# INVESTIGATION OF THE SPATIAL DISTRIBUTION ALGORITHM OF THE TRAFFIC FLOW ANALYSIS AND OF THE ENTITY FLOW-PHASE ANALYSIS

Gábor Lencse
Department of Telecommunications
Széchenyi István University
Egyetem tér 1.
H-9026 Győr, Hungary
e-mail: lencse@sze.hu

László Muka
Elassys Consulting Ltd.
Bég utca 3-5.
H-1022 Budapest, Hungary
e-mail: muka.laszlo@elassys.hu

## KEYWORDS

discrete-event simulation, traffic-flow analysis, entity flow-phase analysis, information and communication systems, business process systems

## ABSTRACT

This paper investigates an important algorithm that is used in both the Traffic-Flow Analysis and the Entity Flow-phase Analysis. These methods are similar to each other and can be used for the fast and approximate (performance) analysis of Information and Communication Technology (ICT) systems and Business Process (BP) systems. Both methods contain an algorithm for the spatial distribution of the traffic (or entities) in the system. It is shown how the error of the spatial distribution can be measured, and the effect of the so called *size of routing unit* parameter of two algorithms is investigated.

## INTRODUCTION

### Performance Analysis Methods

Discrete-Event Simulation (DES) is a widely used method for the performance analysis (Jain 1991) of Information and Communication Technology (ICT) systems and Business Process (BP) systems. There are a large number of various methods used to describe the behaviour of complex systems (Banks et al. 1996; Bratley et al. 1986; Jávor 1985; Jávor 1993). The simulation of large and complex systems requires a large amount of memory and computing power that is often available only on a supercomputer. Efforts are made to use multiprocessor systems or clusters of workstations. The conventional synchronisation methods for parallel simulation (e.g., conservative, optimistic) (Fujimoto 1990) use event-by-event synchronisation and they are unfortunately not applicable to all cases, or do not provide the desirable speedup. The Statistical Synchronisation Method proposed by Pongor (Pongor 1992) does not exchange individual messages between the segments but rather the statistical characteristics of the message flow. This method can produce excellent speed-up (Lencse 1998) but has a limited area of application (Lencse 1999).

The fast (preliminary and approximate) performance estimation can be very useful in the early design state of an ICT or a BP system. We have proposed the Traffic-Flow Analysis (Lencse 2001) for the rapid performance estimation of ICT systems and the Entity Flow-Phase Analysis (Lencse and Muka 2006) for the fast investigation of BP systems.

### Traffic-Flow Analysis

TFA is a combination of simulation and analytical and/or numerical methods. While the traditional discrete-event simulation models the travelling of each packet through the network, TFA uses statistics to model the networking load of applications. TFA works in two steps:

- In the first step, the method distributes traffic (the statistics) in the network, using the normal routing rules of the network.
- In the second part, the influences of the finite line and switching-node capacities are calculated.

The important features of TFA:

- The results are approximate but the absence or the place of bottlenecks is shown by the method.
- The execution time of TFA is expected to be significantly less than the execution time of the detailed simulation of the system.
- TFA describes the steady state behaviour of the network.

### Entity Flow-phase Analysis

EFA has been derived from TFA by applying the TFA principles for BP systems. Methods of EFA (one-phase-method and multi-phase-method) are based on the same principles as TFA, only the interpretation of the model elements is different. The statistics represent entities (not messages) and the interpretation of the routing is also different. While the packets of a network usually do not multiply, the entities may fork (and the descendants must meet somewhere) or split (and the descendants live their own life separately); see more details in (Lencse and Muka 2006).

*From now on we will focus on TFA, knowing that our results can also be applied for EFA.*

Though it is not absolutely necessary, we encourage the reader of this paper to read the original paper on TFA (Lencse 2001) for the deeper understanding of the remaining part of this paper.

## THE PROBLEM OF SPATIAL DISTRIBUTION

TFA is a general method, and can be used with any traffic model that satisfies the requirements of TFA for the traffic model. In the original paper, we proposed bit-throughput distribution and packet-throughput distribution (practically histograms) as traffic models to model the traffic on the lines and in the nodes, respectively.

The traffic model is always an *aggregated traffic model*, that is, it represents the complete traffic of a given type of applications that are connected to the given node. For example it represents the full traffic (in both directions) of 35 FTP applications that are connected to a router (by switches). If static routing is used, we can handle the complete traffic of the before mentioned 35 FTP applications (or 100 web browsers or any other type of applications) together: we must route only one statistics package through the network (containing the two types of histograms). However, if we have adaptive routing, then the traffic of a given type of application should not be handled together, rather it must be routed in multiple packets, each of which represent a given portion of the traffic of the given type of application connected to the given node. When determining the **size of the routing unit** ($S_{RU}$) we must consider the following issues:

The larger $S_{RU}$ we choose, the fewer statistics packages are to be routed in the first phase and the less traffic model addition is to be performed in the second phase of TFA. However, if $S_{RU}$ is too large, the spatial distribution of the traffic may considerably differ from the one that is formed in the detailed simulation of the system (and from the one in the real system). If $S_{RU}$ is small, the spatial distribution of the traffic may be quite precise, but the larger amount of messages to be routed and traffic models to be added slow down the analysis. *The choice of $S_{RU}$ must be a reasonable compromise (between the contradicting requirements) that is made in the knowledge of the whole system modelled.*

To be able to determine a good enough value for $S_{RU}$, we need to introduce a measure that expresses how good or bad a given spatial distribution of the traffic in TFA is, that is how well the given spatial distribution in TFA approximates the spatial distribution of the traffic in the detailed (packet-by-packet) discrete-event simulation of the system.

Before the presentation of the method that we propose for the good choice of $S_{RU}$, we introduce some formalism in the next sections.

## FORMALIZATION AND INTRODUCING METRICS

The **capacity matrix** $\underline{\underline{\mathbf{K}}}$=[$\mathbf{k}_{ij}$] describes the capacity of nodes and lines. The capacity matrix is an **n*n** matrix, where **n** is the number of nodes in the network. Matrix element $\mathbf{k}_{ii}$ is the **routing capacity of node i** (measured in: packets per second), and matrix element $\mathbf{k}_{ij}$, where **i≠j** is the **transmission capacity of the line** from node **i** to node **j**, measured in Mbit/s.
If there is no transmission line from node **i** to node **j** then $\mathbf{k}_{ij} = 0$.

$$\underline{\underline{K}} = \begin{bmatrix} k_{1,1} & k_{1,2} & \cdots & k_{1,j} & \cdots & k_{1,n} \\ k_{2,1} & k_{2,2} & \cdots & k_{2,j} & \cdots & k_{2,n} \\ \vdots & \vdots & & \vdots & & \vdots \\ k_{i,1} & k_{i,2} & \cdots & k_{i,j} & \cdots & k_{i,n} \\ \vdots & \vdots & & \vdots & & \vdots \\ k_{n,1} & k_{n,2} & \cdots & k_{n,j} & \cdots & k_{n,n} \end{bmatrix}$$

The **cost matrix** $\underline{\underline{\mathbf{C}}}$=[$\mathbf{c}_{ij}$] defines the cost of communication through the network. Matrix element $\mathbf{c}_{ii}$ is the cost of the routing of a packet in node **i** and $\mathbf{c}_{ij}$ (**i≠j**) is the cost of transmission of 1Mbit of information through the line from node **i** to node **j**. If there is no line from node **i** to node **j** then $\mathbf{c}_{ij} = 0$.

The communication through the network is described by the **traffic matrix** $\underline{\underline{\mathbf{T}}}$=[$\mathbf{t}_{ij}$]. Matrix element $\mathbf{t}_{ii}$ describes result of TFA for node **i**: both the packet-throughput distribution and the delay distribution of TFA resulted in node **i**, and matrix element $\mathbf{t}_{ij}$ describes the result of TFA for the line from node **i** to node **j**: both the bit-throughput distribution and the delay distribution of the line from node **i** to node **j**.

For the evaluation of the results of the TFA distribution procedure, let us determine the **empirical load (utilization) matrix** $\underline{\underline{\mathbf{R}}}$=[$\mathbf{r}_{ij}$], that gives us the simulation based load of every node and line:

$r_{ii}$ = (the average number of packets for the node i)/ $k_{ii}$

$r_{ij}$ = (the average number of Mbits for the line i→j)/ $k_{ij}$ .

When different applications give us different matrices, then we can interpret the distance of these.

Let this be:

$$\| \underline{\underline{\mathbf{R}}}_1 - \underline{\underline{\mathbf{R}}}_2 \|_1 \text{ where } \|A\|_1 = \sum_{i=1}^{n} \sum_{j=1}^{n} |a_{ij}| \ .$$

This way for example, we can calculate whether the increased accuracy gained from the more detailed simulation is proportional to the increased processor time usage.

For the second phase of TFA we perform summation of traffic/load (see Lencse 2001). We may also decide to compare the resulting distributions.

In case of the comparison of distributions, it is advised to use the statistical distribution fitting method ($\chi^2$ test, Hunyadi at al. 1996)).

Then the distance of the distributions is:

$$d = \chi^2 = \sum_{i=1}^{10} \left(f_i - g_i\right)^2 / f_i \text{ (grade of freedom 9)}$$

where $g_i$ is the observed and $f_i$ is the expected frequency for bin i.

Note that in case of d < 16,9 the distributions are considered equal, with confidence level of 95%.

For the evaluation of the results of the TFA, we introduce the **sample evaluation matrix** $\underline{\mathbf{S}}=[s_{ij}]$. Matrix elements may have the following values:

$s_{ii}$, $s_{ij}$ = **3**, if the number of RUs through a given node or line < 30

$s_{ii}$, $s_{ij}$ = **2**, if the number of RUs through a given node or line ≥ 30 but ≤ 200

$s_{ii}$, $s_{ij}$ = **1**, if the number of RUs through a given node or line > 200

$s_{ij}$ = **0**, if no line exists from node i to node j .

To compare results, we have to summarize all the elements of the $\underline{\mathbf{S}}$ matrix. The lower result is better because the number shows the level of uncertainty.

The **weighted sample evaluation matrix** $\underline{\mathbf{W}}=[w_{ij}]$ can be derived from matrix $\underline{\mathbf{S}}$ just multiplying the elements of $\underline{\mathbf{S}}$ by $k_{ij}/c_{ij}$. The meaning of this multiplication is that, in general, it is more important to have more precise results on large line or node capacities, and the increasing cost is decreasing the weight of a line or node.

The **alternative weighted sample evaluation matrix** $\underline{\mathbf{V}}=[v_{ij}]$ can be derived from matrix $\underline{\mathbf{S}}$ just multiplying the elements of $\underline{\mathbf{S}}$ by $1/(1-r_{ij})$. The meaning of this multiplication is that the less spare capacity we have, it is the more important to have more precise results.

A support matrix $\underline{\mathbf{B}}=[b_{ij}]$ may also be useful in the analysis. Matrix $\underline{\mathbf{B}}$ is a bitmap of the analysed network:

$b_{ii}$, $b_{ij}$ = **1**, if there was RU travelling through the given node or line,

$b_{ii}$, $b_{ij}$ = **0**, if there was not any RU travelling through the given node or line .

## REVEALING THE ROUTING PROPERTIES

We examine a **data communication network** with **nodes** and **lines** between the nodes.

The aim of the routing is to transmit the information through the network with the **least cost and within the shortest time.**

TFA can be used with any routing method; the routing algorithm is the part of the network not of TFA.

### Introducing Statistical Constraints

In statistics (Hunyadi at al. 1996) a sample consisting **N** elements is called as a **small** sample, when **N** < 30. Above a couple of hundreds the sample is a **large** sample and between these boundaries the sample may be looked as an **average** sample.
The results of simulation are reliable if the number of RUs travelling through a node or a line is at least a **several hundred** from an application.

According to considerations mentioned before, the $S_{RU}$ should be determined as to generate *at least 200* statistical packages for a given type of application, otherwise the weight of coincidence would be too high, and the simulation results would not reflect the data or entity traffic on the network correctly.

### Analysing the Routing Behaviour: Building the Routing Decision Tree

To model the decision process of the routing algorithm we use a decision analysis tool, the decision tree (Littlechild and Shutler 1991), and call it Routing Decision Tree (RDT).

We have to reveal the behaviour of the routing algorithm in the network.

Starting approach:

- If we have simulation results we may use it to construct the RDT
- If we have measurement results we may analyse it and then use the results to construct RDT .

### Testing the Network

We make detailed simulation of the network during an appropriate test interval (**$I_T$**) of the examination interval **T**. (**$I_T$** may be equal, for example, to 20% of **T** and may contain intervals considered to be typical.)
Important: the ***track*** (the sequence of nodes and lines for every application and every unit (packet/entity) sent by an application) for each application and for all of the units sent should be ***remembered.***

Now, we consider stopping criteria for the test:

Stopping criterion (S.1) for an application:

- If the number of units sent by an application through all of the lines on the routes of an application ≥ 200, or
- $I_T$ has been spent .

Stopping criterion for the test process:

- Stopping criterion for all applications has been reached or
- $I_T$ has been spent .

A *weak* stopping criterion may result in a less precise routing description.

Weak stopping criterion (S.2.1): stopping criterion based only on line data:

- Number of units from all applications taken together on all of the lines >200 .

Weak stopping criterion (S.2.2): stopping criterion based only on line data:

- Through some satisfactory portion (let us say 80%) of the lines the number of units sent > 200 and there are no *new lines* involved during the last significant interval (let us say during the last 5% of $I_T$ matrix $\underline{\underline{B}}$ is unchanged) of the test .

Weak stopping criterion (S.3): stopping criterion based only on applications:

- All of the applications have sent units > 200.

*Differential* stopping criterion (S.4) (may be combined with other criteria):

We build up the $\underline{\underline{R}}$ matrix:

- Send units
- Build $\underline{\underline{R}}_1$
- Send more units
- Build $\underline{\underline{R}}_2$
- Calculate the distance of $\underline{\underline{R}}_1$ and $\underline{\underline{R}}_2$
- If the distance < lower limit, then stop.

After the test phase, based on track data, we construct RDTs.

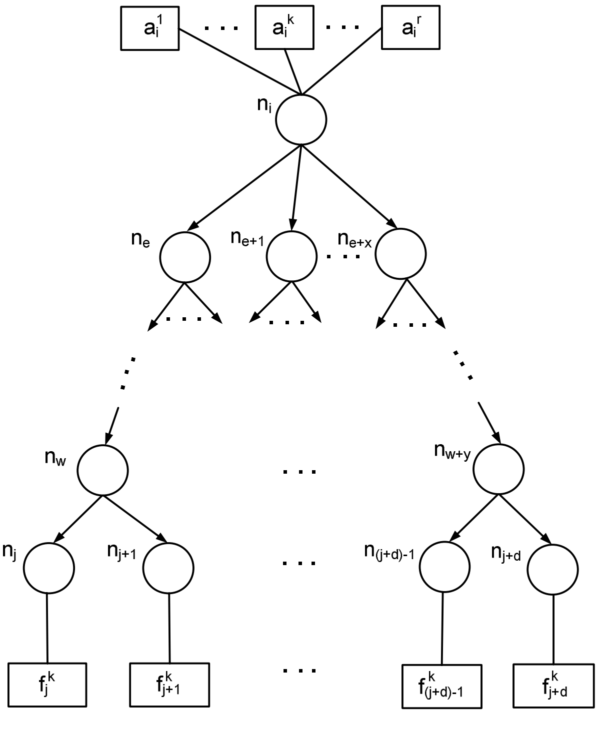

*Figure 1. Routing Decision Tree*

RDT in Figure 1 shows the probability (percentage) based decisions made by nodes on the route starting from the source node ($n_i$) to the destination nodes ($n_j$, …, $n_{j+d}$).

If there are parallel edges in the tree we split them and remember frequencies separately towards the destination (Figure 2).

In case of EFA, we may think about even the **replacing** of the original routing with RDT.

Based on the frequencies got from the test phase the $S_{RU}$ for an application $a_k$ may be calculated:

$$S_{RU} = N_T / \left( \sum_j f_j^k \right) \left( \min f_j^k / 200 \right), \text{ where}$$

$a_i^k$ is an application connected to node **i**.

$N_T$ is the quantity of information (packets for nodes, Mbits for lines) sent by $a_i^k$ during $T$.

$f_j^k$ denotes the measured destination frequency (from $a_i^k$ to node **j**).

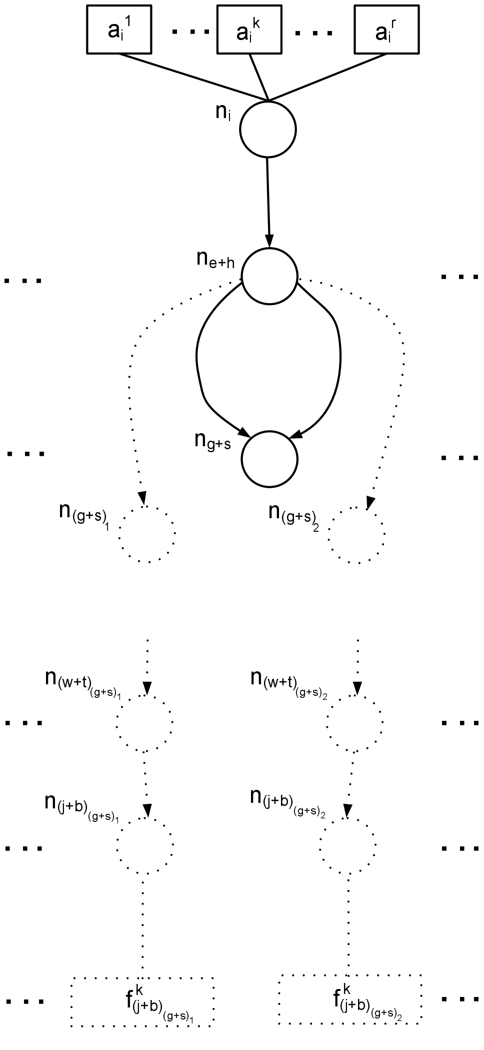

*Figure 2. Splitting Edges in Routing Decision Tree*

## PERFORMING SPATIAL PHASE

Using $S_{RU}$s, calculated in the previous way, we perform the spatial distribution phase of TFA.

### About Dynamic Control of $S_{RU}$: Increasing-Decreasing Decisions During Spatial Phase

Increase $S_{RU}$: during the last significant part of distribution there was no change in the set of lines (routing seems to be static) for all of the applications or for one application.

Decrease $S_{RU}$ one: a new line occurred in the set of lines (there was a change in any element of matrix $\underline{\underline{B}}$). The new line has to be inserted into the RDTs and a new $S_{RU}$ should be calculated.

Decrease $S_{RU}$ two: in the end if *the number of RUs* $< 200$ for a line or a node we may decide to recalculate $S_{RU}$s and repeat the process.

We may also consider using RUs with different sizes on different lines. For example, it may be useful to use smaller RUs in the case of lines in critical or overloaded state. To use smaller RUs instead of the ones that arrived to a given line, a **RU conversion** should be made before entering the line.

If we need to use **smaller** RUs than we have, we generate smaller RUs (for example with exponential distribution) that together represent the same amount of traffic as the original RU represented.

If we need to use **larger** RUs then we have, some smaller RUs are replaced by a larger one. Of course, we can do it only if the smaller ones are present together at a given point of the network, for example waiting in a queue.

### Evaluation of Results

If we have detailed simulation results then using $\underline{\underline{R}}$ we may compare TFA results to detailed simulation results: the closer the results are to detailed simulation results the better the TFA results may be considered.

The sample evaluation matrix $\underline{\underline{S}}$ (together with $\underline{\underline{W}}$ and $\underline{\underline{V}}$) may be used to analyse the reliability of results.

## CONCLUSIONS

We have introduced formal description for the networks and traffic conditions such as: capacity, cost, traffic and utilization matrices as well as metrics for the difference of the traffic and utilization matrices.

On the basis of the statistical constraints on sample size, we have introduced the sample evaluation matrix, the elements of which express if the number of RUs are high enough for a given node or line.

We have given a method, how to calculate the $S_{RU}$ by using the Routing Decision Tree and detailed simulation for an appropriate (short) period of time. We have also given different stopping criteria for the simulation.

We have shown how the $S_{RU}$ can be dynamically controlled during the spatial distribution phase of TFA or EFA.

We conclude that with our results on the appropriate choice of the $S_{RU}$ of TFA or EFA, these methods have been matured for implementation.

## REFERENCES

Banks, J.; J. S. Carson; B. L. Nelson. 1996. *Discrete-Event System Simulation* Prentice Hall, Upper Saddle River, New Jersey

Bratley P.; B. L. Fox; and L. E. Schrage. 1986. *A Guide to Simulation.* Springer-Verlag, New York

Elassys Consulting Ltd. 2007. *ImiNet and ImiFlow Systems* http://www.elassys.hu

Fujimoto, R. M. 1990. “Parallel Discrete Event Simulation” *Communications of the ACM* 33, no 10, 31-53

Hunyadi, L.; Mundruczó, Gy.; Vita, L. 1996. *Statisztika* Aula Kiadó, Budapest

Jain, R. 1991. *The Art of Computer Systems Performance Analysis..* John Wiley & Sons, New York

Jávor, A. (editor) 1985. *Simulation in Research and Development.* North-Holland, Amsterdam

Jávor, A. 1993. *Petri Nets in Simulation* EUROSIM Simulation News Europe, 1993, no. 9, pp. 6-7.

Lencse, G. 1998. “Efficient Parallel Simulation with the Statistical Synchronization Method” *Proceeding of the Communication Networks and Distributed Systems Conference* (CNDS'98), (San Diego, CA, USA, January 11-14 ) SCS, 3-8

Lencse, G. 1999. “Applicability Criteria of the Statistical Synchronization Method” *Proceedings of Communication Networks and Distributed Systems Conference* (CNDS'99), (San Francisco, CA, USA, January 17-20.) SCS, 159-164

Lencse, G. 2001. “Traffic-Flow Analysis for Fast Performance Estimation of Communication Systems” *Journal of Computing and Information Technology* 9, No. 1, 15-27.

Lencse, G., Muka, L. 2006. “Expanded Scope of Traffic-Flow Analysis: Entity Flow-Phase Analysis for Rapid Performance Evaluation of Enterprise Process Systems” *Proceedings of the 2006 European Simulation and Modelling Conference* (ESM'2006) (Toulouse, France, 2006. Oct. 23-25.) EUROSIS-ETI, 94-98.

Littlechild, S. C.; Shutler, M. F. 1991. *Operations Research in Management* Prentice Hall, London

Pongor, Gy. 1992. “Statistical Synchronisation: a Different Approach to Parallel Discrete Event Simulation” *Proceedings of the 1992 European Simulation Symposium* (ESS’92), (Dresden, Germany, Nov. 5-8) SCS Europe, 125-129.

## BIOGRAPHIES

**GÁBOR LENCSE** received his M.Sc. in electrical engineering and computer systems at the Technical University of Budapest in 1994 and his Ph.D. in 2000. The area of his research is (parallel) discrete-event simulation methodology. He is interested in the acceleration of the simulation of info-communication systems. Since 1997, he works for the Széchenyi István University in Győr. He teaches computer networks and networking protocols. Now, he is an Associate Professor. He is a founding member of the Multidisciplinary Doctoral School of Engineering, Modelling and Development of Infrastructural Systems at the Széchenyi István University. He does R&D in the field of the simulation of communication systems for the Elassys Consulting Ltd. since 1998. Dr Lencse works part time at the Budapest University of Technology and Economics (the former Technical University of Budapest) since 2005. There he teaches computer architectures.

**LÁSZLÓ MUKA** graduated in electrical engineering at the Technical University of Lvov in 1976. He got his special engineering degree in digital electronics at the Technical University of Budapest in 1981, and became a university level doctor in architectures of CAD systems in 1987. Dr Muka finished an MBA at Brunel University of London in 1996. Since 1996 he has been working in the area of simulation modelling of telecommunication systems, including human subsystems. He is a regular invited lecturer in the topics of application of computer simulation for performance analysis of telecommunication systems at the Multidisciplinary Doctoral School of Engineering, Modelling and Development of Infrastructural Systems at the Széchenyi István University of Győr.