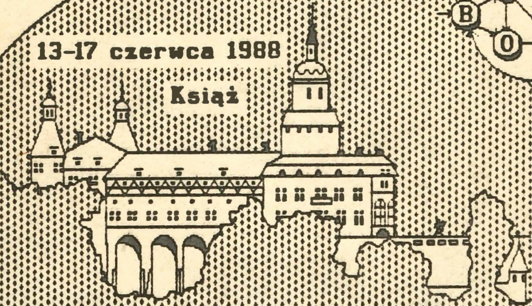
Redaktorzy:
A.Straszak
Z.Nahorski
J.Sikorski

13-17 czerwca 1988

Książ



1. Krajowa Konferencja Badań Operacyjnych i Systemowych

TOM 1

BOS'88

POLSKIE TOWARZYSTWO BADAŃ
OPERACYJNYCH I SYSTEMOWYCH

INSTYTUT BADAŃ SYSTEMOWYCH
POLSKIEJ AKADEMII NAUK

POLSKIE TOWARZYSTWO BADAŃ OPERACYJNYCH I SYSTEMOWYCH

TOM 1

OPTYMALIZACJA METODY I ZASTOSOWANIA

I KRAJOWA KONFERENCJA

BADAŃ OPERACYJNYCH i SYSTEMOWYCH

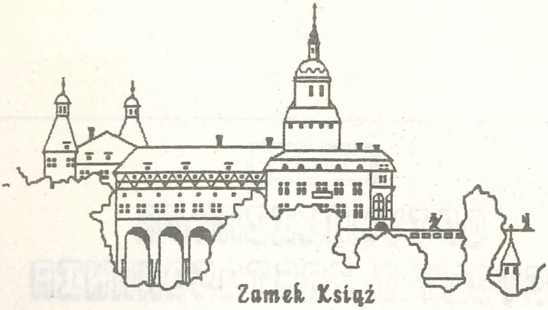
Książ, 13 - 17 czerwca 1988

BOS'88

INSTYTUT BADAŃ SYSTEMOWYCH POLSKIEJ AKADEMII NAUK

1989

WARSZAWA



Zamek Książ

I Krajowa Konferencja Badań Operacyjnych i Systemowych

Organizator konferencji

Polskie Towarzystwo Badań Operacyjnych i Systemowych
przy współpracy
Instytutu Badań Systemowych PAN

Komitet naukowy konferencji

Jerzy Hołubiec, Andrzej Kałuszko, Jerzy Kisielnicki, Henryk Kowalowski, Roman Kulikowski, Franciszek Marecki, Zbigniew Nahorski, Stanisław Piasecki, Jarosław Sikorski, Jan Stachowicz, Jan Stasierski, Andrzej Straszak, Maciej Sysło, Władysław Świtalski

Redaktorzy naukowi materiałów

Andrzej Straszak, Zbigniew Nahorski, Jarosław Sikorski

5. Optymalizacja struktur

5.1

I Krajowa Konferencja
Badań Operacyjnych i Systemowych
Książ, 13 - 17 czerwca 1988r.

ALLOCATION MODEL OF COMPUTATION PROCESSES IN A MULTIPROCESSOR SYSTEM

Konrad Wala, Jan Werewka
Institute of Automatics
Academy of Mining and Metallurgy
Al. Mickiewicza 30
30-059 Kraków

The computation time of algorithms (described by a network of processes) distributed in multiprocessor systems depends heavily on performed process allocation in the multiprocessor system. The static type of allocation is considered, i.e. the allocation is performed prior to the distributed computation beginning. An optimization model is formulated, which depends on last (computation time) distribution between the processors and on communication overhead caused by message transmission. The defined problem is NP-hard. Two relaxations of the model are formulated. For the second relaxed problem an algorithm is proposed, which is characterized by polynomial computation time. In this way a lower bound can be calculated for the estimation of heuristic solutions.

1. Introduction

A distributed computing system is a network of computing elements. Each element has its own storage which is not shared with other elements (processors). A distributed program consists of processes which cooperate to reach a common goal. The processes are allocated in different processors (are assigned to different processors). The processes cooperate by exchanging messages. When the processes are assigned to the same processor an information exchange is performed using the memory of the processor. In the case the processes are in different processors the messages are transmitted through the channels.

In the allocation of processes in a distributed computing system two contrary aspects should be considered. The processes should be allocated over different processors to enable a parallel computation. Allocation of the processes in different processors causes some communication overhead. However, the communication overhead caused by distributed computation is minimal when all the processes are allocated in one processor.

It is assumed that the algorithm to be distributed can be described by a random graph as described by Indurkhya and Stone (1986), where the processes and logical channels between processes (of the algorithm) corresponds with, respectively, nodes and edges of the graph. In the description by random graph the communication paths are known, but calculating some process may cause computation of randomly selected successor processes. Due to this randomness, the processes can not be scheduled for computation in advance. In the distributed computation three main factors which influence the computation time should be considered:

(1) The maximal computation time for each processor should be minimized over the whole processor set. The solution of this demand is a uniform last (i.e. computation time) distribution between the processors.

(2) Between processes assigned to different processors the messages are transmitted through the channels. The message transmission cause some computation delays.

(3) At time only one message can be transmitted trough a channel, so message queues may occur. The messages spend time in queues waiting for transmission. The waiting times of messages in the queues should be also minimized.

The three above factors are considered in the following two demands:

- The processes should be uniformly allocated over the processors to enable maximal parallelism of computation.

- The summarized communication overhead should be minimized.

In this paper it is assumed that allocation will be determined prior to the computation and that allocation will not change during the whole computation period.

The random graphs describe programs of a great practical importance. An example here may be the discrete event simulation, which can be considered as such a program. The program may be distributed, and then we speak about distributed discrete event simulation (see Werewka J. (1988)). Another example is the allocation of computational processes to solve problems of combinatoric optimization using branch and bound method described by Mc Cormack (1982) .

2. Model of the distributed program

Let us study the allocation problem using the following notations and definitions.

$\mathbb{Z} = \{z_1, z_2, \dots z_n\}$ set of processes; n - number of processes.

$\mathbb{P} = \{p_1, p_2, \dots p_m\}$ set of processors; m - number of processors.

$$d_{ij} = \begin{cases} 1 \text{ when a transmission path from the process } z_i \\ \quad \text{to the process } z_j \text{ exists } (i \neq j) \\ 0 \text{ in the other case. For } i,j \in \mathbb{N} = \{1,2,\dots n\}. \end{cases} \qquad (1)$$

Let a_{ij} be the average time required for message transmission between the processors p_i and p_j, for $i,j \in \mathbb{M} = \{1, 2, \dots m\}$.

Assumptions:

$0 \le a_{ij} < \infty$ - there exists an information connection between all the processors, for all $i, j \in \mathbb{M}$.

$a_{ii} = 0$ for all $i \in \mathbb{M}$.

The solution of the allocation problem is given by the sets $\mathcal{X}_1$, $\mathcal{X}_2$, ...$\mathcal{X}_n$, where $\mathcal{X}_j$, ($\mathcal{X}_j \subset \mathbb{Z}$), is a set of processes allocated in the processor p_j, for $j \in \mathbb{M}$.

Constraints:

$$\bigcup_{j \in \mathbb{M}} \mathcal{X}_j = \mathbb{Z}, \ \mathcal{X}_i \cap \mathcal{X}_j = \emptyset, \text{ for } i \neq j \text{ and } i,j \in \mathbb{M}, \qquad (2)$$

i.e. each process is allocated precisely in one processor.

There are two objective functions, which have to be taken into account. The function J describes the load of the processors:

$$J = \sum_{j \in M} \Big(\sum_{z_i \in X_j} t_i \Big)^2$$

Let us notice that by minimizing the objective function J for a homogeneous computer network, the solution of the allocation problem will give a minimal execution time. Here t_i is an execution time of the process z_i. In case when all the execution times have approximately the same value ($t_1 \approx t_2 \approx \dots\ t_n$) the objective function J will be equivalent to the function J_1.

$$J_1 = \sum_{j \in M} | X_j |^2$$

where $|X_j|$ is the number of processes allocated in the processor p_j. Hence in this paper we shall assume that the computation times of the processes are equal.

The function J_2 gives the total cost of message transmission between the allocated processes.

$$J_2 = \sum_{\substack{i,j \in M \\ i \neq j}} \sum_{z_k \in X_i} \sum_{z_l \in X_j} a_{ij} d_{kl} \tag{3}$$

3. Optimization problem

It is assumed that the main goal is the minimization of the objective function J_1. Let us notice that the function J_1 is minimal when the processes are uniformly allocated in the processors. Hence, we have arbitrarily assumed that

$$|X_j| = q \quad \text{for } j \in S \quad \text{and} \quad |X_j| = q-1 \text{ for } j \in M \setminus S \tag{4}$$

where $q = \lceil n/m \rceil$, $\lceil n/m \rceil$ is the least integer number greater or equal to $\frac{n}{m}$, $S \subset M$, $|S| = m_1$ and $m_1 = n - m(q-1)$.

In that way it will be assured that the objective function J_1 is minimal and the bicriterial problem will be changed to an easier one-criterial problem J_2.

Thus the allocation problem consists in the calculation of the sets $X_1, X_2, \dots, X_m$ subject to (2) and (4) which minimize the total cost J_2 of message transmission.

4. Models of relaxed allocation problem

A binary decision variable is introduced

$$x_{ik} = \begin{cases} 1, & \text{when process } z_i \text{ is assigned to processor } p_k \\ 0, & \text{in another case.} \end{cases}$$

The allocation model of computation processes in a multiprocessor system may be formulated as follows.

Minimize the objective function

$$J_2 = \sum_{i,j \in \mathbb{N}} \sum_{\substack{k,l \in \mathbb{M} \\ k \neq l}} a_{kl} d_{ij} x_{ik} x_{jl} \tag{5}$$

subject to constraints:

$$\sum_{k \in \mathbb{M}} x_{ik} = 1, \; i \in \mathbb{N}. \tag{6}$$

(Every process z_i is precisely assigned to one processor)

$$\sum_{i \in \mathbb{N}} x_{ik} = q, \; k \in \mathbb{S} \quad \text{and} \quad \sum_{i \in \mathbb{N}} x_{ik} = q - 1, \; k \in \mathbb{M} \setminus \mathbb{S} \tag{7}$$

(The processes are uniformly assigned to processors)

$$x_{ik} \in \{0,1\}, \; i \in \mathbb{N}, \; k \in \mathbb{M}. \tag{8}$$

Relaxing the model (5)-(8), we assume that $a_{kl} = a = \text{const} > 0$ $(k \neq l)$. In this way we assume that the average cost of information transmission does not depend on the processor position in a multiprocessor system. In this case the objective function is aJ_3, where:

$$J_3 = \sum_{i,j \in \mathbb{N}} \sum_{\substack{k,l \in \mathbb{M} \\ k \neq l}} d_{ij} x_{ik} x_{jl} \tag{9}$$

We introduce a new decision variable

$$y_{ij} = \begin{cases} 1, & \text{when processes } z_i \text{ and } z_j \text{ are assigned to the same processor} \\ 0, & \text{in other case.} \end{cases}$$

From the definition of the binary decision variable y_{ij} results her basic properties

$$y_{ii} = 1 \quad \text{and} \quad y_{ij} = y_{ji} \quad \text{for } i,j \in \mathbb{N} \tag{10}$$

Notice that according to (1), (6) and (10) there is:

$$\sum_{\substack{k,l \in \mathbb{M} \\ k \neq l}} d_{ij} x_{ik} x_{il} = d_{ij}(1 - y_{ij}) \tag{11}$$

According to (9) and (11) we obtain the objective function as

$$J_3 = \sum_{i,j\in N} d_{ij}(1-y_{ij}) = \sum_{i,j\in N} d_{ij} - \sum_{i,j\in N} d_{ij}\, y_{ij} \longrightarrow \min$$

Relaxed allocation problem is a maximization of the objective function

$$J_4 = \sum_{i,j\in N} d_{ij}\, y_{ij} \qquad (12)$$

subject to constraints

$$y_{ii} = 1,\ y_{ij} \in \{0,1\}:\ i,j \in N, \qquad (13)$$

$$y_{ij} = y_{ji}:\ i,j \in N, \qquad (14)$$

$$\sum_{j\in N} y_{ij} = q,\ i \in T \quad \text{and} \quad \sum_{j\in N} y_{ij} = q-1,\ i \in N\setminus T \qquad (15)$$

(to one processor there may be assigned q or q-1 processes)

$$y_{ij}\, y_{ik} \le y_{jk},\ i,j,k \in N \qquad (16)$$

where $T \subset N$; $|T| = m_1\, q$.

The constraints (16), valid for $q>2$, determine additionally the conditions of process assignment to one processor. When the processes z_i, z_j and z_i, z_k are assigned to one processor, then the processes z_j and z_k also are assigned to the same processor.

The equations (15) may be relaxed to the unequalities

$$\sum_{j\in N} y_{ij} \le q,\ i \in T \quad \text{and} \quad \sum_{j\in N} y_{ij} \le q-1 \quad i \in N\setminus T \qquad (17)$$

It may be noticed that this will not influence the optimal value of the objective function J_4, because $y_{ij} \ge 0$, $d_{ij} \ge 0$ and $J_4 \longrightarrow \max$. Thus we have got problem P1 and let

$$v(P1) = \max \{\, J_4:\ \text{subject to (13), (14), (16) and (17)}\}.$$

After the second relaxation, consisting in omitting the conditions (14) and (16), we get a problem described by (12), (13) and (17) called P2, where

$$v(P2) = \max\{\, J_4:\ \text{subject to (13) and (17)}\}.$$

This problem can be solved by a simple formula (18) of a computational complexity O(n):

$$v(P2) = \sum_{i\in N} \min\{d_i, q\} - \max\{0, |N\setminus T| - r\} \qquad (18)$$

where $r = |\{i \in N:\ d_i < q\}|$ is a number of processes z_i for which $d_i < q$, $d_i = \sum_{j\in N} d_{ij}$.

It is easy to notice, that the first part of formula (18) assures the maximum value of the function J_4 subject to constraints

$$\sum_{j\in\mathbb{N}} y_{ij} \le q, \; i\in \mathbb{N}. \qquad (19)$$

By calculating the second part of the formula the constraints (19) are replaced by constraints (17). The value of the objective function thus obtained may be used for determination of the lower bound of the optimal allocation.

From the property of relaxation we get the inequality $v(P1) \le v(P2)$, i.e. $V = v(P1)/v(P2) \le 1$. It is obvious that the value V depends on the structure of the algorithm graph $G = (\mathbb{Z}, \mathbb{U})$, where

$$\mathbb{U} = \{ (z_i, z_j) : d_{ij} = 1 \}$$

and (z_i, z_j) is the arc of the graph representing a logical channel between the processes z_i and z_j. The degree d_i of the z_i node of the graph is equal

$$d_i = \sum_{j\in\mathbb{N}} d_{ij} \qquad \text{for all } i.$$

In case of the graph of a simple structure one can calculate the value v(P1) and hence the exact value V. For instance (for $n > m$):

(i) for the complete graph ($d_i = n-1$ and $d_{ii} = 0$ for all i) we get V=1 because

$$v(P1) = v(P2) = q\,|\mathbb{T}| + (q-1)\,|\mathbb{N}\setminus\mathbb{T}| = n\,(2q-1) - q\,(q-1)\,m$$

(ii) for the nondirected star graph ($d_1 = n-1$ and $d_i = 1$ for $i \ne 1$) we have

$$v(P1) = 2\,(q-1), \; v(P2) = 2\,(q-1) + n - q = n + q - 2$$

and hence

$$V = \frac{2q-2}{q+n-2} = \frac{2\lceil n/m\rceil - 2}{\lceil n/m\rceil + n - 2}.$$

(iii) for the directed star graph we have

$$v(P1) = V(P2) = q - 1 \text{ and } V = 1$$

or

$$v(P1) = q - 1 \text{ and } v(P2) = n - 1$$

(subject to direction of the arcs)

(iv) for the nondirected chain ($d_1 = d_n = 1$ and $d_i = 2$ for $i \ne 1, n$) we get

$$v(P1) = 2\,(q-1)\,(n-m), \; v(P2) = 2\,(n-1)$$

and for the directed chain (path)

$$v(P1) = (q-1)\,(n-m), \; v(P2) = (n-1)$$

5. Final remarks

In Wala and Werewka (1989) approximate algorithms for obtaining the solution of the problem (2)-(4) are proposed. The designed algorithm for solving the relaxed problem enables to find the lower bound of the optimal allocation given by the formula

$$a J_3 = a \left(\sum_{i,j \in N} d_{ij} - v(P2) \right), \text{ where } a = \min_{i,j} a_{ij} \ (i \neq j).$$

References

1. Mc Cormack W. M. et al. (1982) Multi-computer parallel architectures for solving combinatorial problems. Multicomputers and Image Processing Algorithms and Programs. AP, New York, London.
2. Indurkhya B., Stone H. S. (1986) Optimal Partitioning of Randomly Generated Distributed Programs. IEEE Trans. on Soft. Eng. Vol. SE-12, No 3, pp 483-495.
3. Wala K., Werewka J. (1989) Allocation of computational processes in Multiprocessor System, Archiwum Automatyki i Telemechaniki (to be printed).
4. Werewka J. (1988) Verteilte Simulation der Kommunikationssysteme.; Dr. Sc. Diss. Humboldt-Universitaet zu Berlin.

L I T E R A T U R A

(1) Crabill T. B., Gross D., Magazine M. J. (1977) A Classified Bibliography of Research on Optimal Design and Control of Queues. Operations Res. Vol. 25, No. 2, ss. 219–233.

(2) Deb R. K., Serfozo R. F. (1973) Optimal Control of Batch Service Queues. Advances Appl. Probability 5, 340–361.

(3) Kosten L. (1973) Stochastic Theory of a Service System. Pergamon Press, New York 1973.

(4) Weiss H. J. (1979) The Computation of Optimal Control Limits for a Queue with Batch Services, Manag. Sci. 25 ss. 320–328.

(5) Stanek S. (1984) Sterowanie [illegible], Praca doktorska, [illegible]

(6) Королюк В., Турбин А. (1976) Полумарковские процессы и их приложения, Киев.

(7) [illegible] (1982) [illegible] zakończeniem [illegible] statystycznego, Przegląd Stat. XXIX, 3/4.

(8) Neuts M. (1967) A general class of bulk queues with Poisson input, Ann. Math. Stat. 38, ss. 759–770.

(9) ______ (1979) Queues solvable without Rouché's theorem. Operations Res. 27.

(10) Keilson J. (1966) The ergodic queue length distributions for queueing systems with finite capacity. J. Roy. Stat. Soc. B28.

Zarząd

Polskiego Towarzystwa Badań Operacyjnych i Systemowych

Prezes

prof.dr hab.inż. Andrzej Straszak
Instytut Badań Systemowych PAN

Wiceprezes

prof.dr hab.inż. Jan Stasierski
Wojskowa Akademia Techniczna

Wiceprezes

prof.dr hab.inż. Stanisław Piasecki
Instytut Badań Systemowych PAN

Sekretarz generalny

dr inż. Zbigniew Nahorski
Instytut Badań Systemowych PAN

Sekretarz

dr inż. Jarosław Sikorski
Instytut Badań Systemowych PAN

Skarbnik

dr inż. Andrzej Kałuszko
Instytut Badań Systemowych PAN

Członkowie

prof.dr hab. Jerzy Kisielnicki
Wydział Zarządzania UW

doc.dr hab.inż. Jan Stachowicz
Zakład Nauk Zarządzania PAN

doc.dr hab.inż. Bohdan Korzan
Wojskowa Akademia Techniczna

doc.dr hab.inż. Maciej Sysło
Instytut Informatyki UWr.

Komisja rewizyjna

PRZEWODNICZĄCY

dr Władysław Świtalski
Katedra Cybernetyki i Badań Operacyjnych UW

CZŁONKOWIE

dr inż. Janusz Kacprzyk
Instytut Badań Systemowych PAN

doc.dr hab.Henryk Sroka
Akademia Ekonomiczna w Katowicach

dr inż. Marek Malarski
Instytut Transportu PW

dr inż. Leon Słomiński
Instytut Badań Systemowych PAN

PION III