International Journal of Pure and Applied Mathematics
Volume 93 No. 5 2014, 633-646
ISSN: 1311-8080 (printed version); ISSN: 1314-3395 (on-line version)
url: http://www.ijpam.eu
doi: http://dx.doi.org/10.12732/ijpam.v93i5.5

ijpam.eu

REPRESENTING THE PAIRWISE INTERSESSION NETWORK CODING BY HIGH LEVEL PETRI NETS

Odae Al-Aboud[1 §], M.J. Al-Laban[2], A.A. Hanano[3]
[1,2,3]Department of Mathematics
Faculty of Science
Damascus University
Damascus, SYRIA

Abstract: In this paper, we introduce a new approach to the pairwise intersession network coding, which depends on using high level Petri nets to represent the butterfly network, the grail network, all networks which represent an expansion of the butterfly network and all networks which represent an expansion of the grail network, and use this representation to determine the solvability of the pairwise intersession network coding problem.

AMS Subject Classification: 94A99
Key Words: network coding, pairwise intersession network coding, butterfly network, grail network, high level Petri nets

1. Introduction

The idea of network coding is introduced by Ahlswede et al. [1], Follow-up work [2] shows that linear network coding is sufficient for a single multicast session. An algebraic approach to network coding has been developed by Koetter and Medard [3]. Several other related works of intra-session network coding can be found in [4]–[6]. When we generalize the problem such that there is more than

Received: January 26, 2014

[§]Correspondence author

one session and receivers may demand different sets of information, finding the optimal network coding strategy is still an open question. The linear coding is shown to be insufficient for optimal coding in the multi-session case [7]. In their paper [3] Koetter and Medard introduced an algebraic point view on network coding but the solvability decision problem is shown to involve Grobner basis computation, whose complexity may prohibit practical implementations for large problems. In our work [8] we redefine the intersession network coding (INC) as an algebraic structure. The intersession network coding on a communication network is called pairwise intersession network coding if and only if the network contains two source nodes and two sink nodes and each sink node demands the information which is generated at one source node[9, 10]. In this paper we introduce a new approach to the pairwise intersession network coding (PINC), which depends on using high level Petri nets (HLPNs) [11] to represent the butterfly network, the grail network, all networks which represent an expansion of the butterfly network and all networks which represent an expansion of the grail network, and use this representation to determine the solvability of pairwise intersession network problem.

2. Preliminaries

2.1. Pairwise Intersession Network Coding

We assume that the network is given by a directed acyclic graph $G = (V, E)$ where V and E are the set of all nodes and links, respectively [13], links are denoted by round brackets $(v_1, v_2) \in E$ and assumed to be directed, the head and tail of an link $e = (v', v) \in E$ are denoted by $v = head(e)$ and $v' = tail(e)$ [3], for each node $v \in V$ we define $In(v) = \{e \in E; head(e) = v\}$, $Out(v) = \{e' \in E; tail(e') = v\}$,then the sets of source and sink nodes are define as follows: $S = \{v \in V : |In(v)| = 0\}$, $R = \{v \in V : |Out(v)| = 0\}$, let $S = \{s_1, s_2, ...s_w\}$ and $R = \{R_1, R_2, ...R_N\}$.

If $G = (V, E)$ contains exactly two source nodes s_1 , s_2 and exactly two sink nodes R_1 , R_2, where R_1 demands the symbol x_1 which is generated at s_1 and R_2 demands the symbol x_2 which is generated at s_2 , then the network coding problem (NC) over $G = (V, E)$ is called pairwise network coding problem (PINC) [9].

Theorem 1. (see [13]) *Define two sets of graphs, G_b and G_g , as follows:*

- *G_b contains the butterfly as described in Figure(1-a) and all graphs obtained from the butterfly via edge contraction, e.g. , Figure(1-b).*

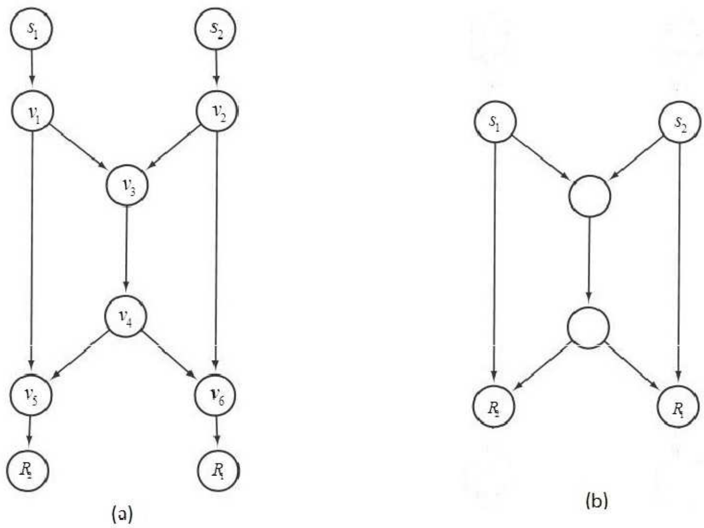

Figure 1

- *G_g contains the grail as described in Figure(2-a) and all graphs obtained from the grail via edge contraction, e.g. , Figure(2-b).*

Suppose there exists a network coding solution to the pairwise intersession network coding problem. Then one of the following two conditions must hold.

1. There exist two EDPs connecting (s_1, R_1) and (s_2, R_2).
2. G contains a subgraph $G^{'} = (V^{'}, E^{'})$ such that (i) $\{s_1, s_2, R_1, R_2\} \subseteq V^{'}$ and (ii) there exists a $G_r \in G_b \cup G_g$ such that $G^{'}$ is a subdivision of G_r.

2.2. Petri Nets

Petri nets are a graphical and mathematical modeling tool applicable to many systems, they are promising tools for describing and studying information processing systems [14].

Definition 1. [8] A net graph is a structure $NG = (P, T; F_{in}, F_{out})$ where:

- P is a finite set of nodes, called Places.
- T is a finite set of nodes, called Transitions, disjoint from P .
- $F_{in} \subset T \times P$ is a set of of directed edges called input arcs.

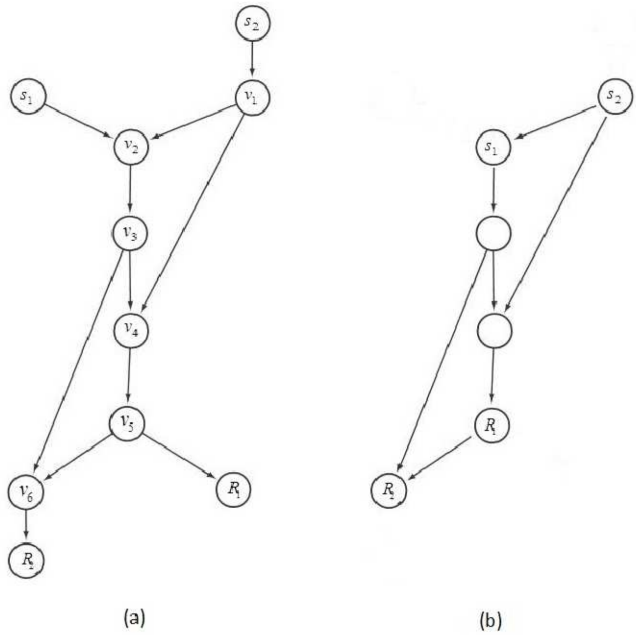

Figure 2

- $F_{out} \subset P \times T$ is a set of of directed edges called output arcs.

Definition 2. [11] A High-level Petri Net (HLPN) comprises:

- A Net Graph.
- Place Types. These are non-empty sets. One type is associated with each place.
- Place Marking. A collection of elements (data items) chosen from the place's type and associated with the place. Repetition of items is allowed. The items associated with places are called tokens.
- Arc Annotations: Arcs are inscribed with expressions which may comprise constants, variables (e.g., x; y) and function images (e.g., f(x)). The variables are typed. The expressions are evaluated by assigning values to each of the variables. When an arc's expression is evaluated, it must result in a collection of items taken from the type of the arc's place. The collection may have repetitions.

- Transition Condition: A Boolean expression (e.g., x < y) inscribing a transition.

3. Representing the Network Graph G by High Level Petri Net

Without loss of generality we assume that the output of each source node is equal to one, and assume that the input of each sink node is equal to one, so if the output of a source node s is more than one, we add a source node $s^{'}$ and an edge from $s^{'}$ to s , and if the input of a sink node R is more than one, we add a sink node $R^{'}$ and an edge from R to $R^{'}$.

3.1. Net Graph

To represent the network graph by a net graph, we partition it into subgraphs through which the same information flows, and represent each subgraph by a place, a transition and an arc from the place to the transition, after that we represent the set of edges which connect nodes from a subgraph to nodes from another subgraph by an arc from the place which represents the first subgraph to the transition which represents the second.

3.2. Place Types

We assume that the finite field F_q is the place type of all places in the high level Petri net.

3.3. Place Marking

If the place p represents the subgraph G_p then the place marking of p is equal to the information symbol which flows through G_p .

3.4. Arc Annotations

- The expression of the arc $(p^{'}, t^{'})$ is equal to the place marking of $p^{'}$.

- The expression of the arc (t, p) is called t 's operation.

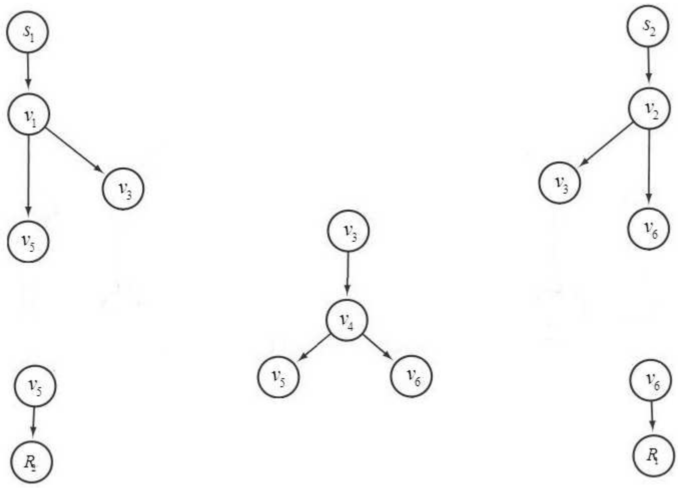

Figure 3

3.5. Transition Condition

We assume that the transition Condition is true for all transitions in the net graph so it can be omitted.

Remark 1. we say that the transition t represents the source (sink) node $s^{'}$ ($R^{'}$) if and only if it represents the subgraph which contains $s^{'}$ ($R^{'}$).

4. Representing the Butterfly Network and the Grail Network by High Level Petri Nets

4.1. The Butterfly Network

The butterfly network in Figure(1-a) can be partitioned into the subgraphs in Figure(3) through which the same information flows.

So the high level Petri net which its net gragh is depicted in Figure(4) represents the butterfly network , where the transition t_1 represents the source node s_1 , the transition t_2 represents the source node s_2 , the transition t_4 represents the sink node R_1 and the transition t_5 represents the sink node R_2

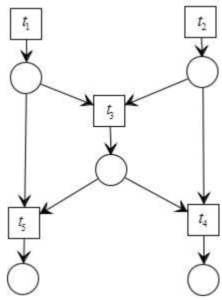

Figure 4

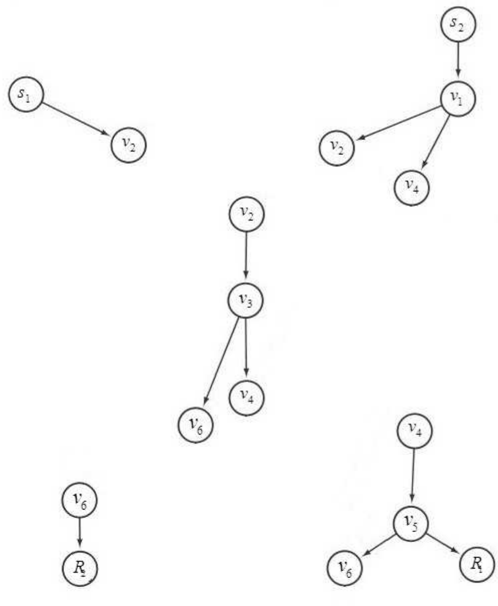

Figure 5

4.2. The Grail Network

The grail network in Figure(2-a) can be partitioned into the subgraphs in Figure(5) through which the same information flows.

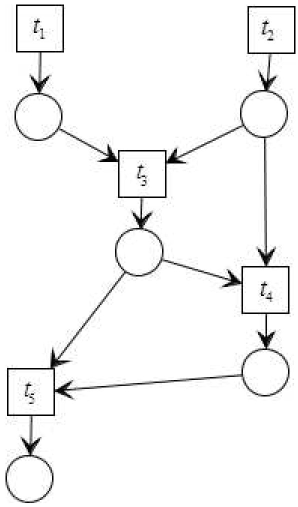

Figure 6

So the high level Petri net which its net gragh is depicted in Figure(6) represents the grail network, where the transition t_1 represents the source node s_1 , the transition t_2 represents the source node s_2 , the transition t_4 represents the sink node R_1 and the transition t_5 represents the sink node R_2 .

5. Representing the Networks which Represent an Expansion of the Butterfly Network (The Grail Network) by High Level Petri Net

In general, an expansion of a graph G is a graph resulting from the subdivision of edges in G . The subdivision of some edge $e = (u, v)$ yields a graph containing one new vertex w , and with an edge set replacing e by two new edges, $(u, w)$and (w, v).

5.1. Representing an Expansion of the Butterfly Network by High Level Petri Net

The network in Figure(7), which represents an expansion of the butterfly network, can be partitioned into the subgraphs in Figure(8) through which the same information flows.

So the network in Figure(7) is represented by the high level Petri which represents the butterfly network, and we note that all networks which represent

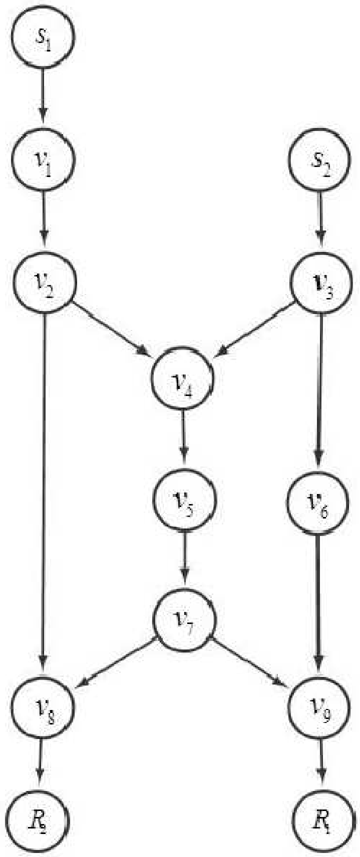

Figure 7

an expansion of the butterfly network are represented by the same high level Petri net.

5.2. Representing an Expansion of the Grail Network by High Level Petri Net

The network in Figure(9), which represents an expansion of the grail network, can be partitioned into the subgraphs in Figure(10) through which the same information flows.

So the network in Figure(8) is represented by the high level Petri which represents the grail network, and we note that all networks which represent an expansion of the grail network are represented by the same high level Petri net.

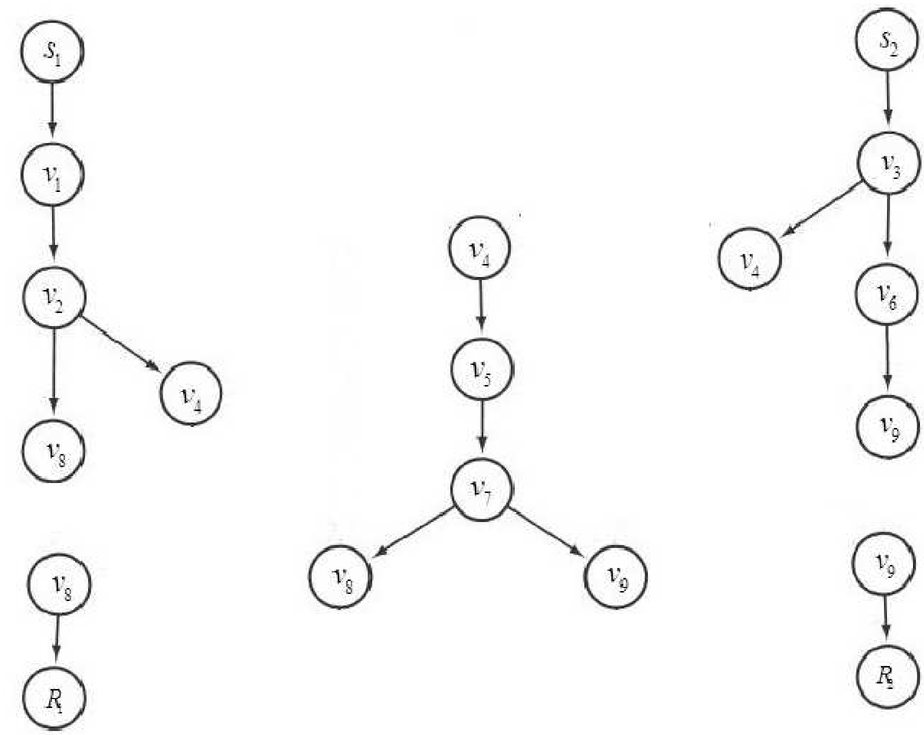

Figure 8

Remark 2. The communication network in Figure(11) is represented by the high level Petri net in Figure(12).

After we represent the butterfly network, grail network and all networks which represent an expansion of butterfly and grail networks we note that:

1. The butterfly network and all its expansions are represented by the same high level Petri net.

2. The grail network and all its expansions are represented by the same high level Petri net.

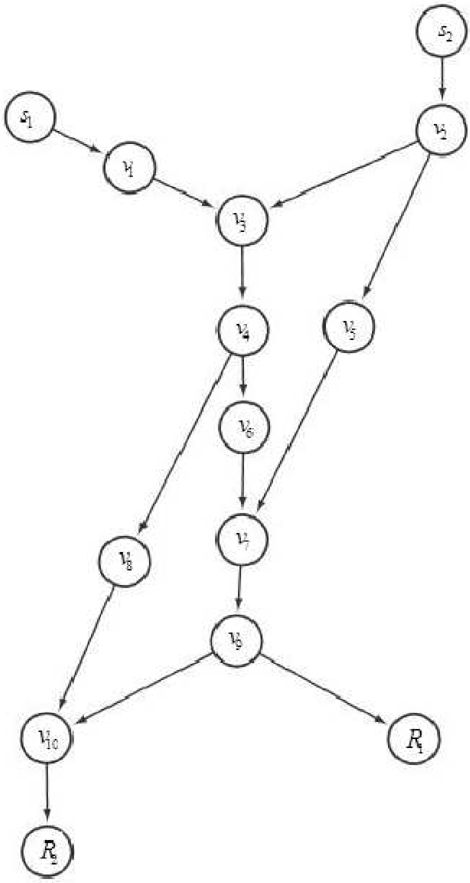

Figure 9

6. Main Result

Theorem 2. *Suppose there exists a network coding solution to the pairwise intersession network coding problem over the communication network G . Then one of the following conditions must hold.*

1. G Contains a subgraph G' which is represented by the high level Petri net which its net gragh is depicted in Figure(12).

2. G Contains a subgraph G' which is represented by the high level Petri net which its net gragh is depicted in Figure(4).

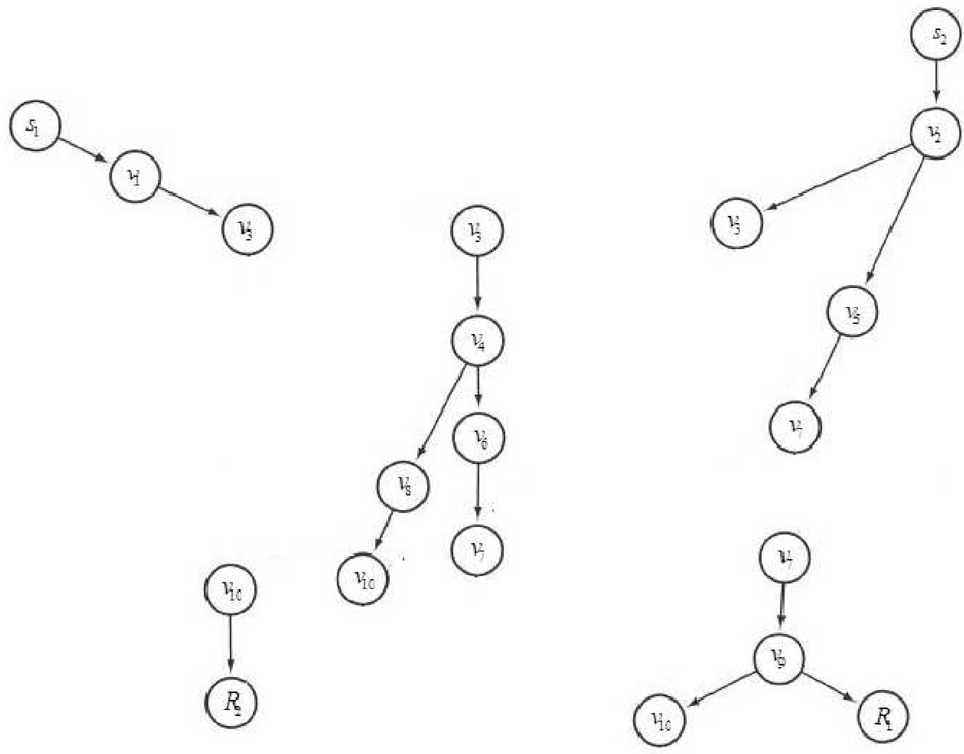

Figure 10

3. G Contains a subgraph $G^{'}$ which is represented by the high level Petri net which its net gragh is depicted in Figure(6).

Proof. It comes directly from the *theorem 1* and the representation of the pairwise intersession network coding by high level Petri nets. □

References

[1] R. Ahlswede, N. Cai, R. Li, and R. W. Yeung, Network information flow, *IEEE Trans. on Information Theory*, **46**, No. 4 (2000), 1204–1216, **doi:** 10.1109/18.850663.

[2] R. Li, R. W. Yeung, and N. Cai, Linear Network Coding, *IEEE Transactions on Information Theory*, **49**, NO. 2 (2003), 371-381, **doi:** 10.1109/TIT.2002.807285.

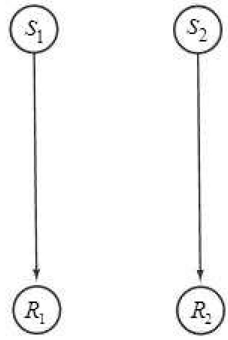

Figure 11

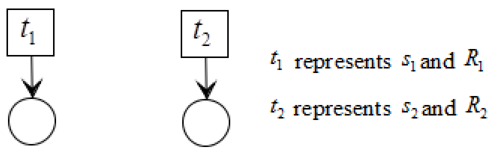

Figure 12

[3] R. Koetter and M. Medard, Beyond routing: An algebraic approach to network coding, *21st Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOMM)*, **1** (2002), 122-130, **doi:** 10.1109/INFCOM.2002.1019253.

[4] S. Jaggi, P. Sanders, P. A. Chou, M. Effros, S. Egner, K. Jain, and L. Tolhuizen, Polynomial Time Algorithms For Network Information Flow, *15th ACM Symposium on Parallel Algorithms and Architectures*, (2003), 286-294, **doi:** 10.1145/777412.777464.

[5] Y. Wu, K. Jain, and S.Y. Kung, A unification of network coding and tree-

packing (routing) theorems, *IEEE Transactions on Information Theory*, **52**, No. 6 (2006), 2398-2409, **doi:** 10.1109/TIT.2006.874430.

[6] D. S. Lun, N. Ratnakar, R. Koetter, M. Medard, E. Ahmed, and H. Lee, Achieving minimum-cost multicast: A decentralized approach based on network coding,*24th Annual Joint Conference of the IEEE Computer and Communications Societies*, **3** (2005) , 1607 - 1617, **doi:** 10.1109/INFCOM.2005.1498443.

[7] J. Sundararajan, M. Medard, R. Kotter, and E. Erez, A systematic approach to network coding problems using conflict graphs, *in Proc. ITA Workshop*, (2006), Invited paper.

[8] O. Al Aboud, M. J. Al Laban and A. Hanano, Using High Level Petri Nets to Model the Inter-Session Network Coding, *International Journal of Pure and Applied Mathematics* (in-press).

[9] K A. Khreishah, Chih-Chun Wang, N. B. Shroff, Rate Control With Pairwise Intersession Network Coding, *Transactions on Information Theory*, **18**, No. 3 (2010), 816-829, **doi:** 10.1109/TNET.2009.2032353.

[10] C. Wang and N. Shroff, Pairwise Intersession Network Coding on Directed Networks, *Information Theory, IEEE Transactions on*, **56**, No. 8 (2010), 3879-3900, **doi:** 10.1109/TIT.2010.2050932.

[11] Software Engineering, ISO/IEC/JTC1/SC7, High-Level Petri Net Concepts Definitions and Graphical Notation, *ISO/IEC 15909-1, Final Committee Draft*, (2002).

[12] C. Wang and N. Shroff, Beyond the Butterfly: A Graph-Theoretic Characterization of the Feasibility of Network Coding with Two Simple Unicast Sessions, *IEEE International Symposium on Information Theory*, (2007), 121-125, **doi:** 10.1109/ISIT.2007.4557214.

[13] C. Wang and N. Shroff, Intersession Network Coding for Two Simple Multicast Sessions, *in Proc. 45th Annual Allerto*, (2007), 682-689.

[14] T. Murata, Petri nets : Properties, analysis and application, *in Proceedings of the IEEE*, **77**, No. 4 (1989), 541-580, **doi:** 10.1109/5.24143.