COMPARATIVE ANALYSIS OF CLASS ALGORITHMS FOR NON-CONFLICT SCHEDULE IN SWITCHING NODES

K Kolchakov
Institute of Information and Communication Technologies, Bulgarian Academy of Sciences
kkolchakov@abv.bg

In the paper are discussed ten algorithms for non-conflict schedule obtaining in the switching nodes of type Crossbar. Comparative analysis of algorithms gives an overview of their potentiality related to the speed, required memory and performance as a function of the size **N** of the connection matrix **T**.

Key words: Network nodes, Message switching Node traffic, Crossbar switch, Conflict elimination, Packet messages, Sparse matrix.

1. Introduction

Discusses ten algorithms to obtain conflict-free schedule in the switching nodes of type Crossbar. Algorithms solve the problem of conflicts in the switching nodes. The conflicts are available in the following two cases:

- When one source of massage requests communication to two or more massage receivers
- When one massage receiver receives communication requests from two or more massage sources.

The evasion of the conflicts is directly related to the switching node performance.

The status of the switch of the switching node is represented with the so called connection matrix. For N x N dimensional switch the dimension of the connection matrix T is N x N also, where every member $T_{ij} = 1$ if the connection request from i-source to j- receiver exists. In the opposite case $T_{ij} = 0$.

A conflict situation arises if any row of the connection matrix has more than a single 1, which corresponds to the case when one source requests a connection with more than one receiver. The presence of more than a single 1 in any column of the matrix T also indicates a conflict situation, it means that two or more sources have requested a connection with the same receiver.[1].

The report is a comparative analysis of six algorithms in terms of speed, required memory and performance in different sizes of connection matrix through eight software models.

A software models performance **(P)** is defined as a ratio of the non- nil resolutions to the total number of the solutions. **R(v)** is the set of the nil solutions, **R(w)** is the set of the non-nil solutions, and **R** is a set of the all solutions[1].

2. An Approach for conflict issue solving

One of the possible approaches for conflict issue solving is based on the research on the algorithms for non-conflict schedules obtaining:

1. Classic algorithm with masks matrixes (**CMA**) [9].
2. Algorithm with joint mask matrixes (**JMA**) [9].
3. Classic algorithm without masks matrixes (**CWA**)[10].
4. Algorithm considering the massage direction (**DAA**) [12].
5. An algorithm by diagonal connectivity matrix activation (**ADA**)[3].
6. Algorithm with joint diagonals activations (**AJDA**)[1].
7. Algorithm with diagonal activations of joint sub-switching matrices (**ADAJS**)
8. An algorithm by diagonal connectivity matrix activation by finite automat (**ADAFA**).
9. Algorithm with joint diagonals activations by finite automat (**AJDAFA**).
10. Algorithm with diagonal activations of joint sub-switching matrices by finite automat (**ADAJSFA**).

Software models are used for the algorithms investigation. Software models are written in Matlab programming language and tested on a Workstation Dell Precision 420.

The software models match completely the algorithms for non-conflict schedule obtaining.

1. A software model based on the classic algorithm with masks matrixes (**SMCMA**) is described and examined in [8].
2. A software model based on the algorithm with joint mask matrixes (**SMJMA**) is described and examined in [8].
3. A software model based on a classic algorithm without mask matrixes (**SMCWA**) is described and examined in [8].
4. A software model based on a algorithm considering the massage direction (**SMDAA**) is described and examined in [8].
5. A software model based on a classic algorithm with spare mask matrixes (**SMCSM**) is described and examined in [5].
6. A software model based on the algorithm with joint spare mask matrixes (**SMJSM**) is described and examined in [5].
7. A software model based on the algorithm by diagonal connectivity matrix activation (**SMADA**) is described and examined in [3].
8. A software model based on the algorithm with joint diagonals activations (**SMAJDA**) is described and examined in [1].
9. A software model based on the algorithm with diagonal activations of joint sub-switching matrices (**SMADAJS**).
10. A software model based on the algorithm by diagonal connectivity matrix activation by finite automat (**SMADAFA**).
11. A software model based on the algorithm with joint diagonals activations by finite automat (**SMAJDAFA**).
12. A software model based on the algorithm with diagonal activations of joint sub-switching matrices by finite automats (**SMADAJSFA**).

The results of the examination of the software models speed (**S**) are represented graphically on Figure 1 and the results of the examination of the software models required memory **M** are represented graphically on Figure 2.

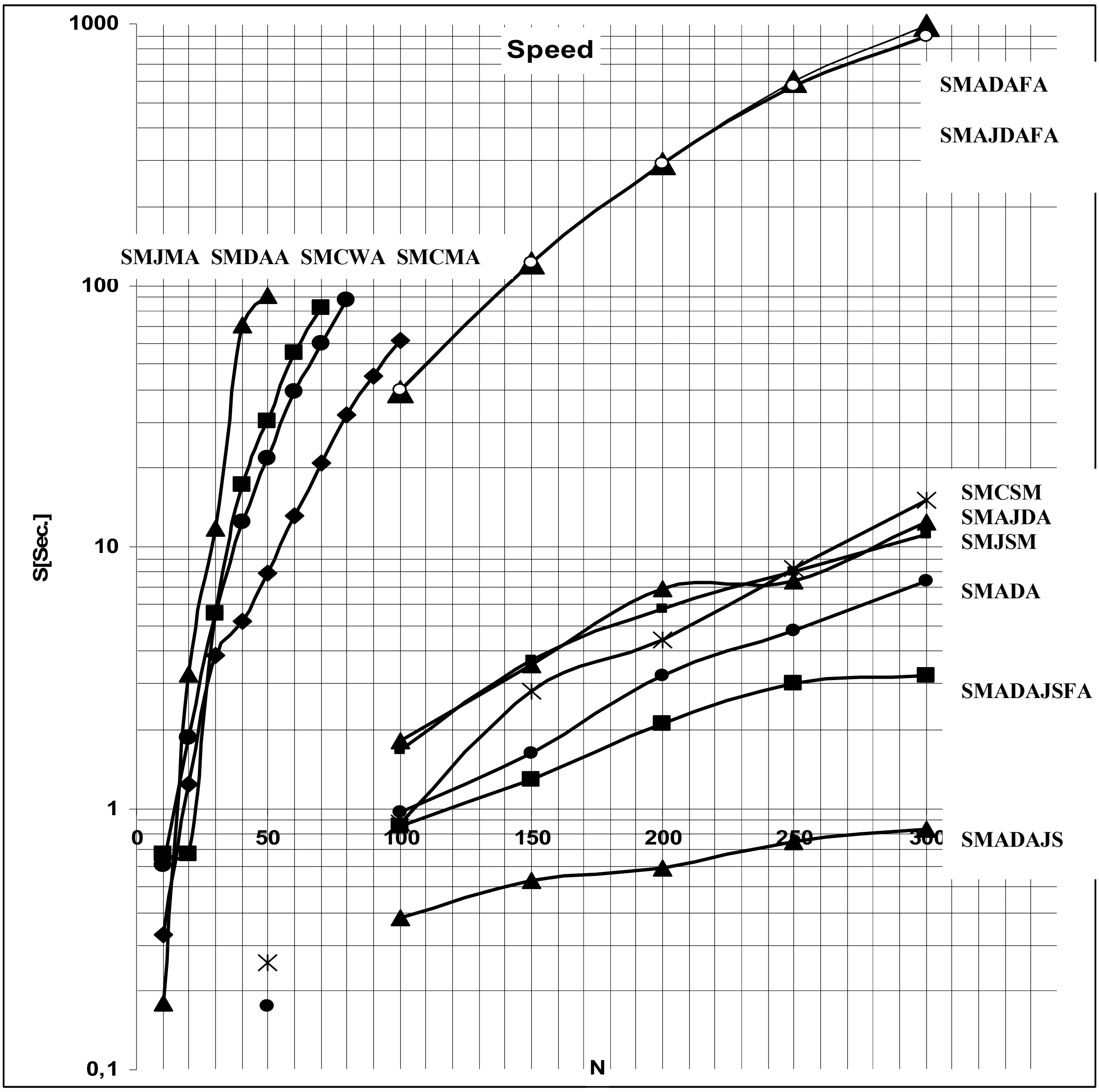

Figure 1: Speed.

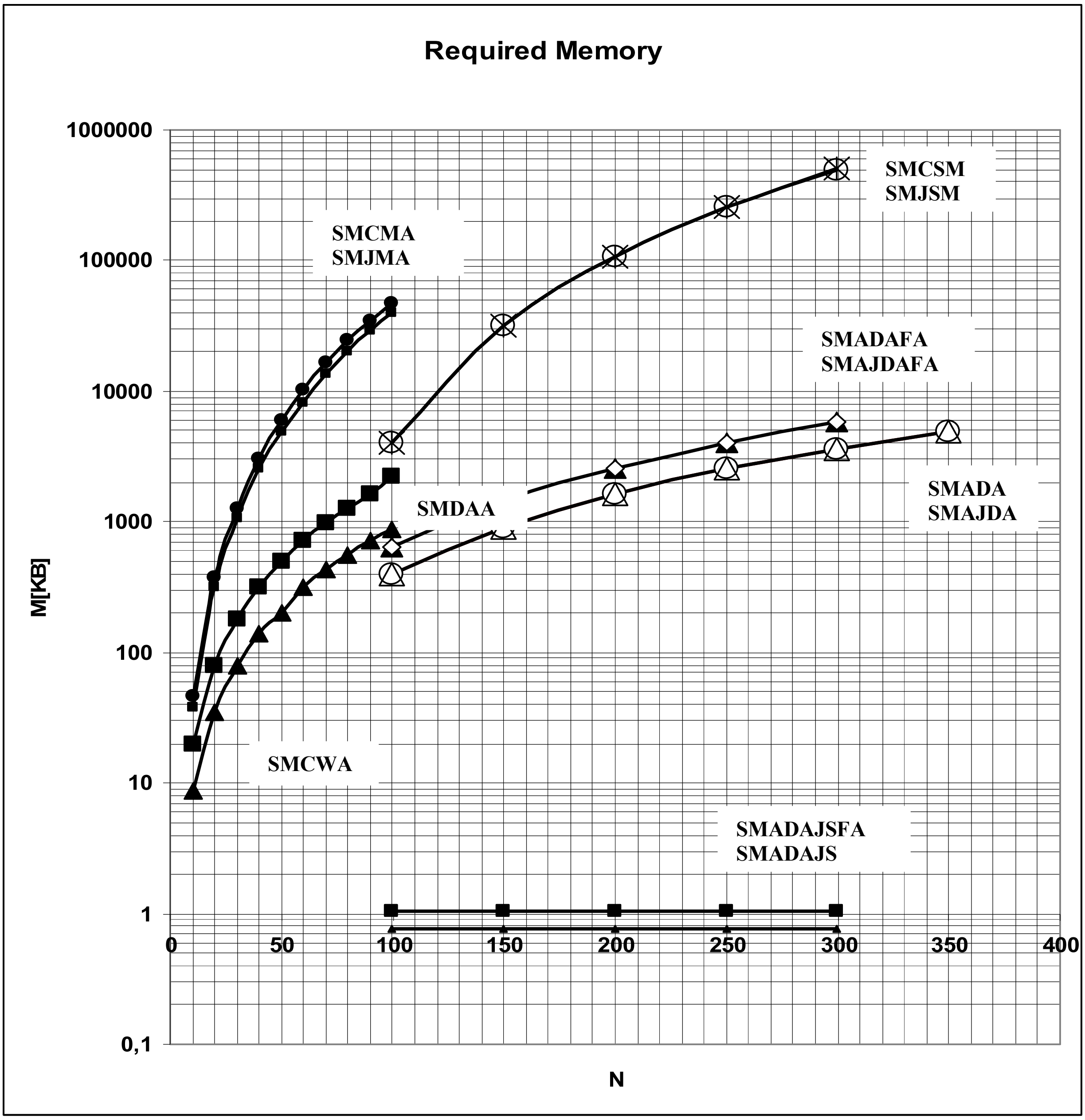

Figure 2 : Required memory.

3. Software models performance.

A software models performance **(P)** is defined as a ratio of the non- nil resolutions to the total number of the solutions. **R(v)** is the set of the nil solutions, **R(w)** is the set of the non-nil solutions, and **R** is a set of the all solutions[1].

R=R(v)+R(w) **(1)**

P=(R(w)/R).100[%] **(2)**

From formula 2 it is visible that when the nil solutions **R(v)** vanish to nil, than the performance P vanish to 100%[1].

To facilitate the performance examination, 5 kinds of matrixes for simulation of the input connectivity matrix **T** are chosen. The special input matrixes **2A, 2B, 2C, 2D** and **2E**[1]are represented on **Figure 3**.

In **Table 1** are represented the investigation results related to the performance **P** of the software models. In the software models that match algorithms using finite automats (**SMADAFA, SMAJDAFA and SMADAJSFA**) can not be determined P because the finite automats do not detect zero solutions.

1	**0**	**0**	**0**	**0**
0	**1**	**0**	**0**	**0**
0	**0**	**1**	**0**	**0**
0	**0**	**0**	**1**	**0**
0	**0**	**0**	**0**	**1**

2A

1	**1**	**0**	**0**	**0**
1	**1**	**1**	**0**	**0**
0	**1**	**1**	**1**	**0**
0	**0**	**1**	**1**	**1**
0	**0**	**0**	**1**	**1**

2B

0	**0**	**1**	**1**	**1**
0	**0**	**0**	**1**	**1**
1	**0**	**0**	**0**	**1**
1	**1**	**0**	**0**	**0**
1	**1**	**1**	**0**	**0**

2C

0	**0**	**0**	**1**	**1**
1	**0**	**1**	**0**	**1**
0	**1**	**1**	**0**	**0**
1	**1**	**1**	**1**	**0**
1	**0**	**1**	**1**	**1**

2D

1	**1**	**1**	**1**	**1**
0	**1**	**1**	**1**	**1**
0	**0**	**1**	**1**	**1**
0	**0**	**0**	**1**	**1**
0	**0**	**0**	**0**	**1**

2E

Figure 3: Five special input matrixes.

P[%]	2A	2B	2C	2D	2E
SMCMA	6,66%	20%	80%	86,6%	53,3%
SMJMA	12,5%	37,5%	87,5%	100%	100%
SMCWA	6,66%	20%	80%	86,6%	53,3%
SMDAA	13,3%	33,3%	80%	93,3%	53,3%
SMCSM	6,66%	20%	80%	93,3%	53,3%
SMJSM	12,5%	37,5%	87,5%	100%	100%
SMADA	6,66%	20%	80%	80%	53,3%
SMAJDA	6,66%	20%	80%	80%	100%
SMADAJS	20%	60%	80%	80%	100%

Table1: Performance.

4.Conclusion

SMJMA, SMDAA, SMCWA, SMCMA, SMADAFA and **SMAJDAFA** software models are slow and could be used in case of N less than 100.

Of the rest six software models **SMADAJS** is the fastest, followed by **SMADAJSFA**, as **SMCSM**, **SMAJDA** and **SMJSM** are almost equivalent.

In terms of memory needed, the fastest **SMADAJS** is the most economical, followed by **SMADAJSFA**.

SMADAJS is the optimum considering rapidity and minimum of needed memory.

The traffic is presented usually the best by the input matrix of type **2D**. Performance of all software models for **2D** is equal to or greater than 80%. **JMA** algorithm is optimum related to the performance **P** in case of **2D** input matrix. , The JMA algorithm is presented by two software models **SMJMA** and **SMJSM** depending on the used matrix masks.

REFERENCES

1. Kolchakov K., Taschev T. "An Algorithm of Non-conflict Schedule with Joint Diagonals Activation of Connectivity Matrix", Proceedings of the International Conference on Computer Systems – CompSysTech'12, 22-23 June 2012, Ruse, Bulgaria, ACM PRESS, ICPS, VOL.630, pp.245-250, ISBN 978-1-4503-1193-9.
2. Kolchakov K. Examination on Algorithms for Non Conflict Schedule with Diagonal Activation in Case of a Large Size Switching Matrix. International Conference Automatics and Informatics `12 03-05.10.2012 Sofia, Bulgaria. Proceedings CD: ISSN 1313 – 1869. pp.341-344.

3. Kolchakov K., "An Algorithm Synthesis of Non-Conflict Schedule by Diagonal Connectivity Matrix Activation" Proceedings of the International Conference AUTOMATICS AND INFORMATICS`11, John Atanasoff Society of Automatics and Informatics, Bulgaria, Sofia 03.10-07.10.2011., pp. B-247 – B251, Proceedings ISSN 1313-1850, CD ISSN 1313-1869
4. K. Kolchakov, "An Approach for Performance Improvement of Class of Algorithms for Synthesis of Non-conflict Schedule in the Switch Nodes" , Proceedings of the 11th International Conference on Computer Systems and Technologies, CompSysTech'10, 17-18 June 2010, Sofia, Bulgaria, pp.235-239, ISBN 978-1-4503-0243-2.
5. Kolchakov K., Software Models with Sparse Mask Matrixes for Non-Conflict Schedule obtaining in a Switching Node, Доклад в трудове на международна конференция (DCCN – 2009), "Distributed Computer and communication networks DCCN 2009", Moscow: R&D Company "Information and Networking technologies" ISBN 978-5-9901871-1-5 София, България, Октомври 5 – 9, 2009, стр. 121-126.
6. K.H.Kolchakov, An approach for non-conflict schedule synthesis – modeling and optimization, Доклад в Сборник научных трудов том 10 Интелектуальные системы и технологии УДК 004.896(06), Научная сесия МИФИ – 2008, Москва 2008, стр. 100 – 101, ISBN 978-5-7262-0883-
7. Kolchakov K.H., Non-conflict Schedule Synthesis Based on Communication Matrix Scanning, Доклад в Сборник научных трудов том 3 Интелектуальные системы и технологии, Научная сесия МИФИ – 2007, Москва 2007, стр.132 - 133.
8. Kolchakov K., A Modeling Approach for Obtaining of a Non – Conflict Schedule, Доклад в трудове на международна конференция (DCCN – 2007), Москва, Русия, Септември 10 – 12, 2007, стр. 168 – 173.
9. К.Колчаков "Моделиране на подход с матрици-маски за получаване на безконфликтно разписание в комутационен възел".Работна статия в поредицата на ИИТ-БАН, IIT/WP-238B, Ноември 2007.
10. K.Kolchakov, Algorithm for Synthesis of Non-conflict Schedule in a Communication Node, Доклад на Българо-руския семинар (DCCN – 2006), Sofia, Bulgaria, Oktober 30 - November 2, 2006, pp. 149 – 155.
11. K.H.Kolchakov, Synthesis approach of optimized non-conflict schedule in communication node, Доклад в Сборник научных трудов том 3 Интелектуальные системы и технологии, Научная сесия МИФИ – 2005, Москва 2005, стр. 162-163.
12. Kolchakov, K., Synthesis of an Optimized Non-conflict Schedule Accounting the Direction of Messages Transfer in a Communication Node. Cybernetics and Information Technologies, Bulgarian Academy of Sciences, Vol. 4, No 2, 2004, pp. 88-95, Sofia.
13. D. Kim, K. Lee and H. Yoo, " A Reconfigurable Crossbar Switch with Adaptive Bandwidth Control for Networks-on-Chip", IEEE International Symposium on Circuits and Systems, 2005.

14. J. Chang, S. Ravi and A. Raghunathan, “FLEXBAR: A crossbar switching fabric with improved performance and utilization”, IEEE Custom Integrated Circuits Conference, May 2002.
15. D. Serpanos, P. Antoniadis:”FIRM: A Class of Distributed Scheduling Algorithms for High-Speed ATM Switches with Multiple Input Queues”, *IEEE Infocom 2000 Conference,* Tel Aviv, Israel, March 2000.
16. M.Schaller, K. Svozil, “Automaton Logic”, International Journal of Theoretical Physics, Vol.35, No. 5, pp. 911-940, 1996, Plenum Publishing Corporation-Springer.