V.M. Dubovoi, M.E. Duda

COORDINATION IN SERIAL-PARALLEL IMAGE PROCESSING

Vinnytsia National Technical University,
95 Khmelnytske shosse, Vinnytsia, 21021, Ukraine;

Abstract: Serial-parallel systems used to convert the image. The control of their work results with the need to solve coordination problem. The paper summarizes the model of coordination of resource allocation in relation to the task of synchronizing parallel processes; the genetic algorithm of coordination developed, its adequacy verified in relation to the process of parallel image processing.
Keywords: coordination processes, coordination model, resource allocation, synchronization task, image processing.

INTRODUCTION

Images are one of the main ways of presenting and transmitting information. Saving and sending, and then receiving and restoring graphical information is associated with the need to perform large amount of conversions depending on image size and quality [1-2]. Increasing the speed of these conversions in real-time systems is an important scientific and practical problem.

ACTUALITY

Serial-parallel systems used to convert the image. Management of their work makes it necessary to solve the problem of coordination [3]. Methods of coordination in relation to control over technological processes discussed in papers [4-6]. However, the task of resource allocation in relation to the problem of parallel processes synchronization is not considered. The analog of image processing resource is an array of image data, which causes coordination problems relevance in relation to the problem of increasing speed of the image conversion in real-time systems.

Non-iterative, iterative and genetic algorithms used to solve the problem of coordination [7]. In the non-iterative algorithm the optimum result is achieved by a single exchange between levels. The disadvantage is the high complexity of the calculations. In iterative algorithm, an optimal solution obtained by repeated information exchange between the center and the elements. The disadvantage is large iterations number, and as a result - high computational time. Genetic algorithm is search algorithm used for solving optimization and simulation task by sequentially selecting, combining and varying desired parameters. Most important thing when coordinating using genetic algorithm is choice of fitness function [7].

AIM OF THE RESEARCH

Let consider the generalized problem of coordination processes for G successive steps of converting the input data (Fig. 1). Let call parallel-serial unit of operations to be “step of conversion”.

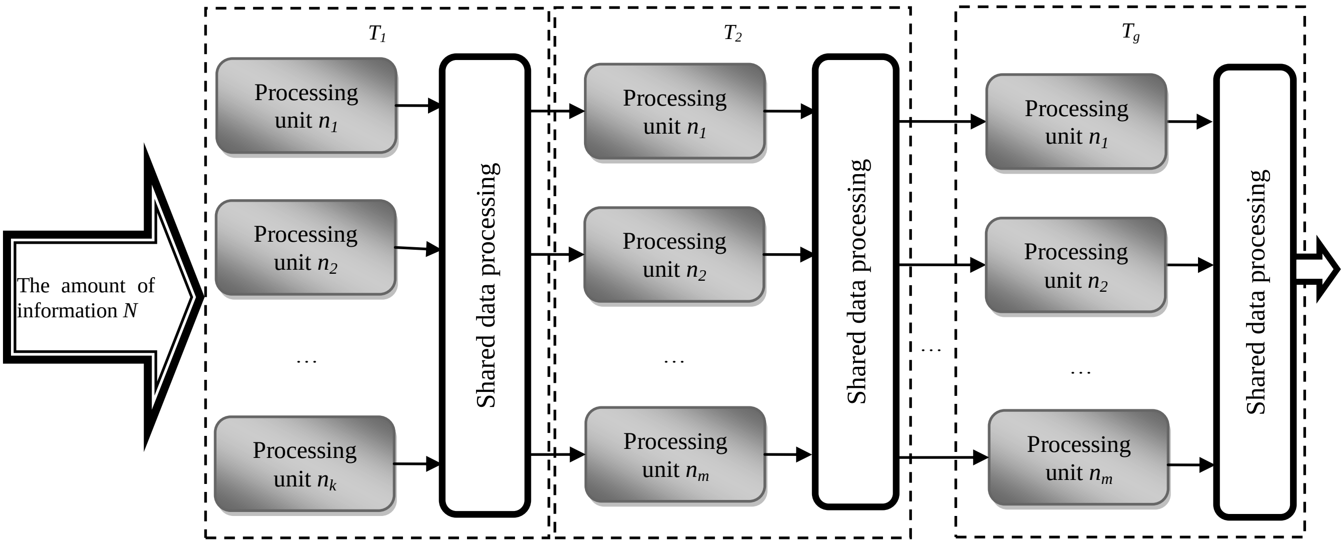


Figure1 – General structure of image processing

The picture has a certain initial amount of information N, which is divided into k clusters. Each cluster of information n_i, $i = \overrightarrow{1..k}$, enters the information processing unit with performance p_i. Information after processing enters the common processing unit in a sequence and is processed with performance $u_i = u$. After serial processing information is supplied to the next step with new number of clusters m.

The aim of coordination process is to reduce total time of the system work T.

SOLUTION

Let consider the example of a Gantt chart for the process that consists of two steps. The first step has two processing units and a general block. The second step consists of four processing units and a general block (Fig. 2).

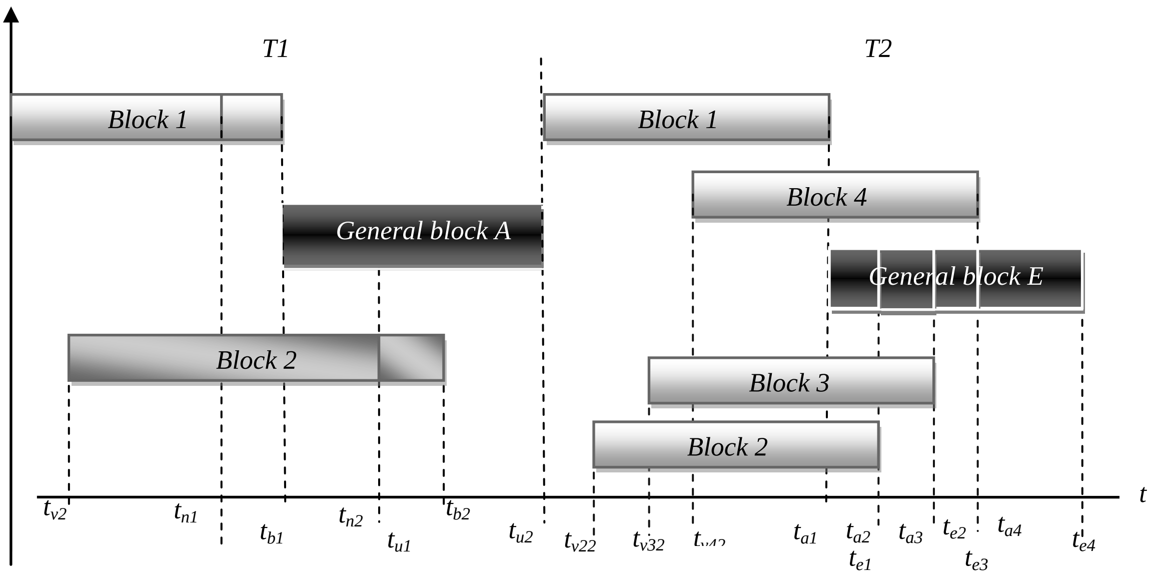


Figure 2 – Gantt Chart for parallel-serial processing

The time of system work will be sum of execution time of sequential steps:

$$T = \sum_{g=1}^{G} \mathrm{T_g} \tag{1}$$

Let consider the first step.

Since the procedure of elements admission to the block of joint data processing is defined, it could be said that the first block processes the information earlier, and the last - later. General operation starts after completion of the first operation. Then

$$T_1 = t_{\mathrm{n}_1} + t_{\mathrm{u}} \tag{2}$$

where t_{u}– action time of general operation processing, t_{n_1} – action time of first processing unit.

Action time of each parallel block depends on the processing performance p_i and the amount of information n_i, which enters the corresponding block:

$$t_{n_i} = \frac{n_i}{p_{n_i}}, \tag{4}$$

It should be noted that the total amount of information like the overall performance satisfies the following condition:

$$\mathrm{N} = \sum_{i=1}^{\mathrm{K}} n_i, \tag{5}$$

$$\mathrm{P} = \sum_{i=1}^{\mathrm{K}} p_{n_i}, \tag{6}$$

The total amount of information N is taken through general processing unit. Then, action time of general block:

$$t_u = \frac{\mathrm{N}}{\mathrm{u}} = \sum_{i=1}^{\mathrm{K}} \frac{n_i}{\mathrm{u}}, \tag{7}$$

where u - the performance of the general unit.

T_1 is computed the following way:

$$T_1 = \frac{n_1}{p_{n_1}} + \frac{\mathrm{N}}{\mathrm{u}}, \tag{8}$$

Each step is similar to the first one, but has different number of blocks and parameters:

$$T_g = \left(\frac{n_1}{p_{n_1}} + \frac{\mathrm{N}}{\mathrm{u}}\right) \tag{9}$$

Then the total time T:

$$T = \sum_{g=1}^{G} \left(\frac{n_1}{p_{n_1}} + \frac{\mathrm{N}}{\mathrm{u}}\right)_{\mathrm{g}}. \tag{10}$$

In the Gantt chart variable t_{v_i} appears – the downtime of the unit. Since information processing begins after the previous processing, the downtime

$$t_{v_i} = t_{n_1} + \sum_{\mathrm{j}=1}^{\mathrm{i}} t_{u_j} - t_{n_i} \tag{11}$$

where t_{u_i} - processing time of information n_i in the general processing unit:

$$t_{u_i} = \frac{n_i}{u}$$

We have the following model:

$$\begin{cases} t_{v_i} = t_{n_1} + \sum_{j=1}^{i} t_{u_j} - t_{n_i} \\ T_g = \left(\frac{n_1}{p_{n_1}} + \frac{N}{u}\right) \\ N = \sum_{i=1}^{K} n_i \, , \\ P = \sum_{i=1}^{K} p_{n_i} \, , \end{cases} , \qquad (12)$$

To test the adequacy of the model let construct dependencies $t_{v_i} = f(n_i)$, $t_{v_i} = f(p_i)$, $t_{v_i} = f(u)$ for the first step of Gantt chart (Fig. 2). Variables could vary in the range from 0 to 100.

Downtime could be calculated through block parameters:

$$t_{v_i} = \frac{n_1}{p_{n_1}} + \sum_{j=1}^{i} \left(\frac{n_i}{u}\right)_j - \frac{n_i}{p_{n_i}}. \qquad (13)$$

Then for the first and second block, respectively:

$$t_{v_1} = \frac{n_1}{p_{n_1}} + \sum_{j=1}^{1} \left(\frac{n_i}{u}\right)_j - \frac{n_1}{p_{n_1}} = 0$$

$$t_{v_2} = \frac{n_1}{p_{n_1}} + \frac{n_1}{u} - \frac{n_2}{p_{n_2}}$$

Using (5) and (6):

$$t_{v_2} = \frac{n_1}{p_{n_1}} + \frac{n_1}{u} - \frac{N - n_1}{P - p_{n_1}}. \qquad (14)$$

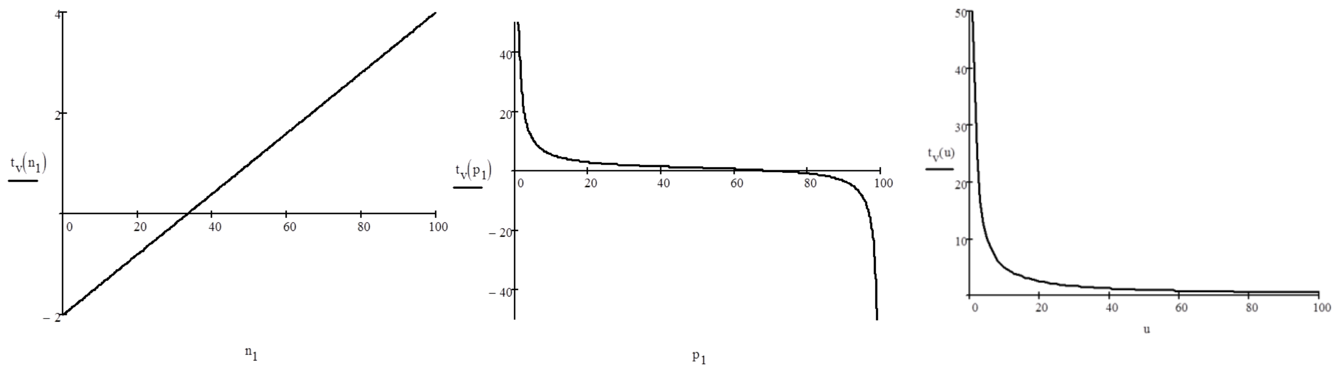
Chart on fig.3a shows directly proportional dependency. Negative downtime values indicate how much earlier second unit needs to start work comparing to the first unit. It is clear that the smaller the amount of information in the first block, the more it is in the second one, and the earlier one needs to start working before the second block.

Chart on fig.3b shows the dependency between downtime and performance of the first block. Obviously, the more the performance of the first block, the lower the performance of the second one is, thus it is more time needed to process; one needs to start working earlier to be in time by general operation ready.

Chart on fig.3c show inverse proportional dependency between downtime and general unit performance: the higher the performance, the less is downtime.

Thus, model analysis confirms its adequacy.

The model represents a nonlinear dependency of the transformation action time with the partition of input data into blocks (5) and computing resources into parallel channels (6). In this regard, genetic algorithm chosen to solve set optimization task.



a) b) c)

Figure 3 – Dependence of downtime t_{v_2} from: a) the amount of information in the first block n_1, with parameters $N = 100, P = 100, p_1 = 50, u = 50$. b) performance of the first unit p_1, with parameters $N = 100,\ P = 100, n_1 = 50, p_1 = 50$. c) performance of the general block u, with parameters $N = 100, P = 100, n_1 = 50, u = 50$

CONCLUSIONS

Generalized model of coordination of concurrent processes developed and investigated; its application to image processing considered. The model allows evaluating the execution time of data conversion in generalized series-parallel system. The result is a target function to optimize system performance.

REFERENCES

[1].Jähne B. Digital Image Processing / Bernd Jähne - Springer-Verlag Berlin Heidelberg, 2005 - № 6 – 608 c - URL: https://books.google.com.ua/books?hl=ru&lr=&id=_SPyCAAAQBAJ&oi=fnd&pg=PA3&ots=N23RatEIfS&sig=yIELDEr8kBBOF11Q_6CxZHQcBNc&redir_esc=y#v=onepage&q&f=false

[2].Urschler M. Segmentation Tool Features / Martin Urschler – URL: http://www.icg.tugraz.at/Members/urschler/master_thesis_features.htm

[3].Woodside C. M. Fast Allocation of Processes in Distributed and Parallel Systems / C. M. Woodside, and G. G. Monforton // IEEE Transactions on parallel and distributed systems . – 1993. – P. 164-174.

[4].Katrenko A.V. Mechanisms of coordination in complex hierarchical systems. / A.V. Katrenko I.V. Savka - 2008 - URL: http://vlp.com.ua/files/16_1.pdf

[5]. Ladanyuk A.P. System task coordination continuous technological complexes type / A.P. Ladanyuk, D.A. Shumyhay, R.O. Boyko - URL:http://dspace.nuft.edu.ua/jspui/bitstream/123456789/4444/1/Sh_3.pdf.

[6].Mesarovic, M. D. Theory of Hierarchical Multilevel Systems / M. D. Mesarovic, D. Macko, Y. Takahara,. — Published by Academic Press, Inc., 1972. – 344 p.

[7].Bayas M. M. Efficient Resources Allocation in Technological Processes Using Genetic Algorithm / M. M. Bayas, V.M. Dubovoy // Middle-East Journal of Scientific Research. – 2013. – Vol. 14, № 1. – P. 1-4. – DOI: 10.5829/idosi.mejsr.2013.14.1.16313.