A Matrix Transition Oriented Net for Modeling Distributed Complex Computer and Communication Systems

Anthony Spiteri Staines,
Dept. of Computer Information Systems
University of Malta, Msida
MALTA
tony.spiteri-staines@um.edu.mt

Filippo Neri
Dept. of Computer Information Systems
University of Malta, Msida
MALTA
filippo.neri@um.edu.mt

Abstract: - This work introduces a new type of a Petri net. This is called a Matrix Transition Net. The Matrix Transition net deals with the modeling of modern distributed complex computer and communication systems. The major features are that for the inputs and outputs of this net, matrices are used instead of places as in standard place transition Petri nets. This increases the modeling power and expressivity of the net. The structures presented are capable of representing complex systems and their dynamic modeling concepts. The main features of a Matrix Transition net are explained using several examples and definitions. A toy case study of a small grid sequential system illustrates the potential use of this modeling notation. Some results are discussed.

Key-Words: - Complex Systems, Matrices, Matrix Transition Net, Modeling, Networks, Petri nets.

1 Introduction

Petri nets are powerful semi formalisms for modeling diverse types of systems, dynamic behaviour and scenarios. One of the main features of Petri nets is the visual identity that they use to represent systems. Visually Petri Nets have many similar properties to other block notations [1]-[2],[4],[11]-[12], [16]-[20].

In the field of computing, Petri nets have served to represent the dynamic behaviour of discrete systems, ranging from simple systems to complex intricate network behaviour [4]-[5],[8],[11]-[12],[16]-[20].

The normal concepts behind Petri nets are very useful for structural and dynamic representation of systems [1]-[2],[7],[11],[14]-[20]. In essence a Petri net is just a set of inputs connected to a transition that in turn connects to a set of outputs and behaves predictably.

One of the limiting factors of ordinary Petri nets is the use of simple token and types. In reality, modern systems that exist in the real world are not so straightforward.

Modern evolution in computing and information technology requires more analysis of expressivity and complexity than ever [15]-[16]. Even systems that, at a superficial level, might look quite straightforward can present a high level of complexity hidden beneath lower levels of implementation. It is possible to have systems, with multiple inputs and outputs that connect to different levels. In the real world, different entities, objects or events might interact with systems they are unrelated to. This could be called a system's real world interface or interaction space.

One possible way to solve the issues mentioned is by using higher order nets, where a token is not just simply a token but has dual identity. Higher order nets approach or formulate a solution to this problem [8]. I.e. it has a token value and an entity value as is done in coloured Petri nets. Higher order nets [8] are still subject to certain limitations and are not always simple to comprehend and use. Observing implemented systems it is observed that they are subject to limitations and it is not always possible to represent certain types of complexities like partial tokens or partial firing, etc.

As a general idea, when modeling systems, it would be required to use the concepts involved in Petri nets whilst simultaneously using improved and more detailed ways of representation and modeling, hence properly representing the intricate complexities of modern systems in a suitable way.

Modern systems are based on concepts of cloud computing, grid computing, parallel computing, distributed computing and all varieties of different computing architectures. To model these systems, new expressive methods are always sought [4].

This paper serves to briefly introduce the challenging ideas and concepts of matrix-transition oriented nets, without delving excessively into details and their formal implementation. The main uses of these nets and their applicability to modern modeling situations should easily be inferred.

2 Motivation

The main motivation for these new models, is to create new more expressive and innovative ways for modeling modern systems that can exist in different configurations and complexities.

At the same time, for this type of modeling, the simple concepts behind normal Petri nets are used and implemented, so that this idea can be developed and researched as required.

One fundamental property of Petri nets is that these are visual and can be shown or depicted using inflows and outflows that connect to a transition. I.e. a transition connects to input and output places via directed arcs. Hence, a normal Petri net is a bipartite digraph. The inputs and outputs of Petri nets can be subject to certain rules and conditions. At a basic level constraints can be placed on input and output arcs and even place capacity. This idea is evidenced when the input and output flow matrices for a classic Petri net are constructed [1]-[2].

The result of replacing or combining Petri net places with matrices, is that it is more expressive and compact than ordinary Petri nets [6],[13]. I.e. a whole series of separate nets can be combined in a given matrix, given certain fundamental rules and limitations that might hold. If matrices are used for input and output places, more detail and information can be contained in the net model.

Matrices can be considered to be forms of extraordinary algebra, i.e. for sophisticated applications, there is a move from a single dimension to multidimensional representation. Because of this their, application to Petri nets, using matrices instead of normal places can result in having extremely powerful representation and modeling capabilities [9],[10].

Normally when firing a transition in a normal Petri net, a token or a set of tokens are moved out of a place or a number of places. Using a matrix, instead of places with values, increases the dimensional representation of what is contained in that place. Hence, in a single transition firing, many processes or events and entities can be contained. Thus, the need to have multiple transition firings can become unnecessary. Many features of ordinary Petri nets are easy to implement in the matrix transition nets.

Matrix related operations can be carried out using normal matrix algebra.

3 Problem Formulation

The problem of this work, is that of increasing the effectiveness of Petri net structures for system representational purposes. Petri net structures have several limitations, based on the fact that the structures are normally designed for representing low level behavior in discrete event systems. Ordinary Petri nets have been used to model complex systems, but unfortunately many features and modifications have to be added to do this. Ordinary tokens are simple and not complex, so they cannot properly represent complex configurations. The strong point of Petri net modeling lies precisely on the visual or graphical representation of these formalisms.

Using ordinary Petri nets, places are limited to representing a single dimension [1]-[2],[7]. This issue can be solved using colored Petri nets [8]. However colored Petri net tokens have a dual identity, i.e. the token value and a special identity value.

Colored Petri nets are useful for modeling complex systems and can give a compacted representation. The idea of a matrix transition net is to replace Petri net places directly with matrices [9]-[10], but there is no dual identity in this case as in colored Petri nets. The only identity is that of the matrix.

This concept used is similar to algebraic and some other forms of higher order nets. The individual matrix element can be considered to be similar to tokens. Petri nets are suitable for representing both the structure and the operational behavior of systems. This same idea is propagated in the matrix-transition net.

The use of the incidence matrix for an ordinary Petri net [1]-[2] already shows the importance of matrices used for describing the relationships inside the net. Place and transition invariants of simple nets can also be derived from matrices.

Here instead of using matrices just to describe the relationships, ordinary places are replaced by matrices [9]-[10]. Thus definitely more information can be contained in this structure. At the same time the rules for firing and post firing become more complex. This implies that some form of input and output control is necessary.

All these concept reflects what is happening in a real system, where it is possible to have multiple states, entities, objects or things that share some form of temporal relationship and different forms of input and output control.

Petri net like structures can offer improved visual and mathematical ways for representing and modeling modern systems, whilst at the same time maintaining simplicity for showing how this can be successfully done. I.e. the methods used should be repeatable, robust and analytical. Precisely for this reason, the matrix transition net has to behave consistently.

4 Problem Solution

This part explains the practical implementation and properties for matrix-transition nets. The matrix transition net should be comprehended to be similar in operation to ordinary Petri nets. It is quite simple to understand its operations and concepts. The solution and the implementation can be modified according to a particular problem scope. The matrix transition net can be defined formally or semi formally as needed. The proposed solution is based on the diagrams in fig.1. The Petri net and the matrix transition net have just been shown together for comparison's sake only. There is no special relationship between the two. For the descriptions and explanations below the matrix-transition net in fig.1 ii) is indicated. The Petri net in fig. 1 i) shows a side by side comparison of the differences between an ordinary place transition net and the matrix transition net in fig. 1 ii).

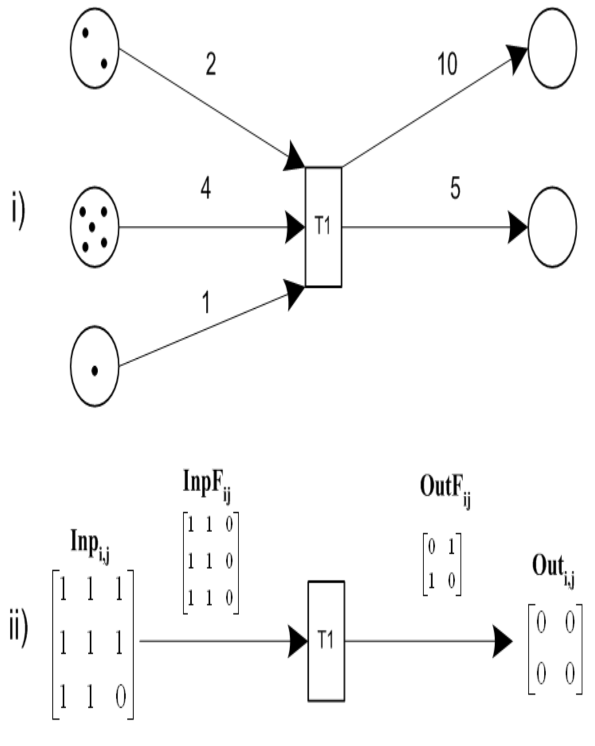

Fig.1. i) Ordinary Petri net, ii) Matrix-Transition net

4.1 Basic considerations for matrix transition nets

Some assumptions and properties of matrix-transition nets and their Petri net counterparts are imperative for proper understanding of the solution.

In simple terms, the matrix-transition net that is being proposed in this work is a modification to normal place transition Petri nets [1]-[2],[7].

The basic idea is that instead of places which contain tokens, matrices are used. Hence places are no longer considered but they are replaced with matrices that contain elements. This presents a new problem because some matching criteria is required for transition enabling and firing. For this purpose the idea of arc inscriptions or arc weights from ordinary Petri nets is elaborated. Instead of arc weights in the matrix-transition net, an i) input function matrix and an ii) output function matrix are considered and linked to the input and output arcs respectively.

The behaviour of the matrix transition net is similar to that of the ordinary Petri net. I.e. in a Petri net for transition enabling and firing the input conditions must be satisfied. For proper transition firing in the matrix transition net, the input matrix must match the criteria of the input function matrix. Similarly, the output of a Petri net transition depends on the output arcs and their weights. In the matrix transition net the output is determined by an output function matrix.

In the matrix-transition net the input and output functions do not need to match each other at all. Consequently the input and output matrices of a transition do not necessarily need to have the same order. This is observable and synonymous with normal Petri nets, where the amount of input places does not necessary match the amount of output places. This is also analogous with Petri nets where the input and output arc inscriptions or weight values can be different.

The input and output matrices can be square or non-square, etc. depending on what is being modeled.

4.2 Global transitions

Global transitions represent how things really take place in the real world. There is no need to formally prove or reject any hypothesis. E.g. in the real world, in any time period ΔT many connected or unconnected events might be taking place. The global transition is the actual passage of time, the axiomatic transitions are the events or activities that take place in ΔT .

Any event that happens in time (t_1,t_2) is actually part of a global transition ΔT . The idea of using matrices instead of places in the Transition net concurs with the idea that events taking place in the real world have a separate identity and might be related or unrelated. I.e. tokens in a Petri net if considered from a real world view, have different relationships and identities. The identity of tokens or events, do not need to match the other elements. But obviously there could be a direct or indirect relationship as dictated by the law of causality.

As an analogy, in a computer system simple clock cycle, many activities take place but they might share or not share anything in common. Resources that are related or unrelated are still consumed or used.

A global transition T is composed of a set of smaller axiomatic transitions t. There could be no special firing order, but firing could occur in some sequence in ΔT . This can give the general impression that firing occurs in parallel or in some sequence.

4.3 Matrices

Matrices [9],[10] are used to represent the input and output and the input and output functions respectively. The input matrix connects to a transition via an input arc and the output matrix connects to the output of a transition via an output arc. The matrices are of the general form M_{ij}. Where the addressing of the values of the matrix are determined by the values i,j. i represents the row number and j represents the column.

4.4 Input matrix

The input matrix is a matrix that contains all the input values to a particular transition. This analogy is similar to a Petri net, input place. The input matrix is connected via an arc to a transition. All the elements of the input matrix are non-negative values. This matrix can be labelled Inp_{ij}.

4.5 Output matrix

The output matrix is a matrix that contains all the output values from a particular transition. The output matrix is connected via an arc from a transition. All the elements of the output matrix are non-negative values. This matrix can be labelled Out_{ij}.

4.6 Input function matrix

The input function matrix determines the firing conditions for the transitions. As an analogy, this is similar to the values of the input arcs in Petri nets. I.e. if in the input matrix there are insufficient values then the transition cannot take place. The input matrix must be of the same order as the input function matrix connecting it. E.g. if the input function matrix is given as $InpF_{ij}$, where i,j are the number of rows and columns of the input function matrix. $InpF_{ij}$ and Inp_{ij} are of the same order and are congruent.

4.7 Output function matrix

The output function matrix determines the output of the transition. This is similar to a Petri net transition output where the output values on the arc explain the number of output tokens from a particular transition firing to be placed into the output place.

The output function matrix must be of the same order as the output function matrix connected to it. E.g. if the output function matrix is given as $OutF_{ij}$, where i,j are the number of rows and columns of the output function matrix. $OutF_{ij}$ and Outij are of the same order and are congruent.

4.8 Complete vs partial transition firing

For complete transition firing the input matrix must satisfy the conditions of the input function matrix. It is not sufficient for values to be present in the input places, but these must match the input function value. Complete firing implies full throughput or that all the conditions of the input function matrix are satisfied.

Given the concept of global transitions, it is possible to have partial transition firing. This implies that the transition is incomplete and only part of the output is modified. This property can be excluded from the net. For partial or incomplete firing, some values in the input matrix must match some values of the input function matrix. However more explanations, rules or definitions might need to be added to properly implement and define partial firing and its results.

In this work the focus is only on complete transition firing.

4.9 Transition firing

In Petri nets transition firing implies the removal of tokens from the input places and the placement of tokens in the output places. The relevant arc weights or marking functions are considered. Similarly in the matrix-transition net, values are removed or subtracted from the input matrix and values are added to the output matrix. The general idea is given below.

4.10 Removal of values from input matrix

This is a simple subtraction of the input matrix from the input function matrix. The result should be the updated input matrix after transition firing, i.e. this is the next marking of that matrix. For the example given this implies $Inp_{ij} = Inp_{ij} - InpF_{ij}$

The resultant values for all the elements in the matrix must be 0 or >0. I.e. $\forall_{i,j} \in M_{i,j}; i \geq 0, j \geq 0$.

4.11 Addition of values from input matrix

Placement of tokens is a simple matrix addition operation. The output function matrix is added to the output matrix, giving the updated output matrix. I.e. the new values of the output matrix after transition firing. For the example given this implies $Out_{ij} = Out_{ij} + OutF_{ij}$.

4.12 Defining a complete matrix transition net

The definition of the matrix-transition net can be elaborated from that of ordinary Petri nets.

A basic matrix-transition net is defined as bi-partite digraph, having two types of vertices. This is represented as a five tuple set, MT = (M,F,T,FM). M is a finite set of Matrices M= $\{m_1,m_2,m_3,\ldots,m_n\}$ and T is a finite non empty net of transitions T= $\{t_1,t_2,t_3,\ldots,t_n\}$. F is a finite non empty set of flows from a matrix to a transition and vice-versa, given as $F \subseteq \{(MxT) \cup (TxM)$. Normally (MxT) represents the input arcs and (TxM) represent the output arcs. FM is a finite set of function matrices FM = $(fm_1,fm_2,fm_3,\ldots,fm_n)$. InpF is the input function matrix for a given input flow f. Similarly OutF is the output function matrix for a given output flow f.

The equations for the input and output function matrices are given in (1) and (2)

$$InpF : F \rightarrow \{ fm_1, fm_2, fm_3,..., fm_n\} \quad (1)$$

$$OutF : F \rightarrow \{ fm_1, fm_2, fm_3,..., fm_n\} \quad (2)$$

Matrices and transitions are disjoint i.e. $M \cup T = \phi$ and $T \cup M = \phi$. Nodes are not isolated. It is possible for the matrix-transition net to have an initial marking which can be composite.

4.13 Constructing complex matrix transition nets

The concepts behind Petri nets, which have been applied to matrix-transition nets, have been partially described in the examples section. These serve to construct more complex structures that can virtually be used to describe many other types of modern systems. E.g. the idea of choice, forking, parallelism, mutual exclusion etc. can be used for enhancing the models.

5 Matrix transition net examples

5.1 Simple transition firing

Fig. 2 shows simple transition firing and the state of the net before and after firing. The changes in the input and output matrix are clearly visible. It is also noted that the input and output matrices are of a different order.

This is synonymous with place transition Petri nets. Also it is clear that the input function matrix and the input matrix have to be congruent.

It is noted that transition firing implies the subtraction of values from the input matrix and the adding of values to the output matrix.

It is possible to have multiple inputs and outputs connected to a transition, as is normal with Petri nets.

5.2 Insufficient conditions for firing

In fig.3 insufficient conditions for transition firing are shown. This implies that there are insufficient resources or values available in the input matrix or matrices. This is shown in fig. 3. I.e. there are insufficient inputs. This condition is easily identified as the values of the input matrix are less than the values of the input function matrix. In this case the transition cannot fire and values cannot be removed from the input matrix. Also values cannot be added to the output matrix. I.e. transition firing is disabled.

5.3 Multi input matrix transition net

The multiple input matrix-transition net is just a normal net with multiple inputs. Obviously it is possible to have multiple outputs in a similar fashion or both multiple inputs and outputs simultaneously. The concept of multiple inputs is shown in fig.4.

5.4 Choice or conflict

Choice in the matrix transition net is similar to choice in Petri nets. Sometimes in Petri nets choice is also associated with conflict. This implies that different execution paths are possible i.e. indecision.

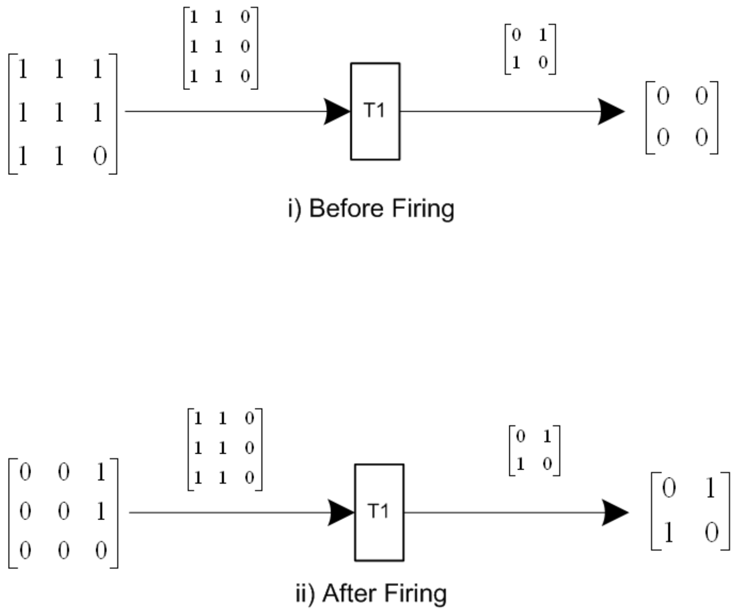

Fig.2. Normal Transition Firing

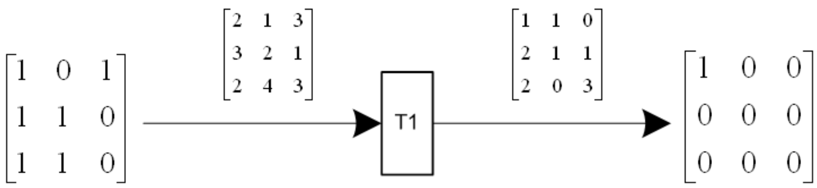

Fig.3. Insufficient Conditions for Transition Firing

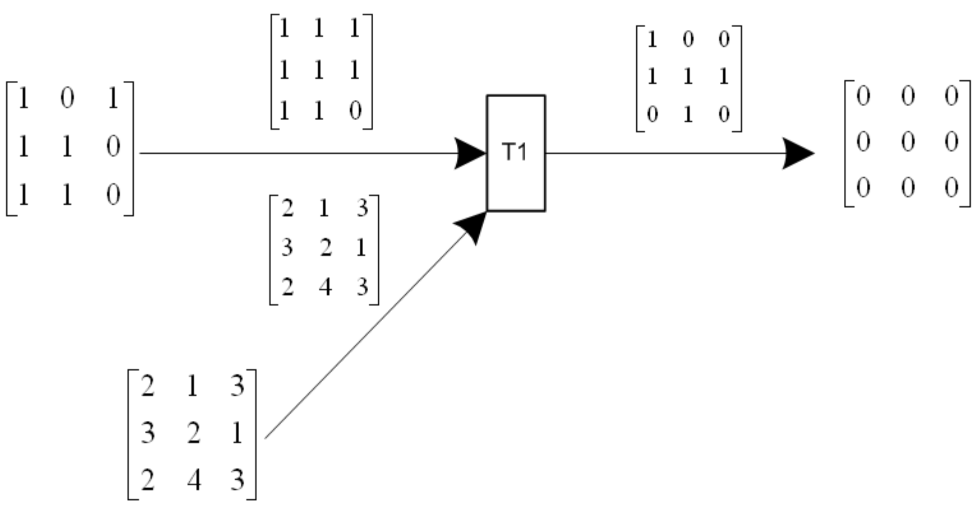

Fig.4. A Multiple Input Matrix Transition Net

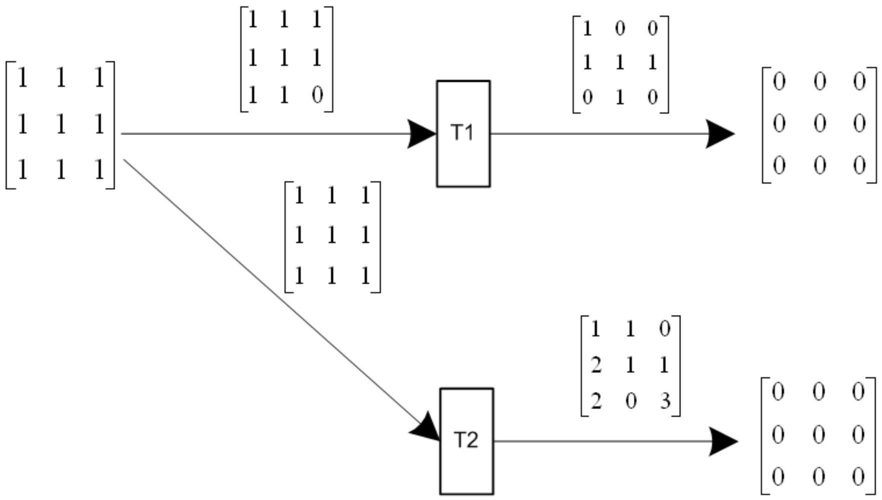

Fig.5. Matrix Transition Net with Choice

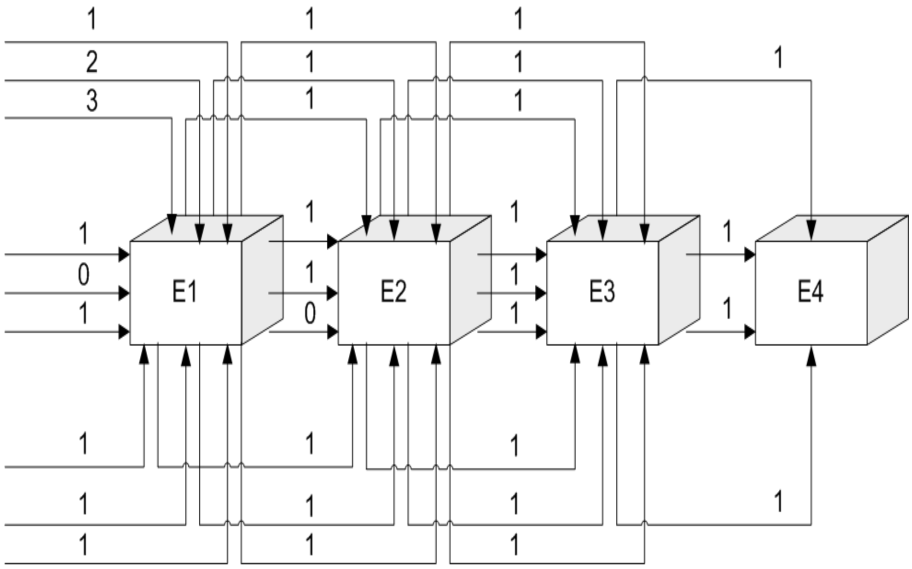

Fig.6. A Grid System with Four Processing Elements in Series

Choice extends the modeling power. The idea of choice is represented in fig. 5, where either T1 or T2 can fire but not both. I.e. the firing of one automatically excludes the firing of the other.

6 A Toy Case Study Example

To illustrate the practical use of the matrix-transition net, a simple toy case study is presented. A set of processors or elements are connected in series in a grid system or for a distributed processing solution. This is illustrated in fig. 6. Each element or processing element has a number of multiple inputs and outputs.

The last element has a different number. For each processing element to work, particular values need to be present at each input to the element. I.e. each element requires different input elements to operate. The elements also have outputs. Outputs from one processing element to another can be different. The input values required by the processing element are represented using the input function matrices and the output values can be represented sing the output function matrices of the matrix-transition net. The processing elements are

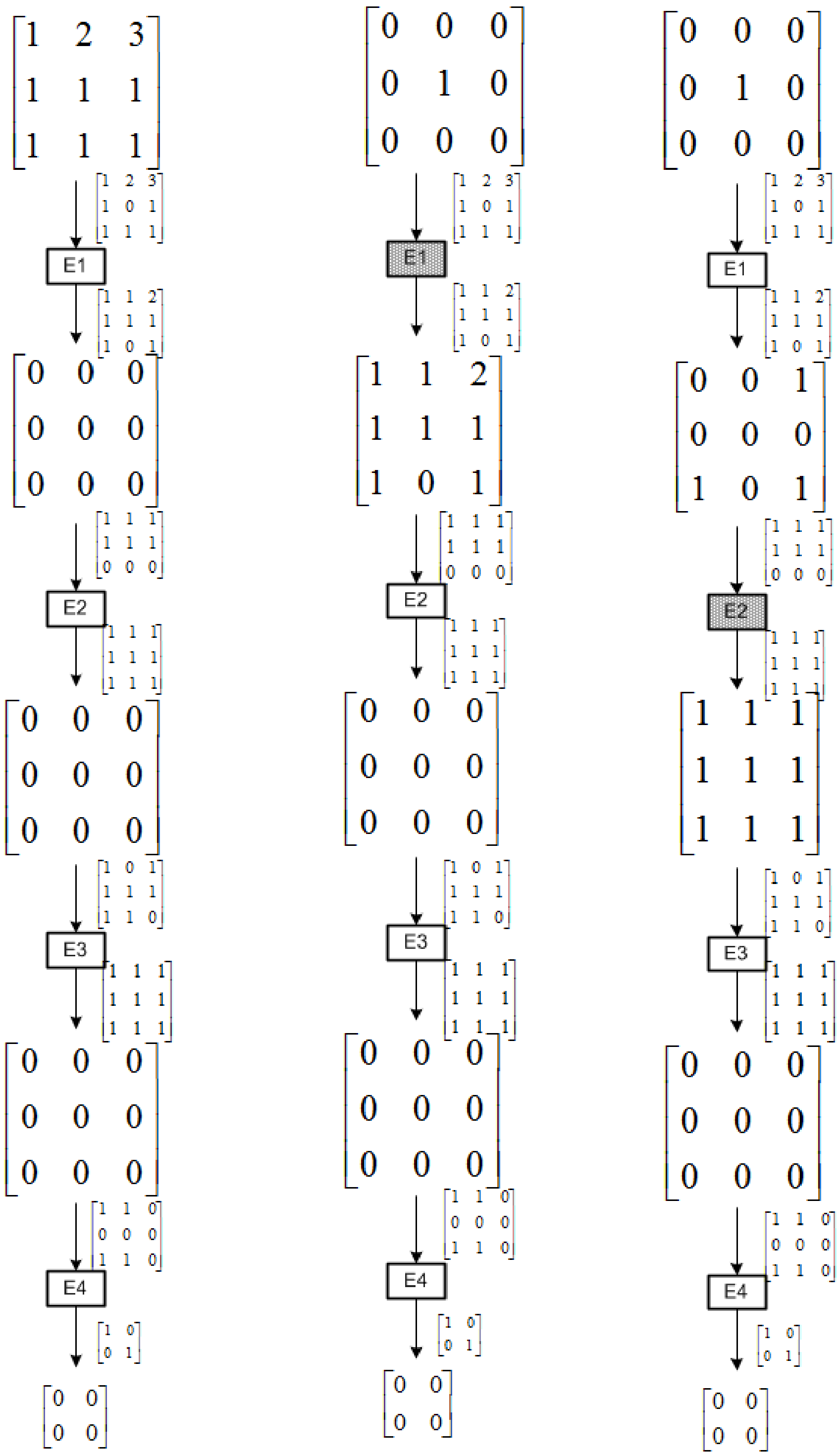

Fig.7. Grid System Matrix-Transition Net Part 1

represented as transitions. A single input arc and a single output arc suffices to represent the element connections for this type of structure.

The resultant matrix transition net for the serial grid processing element nodes is shown in fig. 7 andfig.8. The initial marking of the net is $\begin{bmatrix} 1 & 2 & 3 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$ at the input matrix for E1. Fig. 7 shows the initial state of the matrix-transition net and the actual firing of each transition or processing element in sequence from E1..E4. The results and changes in the input matrices and output matrices for each transition firing are given incrementally. This illustrates how the matrix-transition net can be used to depict the functioning of a complete complex system.

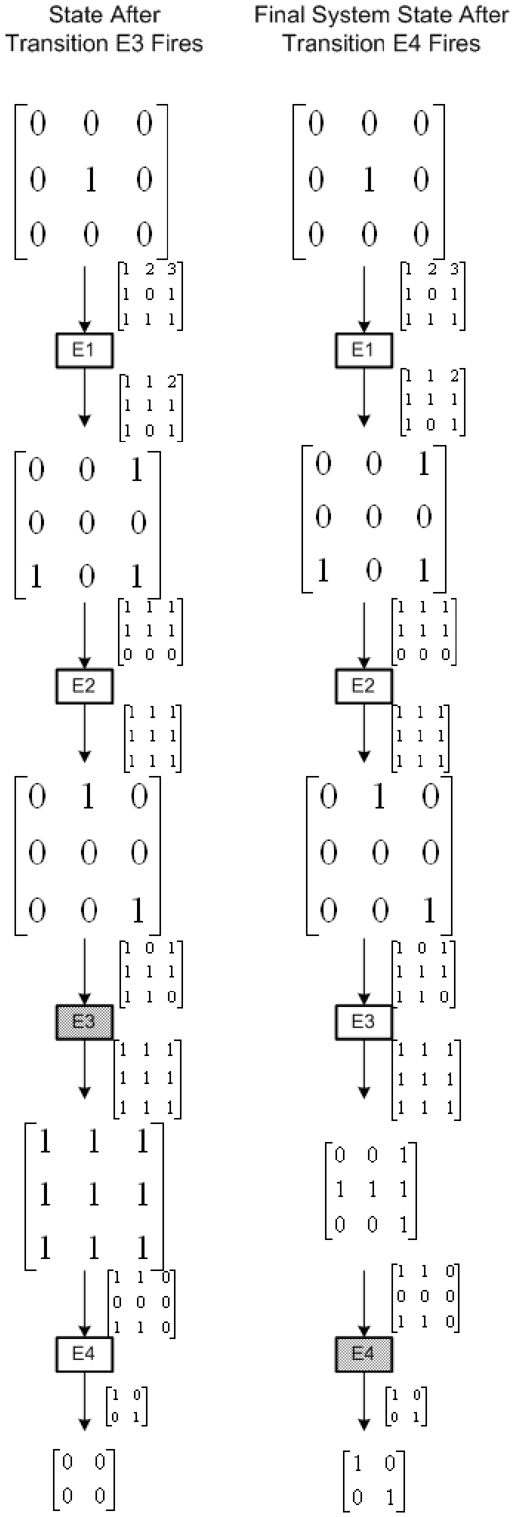

Fig.8. Grid System Matrix-Transition Net Part 2

The case study presents a useful analogy. This is used in grid nodes, network processing, modern communication systems etc.

7 Results and Findings

The model developed, in the case study, is fully executable and functional. The model has a i) limited state space, and ii) limited number of possible markings. Four global composite markings are possible. These are $M_1 \rightarrow M_2 \rightarrow M_3 \rightarrow M_4$. The states for this example are non-reversible. I.e once a state has been reached it is impossible to go back to a previous state.

The matrix–transition net is more expressive than conventional place transition nets. The model is also more compact than a Petri net and captures a greater amount of information. It is more suited to complex systems. It can adequately represent the reactivity of systems. A marking graph or a symbolic reachability graph can be constructed. This can be called an execution path or execution marking.
All the basic constructs that are used in Petri nets like i) sequential , ii) parallel , iii) choice, iv) conflict, v) confusion, vi) start & stop, vii) non-determinism can be included in the matrix-transition net described.

8 Discussion

Petri nets are convenient graphical formalisms for representing systems. Their importance can be seen from their modern use in FMCs and the UML, supporting dynamic behavior [4],[11]. The sound mathematical foundation of Petri net structures is useful, because sophisticated tools, formulae and methods can be used for their analysis [1]. Petri nets are expressible using other formalisms or equations. The behavior can be properly analyzed. The concepts behind Petri nets are applicable to the matrix transition net. The matrix transition net's behavior has been lifted off from that of Petri nets. Many variants of Petri nets exist. E.g. high level nets, algebraic nets, coloured nets, predicate transition nets, object oriented nets, etc [1]-[2], [7]-[8].

Ordinary place transition Petri nets use places for input and output. The places contain tokens that represent resources. This type of place can be seen to represent a one dimensional view only. Petri nets have a rigid fixed structure. In fact the structures cannot normally change dynamically [1]-[2]. Several propositions were made to extend Petri nets by reconfiguration, but sometimes the final result is not a Petri net structure at all. Normal Petri net transitions do not allow for partial firing [2], [7]-[8].

The matrix transition net presented in this work offers several advantages over normal place transition Petri nets. These are i) a combination of multiple places and token values into a single matrix, ii) compactness, iii) extended modeling capabilities and iv) reduction to normal Petri net if required. The advantages imply that a single matrix can represent several places and inputs. Matrices can contain more than a single dimensional view [9]-[10]. The net is expressed in a more compact form rather than is done in classic Petri nets. This implies that the structures also have better visual impact. This structure offers the possibility of extended modeling. This implies that complex systems could be represented using these structures as compared with Petri nets which are not so effective for representing complexity.

The matrix transition net can be decomposed into a Petri net although not all parts might be expressible using Petri nets [6],[14]. By performing some modifications the matrix transition net can be virtually modifiable at run time. I.e. it is possible to change certain matrix values whilst the net is being executed. This could be done by introducing a secondary net. Many of the analysis methods used for place transition Petri nets are applicable to the matrix transition net. [2],[17]-[20]. I.e. it could be possible to test for deadlock, invariants, markings and reachability. These important criteria are useful for analysis and are advantages that are offered by the Petri net formalisms [1],[2].

With some simple modifications the matrix transition net can be used for partial firing. The possibility of partial firing has not been explored in this work. This idea can be formally defined for the matrix transition net. Partial firing can open up a lot of new possibilities which closely resemble system behavior in the real world. This is not possible with normal Petri nets.

In simple terms what has been achieved is the construction of a more advanced form of Petri net that has extended capabilities for system representation, precisely because of the use of matrices instead of input and output places as in normal Petri nets [1]-[2].

The matrix transition net is useful to model classes of problems, where complex communication is realized as taking place between real world system entities or elements. However this would have to be restricted to certain levels of detail. On the other hand Petri nets are highly detailed and more suited to low level modeling. Although this approach does offer advantages, new problems can arise. I.e. if the network is complicated there might not be a correct solution. The solution presented is applicable to model certain types of systems that are composed of elements connected in series or in parallel. The use of large matrices can complicate representation. The concept of grouping places and tokens into a matrix might not be feasible to describe certain types of system behavior. On the other hand the use of matrices opens up the possibility for a lot of new exploration based on matrix theory. The matrix transition net can be represented simply using matrices [9]-[10].

9 Conclusion

A simple new comprehensible solution for representing the complexities of modern systems has been presented using matrix transition nets. The solution is useful for a wide range of static and dynamic systems from an architectural and low-level viewpoint. The matrix transition net would be useful for modeling communication systems or computer systems from an abstract level. The examples given are trivial and the case study is a simple one. The matrix transition net given can be used to generate executable models useful for simulation.

The original ideas presented, can serve for detailed work in many other directions in the future.

For future work the matrix transition net presented can be improved and modified to include more details and labeling. The definitions can be formalized and more structures can be added or combined regarding the modeling scenario that is being tackled.

References:

[1] M. Zhou, K. Venkatesh, *Modeling, Simulation, and Control of Flexible Manufacturing Systems A Petri Net Approach*, Vol. 6 Series in Intelligent Control and Automation: World Scientific, 1999, pp. 1-100.

[2] T. Murata, *Petri nets: Properties, Analysis and Applications*, Proc. of the IEEE, vol. 77, issue 4,1989, pp. 541-580.

[3] H. Bal, *Programming Distributed Systems*, Pretence Hall, 1990.

[4] A. Knopfel, B.Grone, P. Tabeling, *Fundamental Modeling Concepts*, Wiley, 2005.

[5] G.S. Fishman, *Discrete Event Simulation*, Springer, 2001.

[6] R.H. Sloan, U. Buy, *Reduction Rules for Time Petri Nets*, Acta Informatica, Vol. 33, Issue 5, Springer-Verlag, 1996, pp. 687-706.

[7] C.A. Petri, *Introduction to General Net Theory and Applications*, LNCS Springer Verlag, vol. 84, 1990, pp. 1-19.

[8] K. Jensen, G. Rozenberg, *High-level Petri Nets : Theory and Applications*, 1st ed., Springer-Verlag ,1991, ISBN-10: 354054125X ,ISBN-13: 978-3540541257.

[9] K.M. Abadir, J.R. Magnus, *Matrix Algebra*, Cambridge University Press, 2005.

[10] F. Ayres (jr), *Theory and Problems of Matrices*, Schaum's Outline Series, Schaum, 1974.

[11] T. Gehrke, U.Goltz, H. Wehrheim, *The Dynamic Models of UML:Towards a Semantics and its Application in the Development Process*, TR. 11/98, University of Hildesheim, Germany, 1998.

[12] A. Spiteri Staines, *Rewriting Petri Nets as Directed Graphs*, Naun Int. Journal of Computers, vol 5, issue 2, 2011, pp. 289-297.

[13] A. Spiteri Staines, *Some Fundamental Properties of Petri Nets*, IJECCE, Vol. 4, Issue 3, 2013, pp. 1103-1109.

[14] M.B. Dwyer, L.A. Clarke, *A Compact Petri Net Representation and its Implications for Analysis*, IEEE Transactions on Software Engineering, vol. 22, issue 11, 1996, pp. 794 – 811.

[15] F. Neri, *Traffic Packet Based Intrusion Detection: Decision Trees and Genetic Based Learning Evaluation*, WSEAS Transactions on Computers, WSEAS Press (Wisconsin USA), vol. 4, issue 9,2005, pp. 1017-1024.

[16] J. Zhao, X. Yang, J. Jiang, *Implementation and Study of Grid-Connected Control for Distributed Generation System*, WSEAS Transactions on Computers, WSEAS Press (Wisconsin USA) vol. 9, issue 5, 2010,pp. 570-579.

[17] C. Baskocagil, S. Kurtulan, *Generalized State Equation for Petri Nets*, WSEAS Transactions on Computers, WSEAS Press (Wisconsin USA) vol. 10, issue 9, 2011,pp. 295-305.

[18] P. Strbac, M. Tuba, D. Simian, *Hierarchical Model of a Systolic Array for Solving Differential Equations*, WSEAS Transactions on Computers, WSEAS Press (Wisconsin USA) vol. 8, issue 1, 2009,pp. 12-21.

[19] A. Spiteri Staines, *Supporting Requirements Engineering With Different Petri Net Classes*, Naun Int. Journal of Computers, vol 4, issue 4, 2010, pp. 215-222.

[20] A. Spiteri Staines, *A Colored Petri Net for the France-Paris Metro*, Naun Int. Journal of Computers, vol 6, issue 2, 2012, pp. 111-118.