Liet. matem. rink., T. 42, spec. nr., 636–640
© 2002 Matematikos ir informatikos institutas

Automatic aggregate modelling

Aušra ŽVIRONIENĖ (KTU)
e-mail: ausrez@one.lt

Introduction

At this moment the investigation in the direction of specification and simulation of complex systems are implemented. Methods and applications, which bestow on creating of digital and imitation systems' models, are being designed [1], [2]. For designing of imitation models is used the piece-linear aggregate formalism. These created methods are used for designing of models of telecommunication, transport, business and other systems.

As the real object for investigation Ethernet network was taken [3]. One way to describe the aggregate as object is the method of controlling sequences of I.N. Kovalenko and A.A. Borovkov [4], [5], [6], [7], but it is rather difficult to automate him, it is not user-friendly and it takes much time outlay. At the moment the investigated systems are especially complex, so the purpose of this paper is to represent such structural description of the aggregate, that the computer could simulate the wanted aggregate itself and calculate the desirable characteristics.

The aggregate is divided into separate components. Two tasks – the decomposition of system and the composition of those decomposed components into one unit as the system – are being solved. Such system is described by aggregate. In order for the computer itself to model it, I suggest such aggregate model (Fig. 1), that consists of working and distribution modules, and I present its principle schemes of working and distribution modules and composition of these modules.

1. The aggregate model – the principle scheme

Here we analyze the maintenance systems that we describe by aggregates. At random time moments there comes the signal flow into the system. The signal, coming into system, brings the following information: the maintenance duration periods in working modules and reference about time moments of signals coming to the system (Fig. 1).

The arrived signals fall into the common distribution module, from where they accidentally are directed into working modules $D_{p,q}$. In working module, k-th signal for some time a_{kl} is maintained in the l-th level. If arrived signal finds the working module busy, the signal waits, while it will be free (it shown in Fig. 2 by pieces of lines).

The maintained signal gets from working module into distribution module, that is coherent with him, which again directs the signal into further level of working module $D_{p+1,q}$; here the maintained signal is directed into distribution module, that is coherent with him, and so on, until the maintenance is finished.

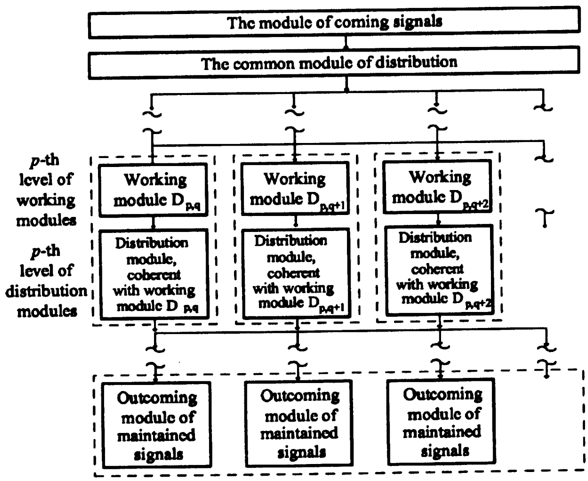


Fig. 1. The aggregate module.

With the help of such aggregate we can investigate the signal flows of Ethernet networks – how much time the signal delays in the network, until it reaches the needed addressee – in other words, the network-load. To specify, at this time this problem is most actual to network engineers, who seek to guaranty the high quality of information transmission. This would also help in calculating other characteristics – on request of the user (for instance, the signal delay).

2. The aggregate working module – analytical and graphical expressions

For description of the working and distribution modules we use the Hevisaid function:

$$\mathbf{1}\,(t - t_k) = \begin{cases} 1, & t \geqslant t_k, \\ 0, & t < t_k. \end{cases}$$

Signals directed to the working module bring information (t_k^*, a_k), $k = 1, 2, \ldots$, where t_k^* – falling moment of k-th signal; and a_k – maintenance time of k-th signal. Then t_k – is finishing moment of maintenance of the k-th signal, variations Δ_1, Δ_2 and $N\,(t)$ realization, generated of working module (Fig. 2) – are calculated by the below given relationships.

$$N\,(t) := 0, \quad t_0^* = t_0 = 0 \text{ – on insertion moment – i.e., the ‘zero’ cycle}$$

$$\ldots$$

$$\begin{aligned}
&t_{k+1} := t_{k+1}^* + N\left(t_{k+1}^*\right) + a_{k+1}, \\
&\Delta_1 := a_{k+1} \cdot \mathbf{1}\left(t_k - t_{k+1}^*\right) \cdot \left(\mathbf{1}\left(t - t_{k+1}^*\right) - \mathbf{1}\,(t - t_k)\right), \\
&\Delta_2 := \mathbf{1}\left(t_k - t_{k+1}^*\right) \cdot \left(a_{k+1} + t_k - t\right) \cdot \left(\mathbf{1}\,(t - t_k) - \mathbf{1}\,(t - t_{k+1})\right) \\
&\qquad + \mathbf{1}\left(t_{k+1}^* - t_k\right) \cdot \left(a_{k+1} + t_{k+1}^* - t\right) \cdot \left(\mathbf{1}\left(t - t_{k+1}^*\right) - \mathbf{1}\,(t - t_{k+1})\right), \\
&N\,(t) := N\,(t) + \Delta_1 + \Delta_2,
\end{aligned}$$

$$\ldots$$

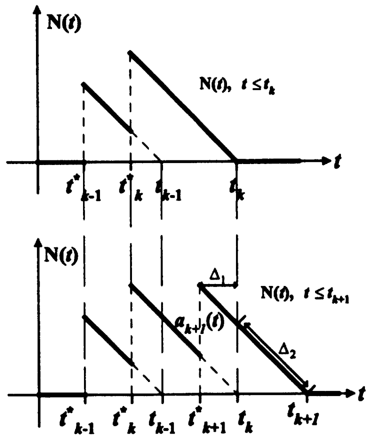


Fig. 2. The working module realization.

Here is the working module of $k+1$-th cycle at the execution time, when the k-th is already realized.

Note, that the working module generates the 'informative' realization $N(t)$ until the end of maintenance moment of last signal (Fig. 2).

3. The aggregate distribution module – graphical and analytical expressions

We do the assumption that the distribution module works autonomously – at his time it is very popular– there are many systems, that are described so [8].

The distribution module is described by pairs of numbers (τ_k, r_k), $r_k = 1, 2, \ldots, m$, which show that in the intervals of time $(s_k, s_{k+1}]$ are the active windows in the r_k-th right line – when $s_k = \sum_{j=1}^{k} \tau_j$, all arrived signals are directed to the r_k-th working module (Fig. 3).

$M(t) := 0$ – on insertion moment – i.e., the 'zero' cycle,

$M(t) := r_1 (\mathbf{1}(t) - \mathbf{1}(t - \tau_1))$ – the 'first' cycle, when the 'zero' cycle is already realized,

...

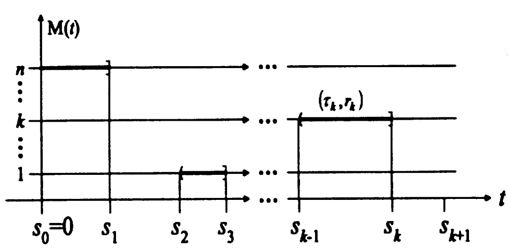


Fig. 3. The distribution module realization.

$$\Delta_3 = r_{k+1}\left(\mathbf{1}\left(t - \sum_{j=1}^{k} \tau_j\right) - \mathbf{1}\left(t - \sum_{j=1}^{k+1} \tau_j\right)\right), \quad k = 1, 2, \ldots,$$
$$M(t) := M(t) + \Delta_3,$$
$$\ldots$$

Here is the $k+1$-th cycle of the distribution module, when k-th is already realized.

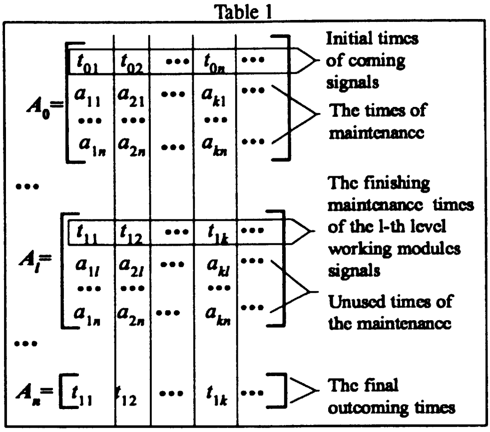
Then the variation Δ_3 and the working realization $M(t)$ of the distribution module, are calculating by the relationships, presented in the above given relationships.

4. The composition of working and distribution modules of aggregate – the principle scheme

We will describe the work of aggregate by using the sequence of matrixes $A_1, A_2, \ldots, A_n$. Their composition is shown in Table 1. Aggregate working and distribution modules are composed by the scheme shown in Fig. 4.

Note, that in matrixes $A_1, A_2, \ldots, A_n$ the times of finishing work in usual way are getting not essentially in growing order. Thus, while beginning to create realization of corresponding level of the working modules, we need to change the columns in corresponding matrix so, that the time sequence (first row of matrixes A_l) would be growing. Aggregate works until the signals arrive to it (the number of signals is finite).

Table 1



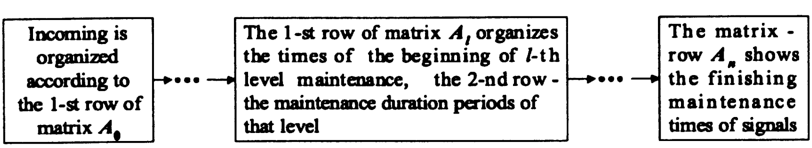


Fig. 4. The composition scheme.

Conclusion

This methodology is suitable to describe the wide class of aggregates. I think that this method will be more user-friendly – this lets the user to make the wanted aggregate himself. In the Faculty of Informatics at Kaunas University of Technology the working programme of presented aggregate scheme is being created and accomplished.

References

[1] K. Wang, H. Pranevičius, Applications of AI to production engineering, *Nordic-Baltic Summer School'97, Lecture Notes*, KTU, Technologija (1997), pp. 269–322.
[2] H. Pranevičius, J.A.G. Knight, I. Pranevičienė, Formal specification and analysis of distributed systems using aggregate approach, *Informacinės technologijos ir valdymas*, 1 (22), 31–38 (2002).
[3] M. Tatipaluma, B. Khasnabish (Eds.), *Multimedia Communications Networks Technologies and Services*, Artech house, Boston-London (1998), pp. 39–102.
[4] И.Н. Коваленко, Н.Б. Бусленко, *Лекции по теории сложных систем*, Москва (1973).
[5] А.А. Боровков, *Вероятностные процессы в теории массового обслуживания*, Наука, Москва, 5–367 (1972).
[6] З.И. Навицкис, Г.И. Пранявичюс, Й.П. Сапаговас, *Математическая модель одноступенчатой системы сбора информации*, *Применение экономико-математических методов и вычислительной техники в непромышленной сфере*, Вильнюс, 113–116 (1973).
[7] A. Žvironienė, R. Tuminauskas, Ethernet komutatoriaus modeliavimas agregatiniu metodu, *Matematika ir matematinis modeliavimas – 2002, Konferencijos pranešimų medžiaga*, KTU (2002), pp. 103–108.
[8] N.Ch. Hock, *Queueing Modelling Fundamentals*, England (2001).

Agregato automatizuotas modeliavimas

A. Žvironienė

Šiuo metu aktualūs uždaviniai apie sudėtingų sistemų tyrimą ir jų charakteristikų skaičiavimą. Pavyzdžiui, tokia sistema yra Ethernet tinklas. Patirtis rodo, kad tokias sistemas patogiausia aprašinėti agregatais.