

MATHEMATICAL MODELLING OF THE R 12 "UNTER"

István Kun – Gábor Farkas

MTA SzTAKI – SzKI

"UNTER" is the abbreviation of Universal Terminal System. It has been developed in the SZKI (Coordination Institute for Computing Techniques).

The main task of the system is to organize the simultaneous work of a few terminals. These terminals are controlled by a R 12 minicomputer. (OS12, partition F1). Their job is to establish a direct interactive communication between an IBM 370/125 computer and the users. Of course the R 12 is not completely exploited by the satellite functions (e.g. preprocessing, postprocessing). In the remaining time it can act as an independent computer.

A special technique, called SPOOL, is applied to perform the input–output operations of the R 12. The functioning of SPOOL is roughly as follows:

A record, entering the system, first goes through a space compression, after which it is stored on a disk file. On the other hand a record, leaving the system, goes through a space decompression (it is restored in the original form), after which it is passed through a channel. Both READ and WRITE operations consist of two parts: a fast processor action initiates a slow peripheral action. Space compression decompression is again a processor job. When a file is transferred, only one of its records can move at the same time. That is this record, until it arrives, blocks the route for the next record of its file. The advantage of this organization is quite evident. The processor can quickly change from one channel to the other, to look for a channel the records of which have a free route to go on, while on the other busy channels the slow peripheral actions are being performed simultaneously. When the processor does not find any channel waiting for it, some other computational work can be done.

For the time being there are 22 channels handled by the R 12. These channels are:

Nr.	Direction	Equipment	
0	I/O	asynchron line to/from the IBM 370/125	
1	I/O	console	
2	I/O	} user terminals (VT 340 displays)	
.			
.			
.			
9	I/O		
10	I		card reader
11	O		line printer
12	I	BG card reader	
13	O	BG line printer	
14		idle	
15	I/O	} user libraries (on the disk)	
.			
.			
.			
22	I/O		

As we have seen before, while a record gets through the core memory, the processor is needed three times. Therefore it is necessary to register for each channel whether and at which phase the processor is expected. This registration is made by SPOOL in three double words:

Task \ Bit Nr.	0	1	2	3	4	...	21	22
READ						...		
COMPR/DECOMPR						...		
WRITE						...		

The usual value of each bit is 0. A change to 1 means a request for the processor. Such a request arises when either a new record enters the system or a peripheral action is finished. Of course there may be no more than one 1 in a column.

The service principle is as follows: The first row is inspected in natural order. If there is a request, it will be served and the inspection recommences from the first bit. The second line comes only when there is no more unsatisfied request in the first row. The satisfaction of a second row request is followed by a return to the first bit of the first row. The third

row is inspected only when there is no more unsatisfied request in the first two rows. Which means that priority decreases from left to right, the last bit of a word being followed by the first bit of the next word.

Of course no request can avoid detection and satisfaction: as we can see from the description of SPOOL, no more than one, often no 1 appears in a column, the latter for much longer periods. Therefore columns with higher priority cannot permanently "capture" the processor.

Mathematical modelling of the system needs some simplifying assumptions:

1. Channels operate in one direction only.
2. Buffers can accept only one record at the same time, as we have supposed.
(Actually: five records).
3. In every channel files arrive according to Poisson processes. File lengths are geometrically, processor and peripheral times are exponentially distributed. All these variables are independent among themselves. (According to experience, some of the service times may be constant, a good approximation of which is the convolution of several exponential distributions. This might be interpreted simply as an increase in the number of exponential service phases.)

Possible states in the case of two channels, with the parameters of the exponential distributions:

	Channel 1			Channel 2	
1.	α_1	Interarrival time	β_1	β_1	Interarrival time
2.		Wait			Wait
3.	α_3	Read/CPU	β_3	β_3	Read/CPU
4.	α_4	Read/Periph.	β_4	β_4	Read/Periph.
5.		Wait			Wait
6.	α_6	Compr./Decompr.	β_6	β_6	Compr./Decompr.
7.		Wait			Wait
8.	α_8	Write/CPU	β_8	β_8	Write/CPU
9.	α_9	Write/Periph.	β_9	β_9	Write/Periph.

States 2,5 and 7 represent the cases when work is interrupted because the processor is engaged with the other channel.

For a given channel the possible states remain the same even if the number of channels is increased.

For N channels the number of possible different states of the system is of course not 9^N but

$$(1) \quad N * 3 * 6^{N-1} + 3^N$$

where the first member gives the number of states when the processor is busy, while the second member gives the number of states when the processor is idle.

Following the method well-known in queuing theory (see e.g. [3]) we can easily set up the birth-and-death type differential equations. The resulting system of equations has the form

$$(2) \quad \underline{\dot{P}}(t) = \underline{A} * \underline{P}(t)$$

where $\underline{P}(t)$ is the vector of the probabilities of possible states and \underline{A} is a matrix with constant elements. We know also that

$$(3) \quad \|\underline{P}(t)\|_{i_1} = 1 \quad 0 \leq t < +\infty .$$

Since (2) is a finite system of linear differential equations with constant coefficients, (2) and (3) give the existence of

$$(4) \quad \lim_{t \rightarrow +\infty} \underline{P}(t) = \underline{P}$$
$$\|\underline{P}\|_{i_1} = 1$$

$$\lim_{t \rightarrow +\infty} \underline{\dot{P}}(t) = \underline{0}$$

so (2) becomes with t

$$(5) \quad \underline{0} = \underline{A} * \underline{P}$$

so a system of linear algebraic equations remains to be solved.

Instead of the detailed description of (5), we try to get some more compact information. For $N = 2$ denote by $P_i(Q_i)$ the probability that the first (second) channel is in the state i , and by $E(F)$ the parameter of the file length distribution in the first (second) channel. Then

$$(6) \quad \begin{aligned} 0 &= -\alpha_1 P_1 + \alpha_9 (1 - E) P_9 \\ 0 &= \alpha_1 P_1 - \alpha_3 P_3 + \alpha_9 E P_9 \\ 0 &= \alpha_3 P_3 - \alpha_4 P_4 \\ 0 &= \alpha_4 P_4 - \alpha_6 P_6 \\ 0 &= \alpha_6 P_6 - \alpha_8 P_8 \\ 0 &= \alpha_8 P_8 - \alpha_9 P_9 \end{aligned}$$

and

$$\begin{aligned}
 (7) \quad & 0 = -\beta_1 Q_1 + \beta_9 (1 - F) Q_9 \\
 & 0 = \beta_1 Q_1 - \beta_3 Q_3 + \beta_9 F Q_9 \\
 & 0 = \beta_3 Q_3 - \beta_4 Q_4 \\
 & 0 = \beta_4 Q_4 - \beta_6 Q_6 \\
 & 0 = \beta_6 Q_6 - \beta_8 Q_8 \\
 & 0 = \beta_8 Q_8 - \beta_9 Q_9
 \end{aligned}$$

Every equation of (6) and (7) is the sum of a few equations in (5). The role of the channels is not symmetrical, due to the priority rule, therefore the identity in the structures of (6) and (7) must have a more general reason. Let us take e.g. the first channel. Let us fix for the stationary distribution

$$(8) \quad w = \frac{\text{waiting time}}{\text{total time}}$$

and consider the remaining $P_i - s$ under the condition (8). If we observe the system only outside the waiting periods, a (2) - type relation for $\underline{P} = (P_1, P_3, P_4, P_6, P_8, P_9)^T$ can be found. (6) will correspond to (5) and of course (8) to the second relation in (4). This reasoning is independent from N , which proves the general validity of (6) for any channel among the N ones.

Of course (5) cannot be eliminated; since (6) gives only the ratio of the $P_i - s$. With $N = 22$ (1) gives a number, which is hopeless for a computer as the dimension of (5). Again some further simplification is necessary.

Suppose that the loadings of the uniform channels 2 - 9 are identical and the same holds for the uniform channels 15 - 22. We try to compress 8 uniform channels into one. This is possible because the peripheral time is 20 - 40 times longer than the processor time. The number of the simultaneously busy channels follows a binomial distribution with expectation K . In the compressed common channel the length of a peripheral period has an exponential distribution with parameter $K\alpha$ where α is the individual parameter. In the compressed common channel the processor period will have a rough but reasonable approximation. We suppose it to be a variable being the product of a negative binomial distribution with expectation K and a constant T , which is the expectation of the individual processor time. Again this distribution has a good approximation by an exponential distribution. This kind of compression works well if for an individual channel the different processor and peripheral periods have nearly the same parameter in the exponential distribution. Of course other ideas of compression may also be used.

The compressed system has 5 channels, which - by means of (1) - involves $5 * 3 * 6^5 - 1 + 3^5 \approx 20000$ equations, each of which have about 10 non-zero elements. This can already be handled with available computers.

Bibliography

- [1] E. G. Coffman: Computer and job-shop scheduling theory. John Wiley and Sons, New York, 1976.
- [2] C. Everling: Exercises in computer systems analysis Lecture Notes in Operations Research, No. 65. Springer Verlag, Berlin – Heidelberg – New York (1972).
- [3] L. Takács: "Priority Queues", Operations Research. 12 (1964).

Összefoglaló

Az R12 "UNTER" matematikai modellezése

Kun István – Farkas Gábor

A cikk sorbanállási modellt ad az SZKI-ban kifejlesztett R12 számítógép perifériáinak működését szabályozó rendszerre. Az egzakt modell a rendszer lehetséges állapotainak száma miatt megoldhatatlan. A cikkben tárgyalt kielégítő pontosságú közelítésekkel azonban a modellt megoldható méretűvé lehet redukálni.

Резюме

Математическое моделирование P12 "УНТЭР"

Иштван Кун – Габор Фаркаш

Настоящая статья дает модель массового обслуживания к системе, развитой в Институте координации вычислительной техники, управляющей ходом периферий вычислительной машины P12. Эзактная модель неразрешима из-за большого числа возможных состояний системы. Но при помощи приближений удовлетворяющей точности, дискретных в статье, модель может уменьшаться до решаемого размера.