision (T_{p3})
The friendly unit was neutralised during the battle (state 5').	½	$T_{p3}(5) = 10.2$
The mission was completed (state 6').	½	$T_{p3}(6) = 10.2$

As earlier, the results in Table 4 are the values for the probability and mean time given for the PN in Figure 4(b) where the initial marking is one token in place p3. Thus, these results relate only to the movement of the token from place p3 to either place p5 or place p6. By combining the results in Table 3 and Table 4, we can derive the overall results of the example decision process shown in Figure 2. The overall results are given in Table 5.

Table 5: Results for STPN in Figure 2 for the values in Table 2

Decision (state j is reached)	Probability of reaching decision (P(j))	Mean conditional time to reach decision (T_{p1})
The friendly unit was neutralised before it found the target (state 2).	$P(2) = 1/2$	$T_{p1} = 7$
The friendly unit was neutralised during the battle (state 5).	$P(5) = P(3) \times P(5') = 1/4$	$T_{p1}(3) + T_{p3}(5) = 17.2$
The mission was completed (state 6).	$P(6) = P(3) \times P(6') = 1/4$	$T_{p1}(3) + T_{p3}(6) = 17.2$

5. Conclusions and Future Work

The main problem with the technique developed in this paper is that it can only be applied to a restricted subset of STPNs. One way around this is to use the state space of the STPN. If the state space forms a discrete or continuous time Markov chain, then the results in [Bowden and Pearce, 1999], [Bowden and Pearce, 2000a] and [Bowden and Pearce, 2000b] can be used to determine the mean conditional absorption time. Future work is aimed at expanding this type of analysis both directly on the PN and on the state space. Already the decision process building block has been extended. These extensions include T-invariants within T-invariants, T-invariants with absorbing states, and crossovers where there can be more than one way of reaching the initial place of the next building block.

It should be noted that although the results presented here are specified as giving the mean conditional absorption reward, they are actually more general than this. These results relate to the mean conditional first hitting reward. There is no requirement in any of the theory presented here that the place reached be an absorbing place.

As mentioned earlier there are two planned primary uses of the results presented in this paper. The first is the application of the absorption probability and mean conditional first hitting reward

to the course of action analysis. These two results[3], along with the other results presented here, can be used to determine the probability and conditional mean reward (cost) of reaching a given state. The second application for these results is in the validation and analysis of decision processes. This may be used for direct analysis of decision processes or to aid in the validation of models that are to be included as part of larger models, or command and control modelling environments such as those described in [Bowden *et al.*, 1997], [Bowden *et al.*, 2000] and [Ashton *el al.*, 2000].

6. References

[van der Aalst and Odijk, 1995] W.M.P. van der Aalst and M.A. Odijk. *Analysis of Railway Stations by Means of Interval Timed Coloured Petri Nets*. Real-Timed Systems, Vol. 9, 1995, pp. 241-263.

[Ajmone Marsan *et al.* 1995] M. Ajmone Marsan, G. Balbo, G. Conte, S. Donatelli, G. Franceschinis. *Modelling with Generalized Stochastic Petri Nets*. Wiley Series in Parallel Computing, John Wiley & Sons Ltd. 1995.

[Ashton *et al.*, 2000] K. Ashton, F. Bowden, A-M. Grisogono, W. Johnson, D. Krause, J. Krieg, J. Lorke, F. Lui, I. Menadue, G. Pearce, S. Robertson, D. Sands, R. Seymour, M. Scholz and J. Vaughan. *Implementation Of The Land C4ISR Synthetic Environment.* Pesented at SimTecT 2000, Sydney, Australia, February 2000.

[Baccelli *et al.* 1992] F.L. Baccelli, G. Cohen, G.J. Olsder, and J. Quadrat. *Synchronization and Linearity: An Algebra for Discrete Event Systems.* John Wiley and Sons, 1992.

[Berthomieu and Diaz, 1991] B. Berthomieu and M. Diaz. *Modeling and Verificaion of Time Dependent Systems Using Time Petri Nets.* IEEE Trans. on Software Engineering, Vol. 17, No. 3, March 1991, pp. 259-273.

[Bowden, 2000] F.D.J. Bowden. *A Brief Survey and Synthesis of the Roles of Time in Petri Nets.* Accepted for publication for Mathematical and Computer Modelling.

[Bowden, 1996] F.D.J. Bowden. *Modelling Time in Petri Nets*. proceedings of the Second Australia-Japan Workshop on Stochastic Models, Gold Coast, July 1996.

[Bowden and Davies, 1997] F.D.J. Bowden and M. Davies. *Application of a Role-Based Methodology to Represent Command and Control Processes Using Extended Petri Nets.* Proceedings of the 1997 IEEE International Conference on Systems, Man and Cybernetics, Orlando, Florida, USA, October 12-15, 1997, pp. 4348-4353.

[Bowden *et al.*, 1995] F.D.J. Bowden, M. Davies and J.M. Dunn. *Representing Role-Based Agents Using Coloured Petri Nets*. The Proceedings of the 5th Conference on Computer Generated Forces and Behavioural Representation, May 1995, Orlando, Florida, pp. 513-520.

[Bowden *et al.*, 1997] F.D.J. Bowden, C. Gabrisch and M. Davies. *C3I Systems Analysis Using The Distributed Interactive C3I Effectiveness (DICE) Simulation*. Proceedings of the 1997 IEEE International Conference on Systems, Man and Cybernetics, Orlando, Florida, USA, October 12-15, 1997, pp. 4326-4331.

[Bowden *et al.*, 2000] F. Bowden, P. Gaertner and P. Williams. *A Near Real-Time Tactical Land C4I Assessment Capability*. Pesented at SimTecT 2000, Sydney, Australia, February 2000.

[Bowden and Pearce, 2000a] F.D.J. Bowden and C.E.M. Pearce. *Conditional Rewards in Markov Processes*. Submitted.

[3] As given in Appendix I.

[Bowden and Pearce, 2000b] F.D.J. Bowden and C.E.M. Pearce. *Conditional Hitting Times in Markov Processes*. Submitted.

[Bowden and Pearce, 1999] F.D.J. Bowden and C.E.M. Pearce. *Some Questions Involving First-Passage Problems in Markov Processes*. Proceedings of the 1999 National Conference of the Australian Society for Operations Research.

[Ciardo *et al.*, 1993] G. Ciardo, A. Blakemore, P.F. Chimento, J.K. Muppala and K.S. Trivedi. *Automated Generation and Analysis of Markov Reward Models Using Stochastic Reward Models*. Linear Algebra, Markov Chains, and Queueing Models, C. Meyer and R.J. Plemmons, Eds. Vol. 48 of IMA Volumes in Mathematics and its Applications, New York: Springer-Verlag, 1993, pp. 145-191

[German, 1995] R. German. *Transient Analysis of Deterministic and Stochastic Petri Nets by the Method of Supplementary Variables*. Quality of Communication-Based Systems. Kluwer Academic Publishers, 1995, pp 105-121.

[Haas and Shedler, 1986] P.J. Haas and G.S. Shedler. *Regenerative Stochastic Petri Nets.* Performance Evaluation, Vol. 6, 1986, pp. 189-204.

[Lindemann and Shedler, 1996] C. Lindemann and G.S. Shedler. *Numerical analysis of deterministic and stochastic Petri nets with concurrent deterministic transitions*. Performance Evaluation Journal, 27&28, 1996, pp. 565-582.

[Karlin, 1966] S. Karlin. *A first course in stochastic processes*. Academic Press, New York, 1966.

[Merlin, 1974] P.M. Merlin. *A Study of recoverability of computer Systems*. PhD. Thesis, Computer Science University of California IRVINE, 1974.

[Murata, 1996] T. Murata. *Temporal Uncertainty and Fuzzy-Timing High-Level Petri Nets.* Lecture Notes in Computer Science 1091, Application and Theory of Petri Nets 1996, Proceedings of 17th. International Conference, Osaka, Japan, June, 1996 pp. 11-28.

[Nowicki and Wilczkowiak, 1989] T. Nowicki and E. Wilczkowiak. *Study of Local Area Networks Using Petri Nets., Microelectronics and Reliability*. Vol.29, No.3, Pergamon Press, 1989, pp. 433-437.

[Peterson, 1981] J.L. Peterson. *Petri Net Theory and the Modeling of Systems.* Prentice-Hall, Inc., Enlgewood Cliffs, N.J., 1981.

[Petri, 1962] C. Petri. *Kommunikaiton mit Automaten.* PhD. dissertation, University of Bonn, West Germany, 1962.

[Ramchandani, 1973] C. Ramchandani. *Analysis of Asynchronous Concurrent Systems by Petri Nets*. PhD. Thesis, MIT, July 1973.

[Reisig, 1982] W. Reisig. *Petri Nets, An Introduction*. EATCS, Monographs on Theoretical Computer Science, Springer Verlag, Berlin (1985). Translation of the original German edition, "Petrinetze" (1982).

[Sifakis, 1977] J. Sifakis. *Use of Petri Nets for Performance Evaluation.* Measuring, Modelling and Evaluating Computer Systems, ed H. Beilner and E. Gelenbe, North Holand, 1977, pp. 75-93.

Appendix I. Analysing Decision Processes

This appendix defines the theories for the analysis of the decision process building blocks. Initially it presents two important Markov chain results that are key to the direct analysis.

I.1 Markov Chain Analysis

As stated in Section 2, the main way of doing analysis in a PN is on its state space. In the case of PNs with stochastic timings this generally means Markovian analysis. The analysis presented here on the decision process building block uses a number of results from the analysis of Markovian state spaces. These will now be presented. Futher details and proofs of these results can be found in [Bowden and Pearce, 1999], [Bowden and Pearce, 2000a] and [Bowden and Pearce, 2000b].

Theorem 1
Let κ be a nonempty set of states in a discrete-time Markov chain and B the set of states which have access to κ but do not belong to κ. Suppose B is also nonempty. For $i \in B$, denote by a_i the probability that for initial state i the process ever reaches κ. Then $(a_i)_{i \in B}$ is the smallest non-negative solution to the equations

$$a_i = \sum_{j \in \kappa} p_{ij} + \sum_{j \in B} p_{ij} a_j \quad (i \in B),$$

where p_{ij} is the one-step transition probability from state i to state j in the discrete-time Markov chain.

This theorem can be applied to any Markov process by defining the underlying discrete-time Markov chain as set out in [Karlin, 1966].

Theorem 2
Let κ be a nonempty set of states in a semi-Markov chain and let B be the set of all states which have access to κ but do not belong to κ. Suppose B is nonempty. For $i, j \in B$ $(i \neq j)$ put

$$U_{ij} := \frac{p_{ij} a_j}{a_i}.$$

Also

$$R_{ij} := \int_0^\infty r_{ij}(t)\, dF_{ij}(t)$$

as the reward gained from the transition from state i to state j given the transition occurs, where $r_{ij}(t)$ is the reward gained by spending time t in state i before moving to state j and $F_{ij}(t)$ is the probability that the time to go from state i to state j is t conditional on going from i to j in one step. If R_i is the mean conditional first-passage reward to κ from initial state $i \in B$, then $(R_i)_{i \in B}$ satisfies

$$R_i = \frac{v_i}{a_i} + \sum_{j \in B} U_{ij} R_j \quad (i \in B)$$

where

$$v_i = \sum_{j \in B \cup \kappa} p_{ij} a_j R_{ij}$$

is the reward given by the next step and $a_i := 1$ for $j \in \kappa$.

I.2 Analysing decision processes

Consider the case when time in the decision process building block is represented by enabling durations associated with input arcs and holding durations associated with output arcs. Thus STPNs are used.

To simplify the analysis, initially a simpler building block will be used. In this case the T-invariants all involve a single transition, as shown in Figure 5. The reason for this simplification will become evident later.

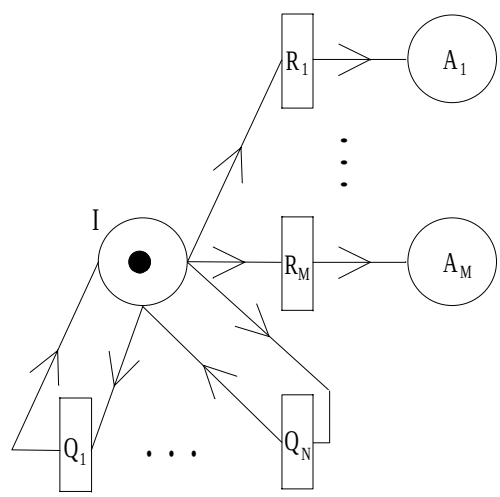


Figure 5: Simple decision process building block

The state space of the simple decision process building block is very similar to the PN, see Figure 6. This similarity is due to the Markovian nature of the building block. Thus the Markov chain results given in Section 0 can be used to develop theories on the absorption probabilities and the mean conditional rewards for the state space and thus the STPN building block.

The delay (time) functions associated with the arcs will be referred to in terms of their transitions. This can be done without confusion as each transition has at most one input and output arc. Let $E_{q\,i}(t)$ (i = 1 to N) and $E_{r\,j}(t)$ (j = 1 to K) be the enabling delay functions for the Q and R transitions, respectively. Also define $H_{q\,i}(t)$ (i = 1 to N) as the holding delay function for transition Q_i and let $H_{r\,j}(t)$ (j = 1 to K) be the holding delay function for transition R_j. For this analysis the race policy will be used to resolve conflict and enabling memories will be used as the memory policy.

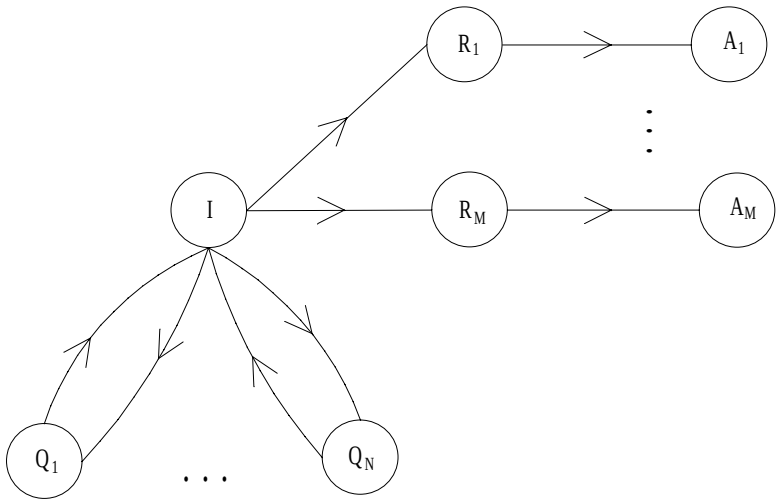


Figure 6: Simple decision process building block state space

In considering the states of the simple decision process building block, state i is defined as a state in Q_1 to Q_N. Similarly j is used to describe states in R_1 to R_M. In this case what is of interest is the probability and mean conditional first passage reward to reach state A_k, for $k \in [1, \ldots, M]$.

Define $\Omega(t)$:= P(no transition fires before time t). Then

$$\Omega(t) = \left\{\prod_i \left(1 - E_{qi}(t)\right)\right\} \times \left\{\prod_j \left(1 - E_{rj}(t)\right)\right\}.$$

Define $I_i(t)$ to be the probability that transition Q_i fires before time t. Similarly define $J_j(t)$ as the probability that transition R_j fires before time t. So,

$$I_i(t) = \int_0^t P\begin{pmatrix} \text{transition } Q_i \text{ fires in the time period } (u,\ u + du) \\ \text{and no transition fires before time u} \end{pmatrix}$$

$$= \int_0^t P(\text{no transitions fire before time u}) \times$$

$$P(\text{transition } Q_i \text{ fires in the time period } (u,\ u + du) \mid \text{no transition before time u}).$$

If there is a positive probability that two or more transitions fire at the same time then the above result does not hold. In these cases the race policy cannot be used to resolve conflict so an extra conflict resolution policy must be specified. If an alternative conflict resolution policy were specified for those transitions that could have identical enabling durations, then the following results could be adjusted to allow for this. This situation will not be considered further.

Expanding the conditional probability in $I_i(t)$ gives

$$I_i(t) = \int_0^t P(\text{no transitions fire before time u}) \times$$

$$\frac{P(\text{transition } Q_i \text{ fires in the time period } (u,\ u + du))}{P(\text{transition } Q_i \text{ does not fire before time u})}.$$

Therefore

$$I_i(t) = \int_0^t \left\{ \prod_{m=1}^{N} \left(1 - E_{qm}(u)\right) \right\} \left\{ \prod_{j=1}^{K} \left(1 - E_{rj}(u)\right) \right\} \frac{dE_{qi}(u)}{\left(1 - E_{qi}(u)\right)}$$

$$= \int_0^t \frac{\Omega(t)\, dE_{qi}(u)}{\left(1 - E_{qi}(u)\right)}.$$

A similar argument gives

$$J_j(t) = \int_0^t \frac{\Omega(u)\ dE_{rj}(u)}{1 - E_{rj}(u)}.$$

The probability that a transition in the Q system fires by time t is given by

$$I(t) = \sum_{i=1}^{N} I_i(t).$$

The probability that a transition in the R system fires by time t is given by

$$J(t) = \sum_{j=1}^{K} J_j(t).$$

This means that the probability that transition Q_i (R_j) fires is given by $I_i(\infty)$ ($J_j(\infty)$). So in terms of the Markov process in Figure 6

$$p_{IQ_i} = I_i(\infty) \text{ and } p_{IR_j} = J_j(\infty).$$

Also the probability that a Q (R) transition fires is $I(\infty)$ ($J(\infty)$). That is,

$$\sum_{i=1}^{N} p_{IQ_i} = I(\infty),\ \sum_{j=1}^{M} p_{IR_j} = J(\infty) \text{ and } I(\infty) + J(\infty) = 1.$$

Now

$$P(\text{i fires at time t or earlier} \mid \text{i fires}) = \frac{P(\text{i fires at time t or earlier} \cap \text{i fires})}{P(\text{i fires})}$$

$$= \frac{P(\text{i fires at time t or earlier})}{P(\text{i fires})}$$

$$= \frac{I_i(t)}{I_i(\infty)}.$$

Similarly

$$P(\text{j fires at time t or earlier} \mid \text{j fires}) = \frac{J_j(t)}{J_j(\infty)} .$$

Theorem 3
In a STPN simple decision process building block the absorption probability of state j, given that the probability that two transition will have the same enabling time is zero, is given by

$$P(j) = \frac{J_j(\infty)}{J(\infty)} . \qquad \textbf{(1)}$$

Proof:
For the state space shown in Figure 6 $\kappa = A_k$ and $B = \{I, Q_1, \ldots, Q_N, R_k\}$, which means Theorem 1 gives

$$a_w = p_{wA_k} + \sum_{x \in B} p_{wx} a_x .$$

For all $x \in B$, $p_{R_k x} = 0$ and $p_{R_k A_k} = 1$, so $a_{R_k} = 1$

Also $p_{Q_i w} = 0$ for all $w \neq I$ and $p_{Q_i I} = 1$ so $a_{Q_i} = p_{Q_i I} a_I = a_I$.

Further $p_{IA_k} = 0$ and so

$$a_I = \sum_{x \in B} p_{Ix} a_x = \sum_{i=1}^{N} p_{IQ_i} a_{Q_i} + p_{IR_k} a_{R_k} = \sum_{i=1}^{N} p_{IQ_i} a_I + p_{IR_k} .$$

So

$$a_I = \frac{p_{IR_k}}{1 - \sum_{i=1}^{N} p_{IQ_i}} = \frac{J_i(\infty)}{1 - I(\infty)} = \frac{J_i(\infty)}{J(\infty)}$$

as required. ■

It should be noted that the probability of reaching a given absorbing state is independent of the circuits. Thus the above result holds not only for the simple decision process building block but also for the more complex ones with non-loop circuits. Accordingly we have the following.

Lemma 1
The probability of reaching the absorbing state j, of a STPN decision process building block is given by Theorem 3.

Theorem 4
The mean first-passage reward for state A_k of a STPN simple decision process building block is given by

$$R_I = \frac{1}{J(\infty)} \sum_{i=1}^{N} \left(\int_0^\infty r_{IQ_i}(t) dI_i(t) + I_i(\infty) \int_0^\infty r_{Q_i I}(t) dH_{qi}(t) \right) +$$

$$\frac{1}{J_k(\infty)} \int_0^\infty r_{IR_k}(t) dJ_k(t) + \int_0^\infty r_{R_k A_k}(t) dH_{rk}(t) \quad ,$$

where $r_{Iw}(t)$ is the reward gained by spending time t in I before a transition w ($\in \{Q_1, \ldots, Q_N\} \cup \{R_k\}$) fires, $r_{Q_i I}(t)$ is the reward gained by the token in I being held for time t after transition Q_i fires and $r_{R_k A_k}(t)$ is the reward gained by the token spending a holding time of t in place A_k after the firing of transition R_k.

Proof:

As with Theorem 3 the results from Section I.1 can be applied to the state space shown in Figure 6 to get the desired result. In this case Theorem 2 is used. Once more $\kappa = A_k$ and $B = \{I, Q_1, \ldots, Q_N, R_k\}$ so applying Theorem 2 gives

$$R_x = \frac{v_x}{a_x} + \sum_{w \in B} U_{xw} R_w \quad \text{where } v_x = \sum_{w \in B \cup \kappa} p_{xw} a_w R_{xw} \quad \text{for all } x \in B.$$

Since $p_{R_k w} = 0$ for all $w \neq A_k$,

$$R_{R_k} = \frac{v_{R_k}}{a_{R_k}} + U_{R_k A_k} R_{A_k} ,$$

where $v_{R_k} = p_{R_k A_k} a_{A_k} R_{R_k A_k}$. Also $p_{R_k A_k} = a_{R_k} = a_{A_k} = 1$ and $R_{A_k} = 0$ so

$$R_{R_k} = R_{R_k A_k} .$$

For $x \in \{Q_1, \ldots, Q_N\}$, $p_{xw} = 0$ for all $w \neq I$ giving

$$R_x = \frac{v_x}{a_x} + U_{xI} R_I$$

where $v_x = p_{xI} a_I R_{xI}$. Also $p_{xI} = 1$ and $a_x = a_I$ so

$$R_x = R_{xI} + R_I .$$

For x=I

$$R_I = \frac{v_I}{a_I} + \sum_{w \in B} U_{Iw} R_w$$

where $v_I = \sum_{w \in B} p_{Iw} a_w R_{Iw}$. Also since $p_{IA_k} = 0$

$$
\begin{aligned}
R_I &= \sum_{w \in B} U_{Iw}\left(R_{Iw} + R_w\right) \\
&= \sum_{i=1}^{N} U_{IQ_i}\left(R_{IQ_i} + R_{Q_i I}\right) + U_{IR_k}\left(R_{IR_k} + R_{R_k}\right) \\
&= \sum_{i=1}^{N} p_{IQ_i}\left(R_{IQ_i} + R_{Q_i I} + R_I\right) + \frac{p_{IR_k}}{a_I}\left(R_{IR_k} + R_{R_k A_k}\right).
\end{aligned}
$$

Thus

$$
\begin{aligned}
R_I &= \frac{1}{1 - \sum_{i=1}^{N} p_{IQ_i}}\left[\sum_{i=1}^{N} p_{IQ_i}\left(R_{IQ_i} + R_{Q_i I}\right) + \frac{p_{IR_k}}{a_I}\left(R_{IR_k} + R_{R_k A_k}\right)\right] \\
&= \frac{1}{J(\infty)}\sum_{i=1}^{N} I_i(\infty)\left(R_{IQ_i} + R_{Q_i I}\right) + R_{IR_k} + R_{R_k A_k}
\end{aligned}
\qquad \textbf{(2)}
$$

since

$$
\frac{p_{IR_k}}{a_I} = \frac{J_j(\infty)}{J_j(\infty)\big/J(\infty)} = J(\infty) \text{ and } 1 - \sum_{i=1}^{N} p_{IQ_i} = 1 - I(\infty) = J(\infty).
$$

The expected reward gained by the firing of transition Q_i given this transition occurs is given by

$$
R_{IQ_i} = \int_0^\infty r_{IQ_i}(t)\, d\left[\frac{I_i(t)}{I_i(\infty)}\right] = \frac{1}{I_i(\infty)}\int_0^\infty r_{IQ_i}(t)\, dI_i(t)\left(= \frac{1}{I_i(\infty)}\int_0^\infty \frac{r_{IQ_i}(t)\,\Omega(t)}{1 - E_{qi}(t)}\, dE_{qi}(t)\right). \quad \textbf{(3)}
$$

Thus R_{IQ_i} is the expected reward gained in state I before a transition is made to state Q_i. Similarly

$$
R_{IR_j} = \int_0^\infty r_{IR_j}(t)\, d\left[\frac{J_j(t)}{J_j(\infty)}\right] = \frac{1}{J_j(\infty)}\int_0^\infty r_{IR_j}(t)\, dJ_j(t)\left(= \frac{1}{J_j(\infty)}\int_0^\infty \frac{r_{IR_j}(t)\,\Omega(t)}{1 - E_{rj}(t)}\, dE_{rj}(t)\right)
$$

is the expected reward for going from state I to state R_j.

Since there is only one possible transition out of states Q_i and R_k, the expected rewards from these states are simply given by

$$
R_{Q_i I} = \int_0^\infty r_{Q_i I}(t)\, dH_{qi}(t) \text{ and} \qquad \textbf{(4)}
$$

$$
R_{R_k A_k} = \int_0^\infty r_{R_k A_k}(t)\, dH_{rk}(t). \qquad \textbf{(5)}
$$

Substituting equations (3) to (5) into (2) gives the required result. ■

To get the more general result that applies to the decision process building block (as shown in Figure 3), the meaning of equation 4 must be examined in greater detail. In a broader sense this term gives the expected reward gained in the circuit. Thus to expand this result from the simple

decision process building block is simply a matter of determining the expected reward of the more complex circuit.

Consider the PN shown in Figure 7. This is a representation of the T-invariants shown in Figure 3. The corresponding state space is given in Figure 8. Clearly, the expected reward for such a state space is the sum of the expected rewards of the individual state transitions, which gives the following lemma.

Lemma 2

The mean first-passage reward for state A_k of a STPN decision process building block is given by

$$R_I = \frac{1}{J(\infty)} \sum_{i=1}^{N} \left(\int_0^\infty r_{IQ_i}(t)\, dI_i(t) + I_i(\infty)\, R_{Q_i} \right) + \frac{1}{J_k(\infty)} \int_0^\infty r_{IR_k}(t)\, dJ_k(t) + \int_0^\infty r_{R_k A_k}(t)\, dH_{rk}(t) \quad , \tag{6}$$

where $r_{Iw}(t)$ is the reward gained by spending time t in I before a transition w ($\in \{Q_1, \ldots, Q_N\} \cup \{R_k\}$) fires, R_{Q_i} is the expected reward gained in circuit Q_i and $r_{R_k A_k}(t)$ is the reward gained by the token spending a holding time of t in place A_k after the firing of transition R_k.

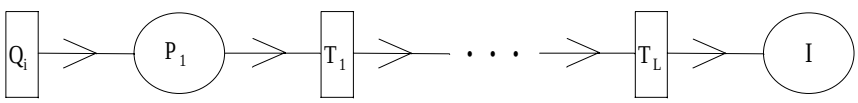


Figure 7: Extended T-invariant as stand alone PN

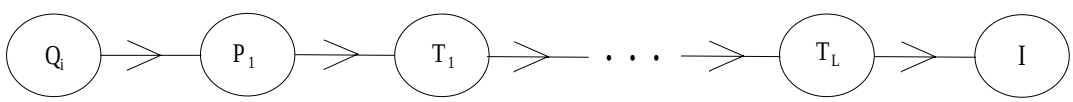


Figure 8: State space of PN in Figure 7

Corollary 1

The mean first-passage time for state A_k of a STPN decision process building block is given by

$$T_I = \frac{1}{J(\infty)} \sum_{i=1}^{N} \left(\int_0^\infty t\, dI_i(t) + I_i(\infty)\, \tau_{Q_i} \right) + \frac{1}{J_k(\infty)} \int_0^\infty t\, dJ_k(t) + \tau_{R_k} \, ,$$

where τ_{Q_i} is the mean time spent in the circuit Q_i and τ_{R_k} is the expected holding time of transition R_k.

Proof:

If the reward of interest is the time taken, then $r_{ij}(t) := t$ for all i and j. Substituting this into equation (6) gives

$$T_I = \frac{1}{J(\infty)}\sum_{i=1}^{N}\left(\int_0^\infty t dI_i(t) + I_i(\infty)R_{Q_i}\right) + \frac{1}{J_k(\infty)}\int_0^\infty t dJ_k(t) + \int_0^\infty t dH_{rk}(t)$$

$$= \frac{1}{J(\infty)}\sum_{i=1}^{N}\left(\int_0^\infty t dI_i(t) + I_i(\infty)\tau_{Q_i}\right) + \frac{1}{J_k(\infty)}\int_0^\infty t dJ_k(t) + \tau_{R_k},$$

since the reward gained is given by the time taken $R_{Q_i} = \tau_{Q_i}$. ■

Consider the case when the transitions have exponentially-distributed enabling durations. Let the firing rate of the transition Q_i, i = 1 to N, be λ_i and the firing rate to the transition R_j, j = 1 to K be μ_j. This means that

$$E_{qi}(t) = 1 - e^{-\lambda_i t} \qquad \textbf{(7)}$$

and

$$E_{rj}(t) = 1 - e^{-\mu_j t}. \qquad \textbf{(8)}$$

Also define $\Lambda := \sum_{i=1}^{N}\lambda_i$ and $M := \sum_{j=1}^{K}\mu_j$.

Equations (7) and (8) give

$$\Omega(t) = e^{-(\Lambda+M)t},\ I_i(t) = \frac{\lambda_i}{\Lambda+M}\left(1 - e^{-(\Lambda+M)t}\right),\ I_i(\infty) = \frac{\lambda_i}{\Lambda+M},$$

$$J_j(t) = \frac{\mu_j}{\Lambda+M}\left(1 - e^{-(\Lambda+M)t}\right),\ J_j(\infty) = \frac{\mu_j}{\Lambda+M} \text{ and } J(\infty) = \frac{M}{\Lambda+M}.$$

Corollary 2

The absorption probability of state k of a decision process building block with exponentially distributed enabling durations is given by

$$P(j) = \frac{\mu_k}{M}.$$

Proof: This follows by substituting the above stated values of $J_k(\infty)$ and $J(\infty)$ into (1).■

Corollary 3

The mean conditional absorption time in state k of a decision process building block with exponentially distributed enabling durations is given by

$$T_I = \frac{1}{M}\left[1 + \sum_{i=1}^{N}\tau_i \qquad \lambda_i\right] + \tau_k.$$

Proof: We have

$$\int_0^\infty t\, dI_i(t) = \frac{\lambda_i}{(\Lambda+M)^2} \text{ and } \int_0^\infty t dJ_j(t) = \frac{\mu_j}{(\Lambda+M)^2}.$$

Thus

$$T_I = \frac{\Lambda + M}{M}\left(\sum_{i=1}^{N} \frac{\lambda_i}{(\Lambda + M)^2} + \sum_{i=1}^{N} \frac{\lambda_i}{\Lambda + M}\kappa_i \right) + \frac{\dfrac{\mu_i}{(\Lambda + M)^2}}{\dfrac{\mu_i}{\Lambda + M}} + \tau_k$$

$$= \frac{1}{M}\left[\frac{\Lambda}{\Lambda + M} + \sum_{i=1}^{N} \lambda_i \kappa_i \right] + \frac{1}{\Lambda + M} + \tau_k \quad ,$$

which on simplification gives the desired result. ■

Note that, as before, this result is independent of the distribution function of the second transition in the circuit. So the enabling duration distributions of the transitions in the circuits do not have to be exponential.