Algorithm for Structural Analysis of the Measurement Path of Computer System Performance Parameters [*]

Ya. A. Kogan, A. A. Korshunov, and I. B. Muchnik

UDC 681.322.05

Algorithms are discussed for structural analysis of the path of measurement of computing system performance parameters: an algorithm for forming a dictionary of reference fragments and an algorithm for simultaneous dictionary identification and structural sequence decomposition.

1. Two Forms of Describing the Measurement Path Structure

An indispensable part of experimental analysis and subsequent optimization of computing system is the analysis of the structure of actual measurement path recorded in the course of computing process with the purpose of constructing their satisfactory models. Such paths consist of a sequence of accesses to the elements of some fixed finite set, for example, to program pages or modules, or to the physical addresses of data blocks stored in external memories (magnetic disks, drums, and tapes). A general approach to the identification of the structural path organization, which reflects its basic properties, has been proposed in [1] and consists in the following.

First, the original path is replaced by a structural sequence. For this purpose the fixed set of elements, mentioned above, is decomposed into a small number of standards, in a given sense, classes. The need of such a decomposition is due to the relatively high power of the original set (several thousand). Transformation of the original elements into indexes indicating their belonging to different classes produces a path notation consisting of long sequences of identical indexes. The latter is more conveniently represented by a short sequence of unrepeating class indexes. By its construction, such a sequence reflects all main changes taking place in the course of the computing system operation. Hence it inherits its name – structural sequence [2]. Such a structural sequence is several hundreds symbols long and can be analyzed in detail.

[*] Moscow. Translated from Avtomatica i Telemekhanika, No. 3, pp. 161 – 167, March, 1984. Original article submitted April 5, 1982.

0005 – 1179/84/4503 – 0411 $08.50 © 1984 Plenum Publishing Corporation.

Identification of structural path organization begins after its structural sequence has been obtained. This can be represented in two forms.

The first [3] assumes that a certain collection (vocabulary) of so-called reference fragments, consisting of chains of class indexes, has been a priori compiled. These chains hold a priori information on most frequent transitions between element classes. In particular, if the path elements are the addresses of blocks of data stored on magnetic disks, the vocabulary of such chains explicitly defines the most frequent switchings of write/read heads between cylinders.

With the aid of the vocabulary the path is decomposed into fragments as closely as possible approximating in the sense of a definite metrics the reference fragments, and then a reference fragment corresponding to it replaces each individual fragment. As a final result, its model consisting of a sequence of words of a given vocabulary represents the path. In addition, a criterion is computed, which is an indication of the adequacy of the constructed model.

If the degree of adequacy is found to be insufficient, a new vocabulary must be compiled and used to construct a more adequate model.

In the second version the decomposition of structural sequence into fragments is assumed to be known a priori. Obviously, if the a priori decomposition is to hold essential information on the structural path organization, the decomposition must reflect drastic changes in the organization of the structural sequence. Such a decomposition can be easily constructed by a single inspection of the sequence with the aid of a variable-size window, which isolates the current fragment and evaluates the degree of its similarity to the preceding fragment in the sense of the given metrics. As soon as the decomposition of the structural sequence has been determined, one can apply some automatic-classification algorithm [4] to the set of obtained fragments. After this algorithm is applied, the set of fragments constituting the path being analyzed is divided into a small number of subsets, the elements of each of them being similar to each other in the sense of accepted metrics. Each of these subsets can then be associated with a single fragment, which on the average is most closely related to all the fragments of the given subset. This fragment can be found by applying prototype constructing algorithm [2].

If now each fragment of the structural sequence is replaced by its corresponding prototype, we obtain a sequence, which, of all sequences that can be constructed from the obtained prototypes, most closely approximates (in the sense of the accepted metrics) the given one. This is due to the fact that if the automatic-classification algorithm of [4] is applied, approximation (in the sense of the given metrics) of each fragment of the structural-sequence decomposition by its corresponding prototype gives better results than approximation by some otherwise constructed prototype. Thus, these prototype fragments can be naturally accepted as reference fragments and their set, as the basic vocabulary.

The above two versions of analysis describe two sides of the structural sequence: In one case we seek a decomposition corresponding to a given vocabulary, whereas in the other, we seek a vocabulary satisfying a given decomposition. This, the second version can be treated as a stage in reorganizing the reference fragment vocabulary when the degree of model adequacy is unsatisfactory; both versions of analysis can then be combined into single algorithm for simultaneously finding both the structural sequence decomposition and its corresponding vocabulary. Such an algorithm is more simply constructed as a succession of alternate application of both procedures. The combined algorithm is described in Section 3.

The efficiency of this algorithm stems from the elimination of the difficult procedure classifying the set of fragments when compiling the reference fragment vocabulary. For this purpose, the article proposes to formulate the vocabulary as a solution of an optimization problem, which has an exact effective solution. An algorithm providing a solution of this problem and based on the method of monotonic functions [5,6] is described in the next section.

2. Algorithm for Compiling the Reference Fragment Vocabulary

To formulate the problem of compiling a reference fragment vocabulary and describe its solution algorithm it is necessary to introduce a metrics between two arbitrary chains of indexes as in [3].

For this purpose we use two types of elementary transformations by means of which an arbitrary chain T_2 can be transformed into any other chain T_1.

1. $Ins(i,t_q)$, insertion of an index $t_q \in T_2$ between the i-th and $(i+1)$-th indexes of chain T_1.
2. $Str(i)$, striking out the index at the position i in the chain T_2 from this chain.

These two transformations define the set of transformations of chain T_2 into T_1. The length $\lambda(T_1,T_2)$ of the succession of (the number of applied) elementary Ins and Str transformations measures the transformation complexity.

If we now select a transformation T_2 into T_1 such that the length

$$\rho(T_1,T_2) = min\{\lambda(T_1,T_2)\} \tag{1}$$

is minimum, $\rho(T_1,T_2)$ will define the distance between the two chains. Algorithm [2] computes this quantity with the aid of a procedure for finding a minimum-length path between two isolated nodes x_s and x_f on a special oriented planar graph $G(T_1,T_2)$ whose structure is shown in Fig. 1. Horizontal and vertical motion along this graph corresponds to transformations of the type Ins and Str respectively, and motion along diagonals corresponds to trivial transformations $(t_q \in T_2) \to (t_p \in T_1)$ that leave the indexes unchanged. Accordingly, horizontal and vertical edges are given the weight 1 and diagonal, the weight 0.

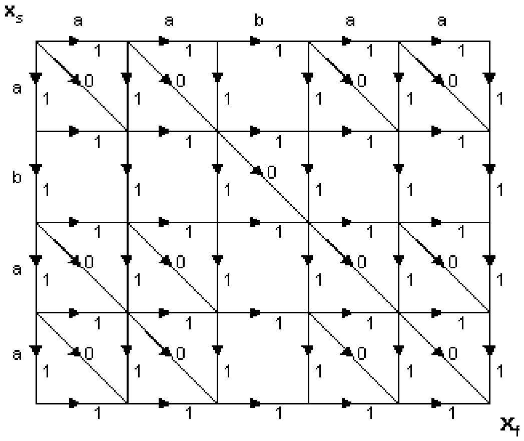

Fig. 1.
The graph $G(T_1,T_2)$ for the transformation of chain $\{a,a,b,a,a\}$ into the chain $\{a,b,a,a\}$.

Most suitable for the construction of a single-step procedure for compiling a reference fragment vocabulary is the method of finding in the complete weighted graph a subset of so-called “maximally distant” nodes [6], which consists in the following.

Let W be a set of nodes on which a complete, weighted graph is defined with a symmetric weight matrix $\left\| \rho_{i\,j} \right\|$ and let X be the set of all nonempty subsets. Let us associate with each subset $H \in Z$ a number $F(H)$ in accordance with the following rule:

$$F(H) = \min_{i \in H} \pi(i,H),\ \pi(i,H) = \sum_{j \in H} \rho_{i\,j}\ . \tag{2}$$

This defines a scalar function on X. The problem of finding a subset H^* of “maximally distant” nodes has been formulated in [5] as a problem of finding the global maximum of the function $F(H)$:

$$F(H^*) = \max_{H \in X} F(H)\,. \tag{3}$$

Since finding $F(H)$ for each H means, as indicated by (2), a search for some “central” element, the solution of (3) is a search for a subset of “maximally spaced centers” or, in other words, prototypes for the corresponding subsets.

Treating the isolated elements as prototypes of their class makes it possible to consider problem (3) as a modification of the well-known problem of automatic classification. In fact, let H^* be a set of prototypes for the respective subsets of nodes of the original set W. Then the non-selected nodes can be classified by associating them with their nearest prototypes; the rule for associating an arbitrary node $\alpha \in W$ with the subset $W_S \subset W$ is given by the relation

$$\rho_{\alpha\,S} = \min_{i \in H^*} \rho_{\alpha\,i}\,. \tag{4}$$

The problem of finding the global maximum of the function $F(H)$ makes it thus possible to devise a single-step procedure for compiling the reference fragment vocabulary.

Consider a certain decomposition DL of the structural sequence L into segments: $DL = \{L_1,...,L_d\}$; let us compute in accordance with (1) the matrix $\left\| \rho(L_i, L_j) \right\|$ of pairwise distances between them. Then, the subset of segments $L^* \subseteq DL$, which is the solution of problem (3) obtained for the matrix $\left\| \rho(L_i, L_j) \right\|$, can be taken as the vocabulary Y of reference fragments most consistent with the given decomposition.

From the definition of pairwise distances between segments (1) we have

$$\rho(L_i, L_j) \geq 0,\ i \neq j,\ i, j = \overline{1,d},\ \rho(L_i, L_i) = 0.$$

Hence follows immediately that the system of numbers

$$\pi(L_i, H) = \sum_{L_j \in H} \rho(L_i, L_j),$$

called the system of weights of segments $L_i \in H$ on H, satisfies the condition

$$\pi(L_i, H \setminus L_j) \leq \pi(L_i, H)\ \forall L_i \in H \setminus H_j\ \left(i \neq j\right). \tag{5}$$

Such systems are called monotonic in [5].

Following the property of monotonicity of (5), Kuznetsov [6] [1] has proposed an algorithm for an exact solution of the problem (3). The algorithm is as follows.

1. On the set $H_1 = DL$ is isolated a segment $L_{i_1} \in H_1$ such that

$$\pi(L_{i_1}, H_1) = \min_{L_i \in H_1} \pi(L_i, H_1).$$

Its weight $\pi(L_{i_1}, H_1)$ is denoted by u_1. [2]

2. The segment L_{i_1} is deleted from the set H_1; step 1 of the algorithm is executed on the new sequence $H_2 = H_1 \setminus L_{i_1}$ and the weight of the segment L_{i_2} is compared with the threshold u_1. If

$$\pi(L_2, H_2) \leq u_1,$$

the threshold u_1 is preserved:

$$u_2 = u_1.$$

Otherwise, the former threshold is replaced with the new value:

$$u_2 = \pi(L_{i_2}, H_2).$$

[1] The algorithm, which solves the problem may be found in the second part of [5], see http://www.datalaundering.com/download/extrem02.pdf , as well as in the seminal paper http://www.datalaundering.com/download/modular.pdf , notice made by JM.

[2] The number u_1 will be used as the initial threshold for comparing with weights of other segments considered at the next step on the set $H_1 \setminus L_{i_1}$. The values of the threshold can vary.

The algorithm end when the original set DL is completely exhausted, i.e., all its segments have been arranged into a sequence $\left\langle L_{i_1}, L_{i_2}, ..., L_{i_d} \right\rangle$, called the defining in [5]. At the same time, the accompanying sequence of subsets

$$\overline{H} = \left\langle H_1, H_2, ..., H_d \right\rangle$$

is also obtained, where $H_1 = DL$, $H_{k+1} = H_k \setminus L_{i_k}$, and L_{i_k} is the k-th element of the defining sequence.

From the defining sequence is isolated a special subsequence $\left\langle L_{j_1}, L_{j_2}, ..., L_{j_p} \right\rangle$ whose segments determine the steps of the algorithm in which the comparison threshold vary. The function $F(H)$ then reaches an absolute maximum on the subset H^* of segments included, together with the segment L_{j_y}, into the defining sequence after the last change of the threshold value at the j_p-th step [5]. Hence, H^* is the solution of the problem of isolating “maximally spaced centers.”

Note that unlike in the traditional treatment of class prototypes as segments on the average most similar to all segments of the respective classes, i.e., as centers, here we use as class prototypes segments that can be distant from the centers of these classes. At the same time, if the initial set the segment clusters are “compact” enough, both methods isolate these clusters in the same way. For this it is essential for each of the segments clusters of the original set to be sufficiently distant from all others. Transforming the original weight matrix $\left\| \rho_{i\,j} \right\|$ into the matrix $\left\| \rho_{i\,j}^{p} \right\|$ whose all elements have been raised to a sufficiently high power $p > 0$ can effect the latter.

3. Simultaneous Identification of Vocabulary and Decomposition of Structural Sequence

To describe the algorithm of simultaneous vocabulary identification of structural sequence decomposition not, first of all, that the problem of constructing an approximating model of the chain T in a given vocabulary $Y = \left\{ y_k, k = \overline{1,m} \right\}$ has been formulated in [3] as a problem of finding its decomposition D^*T for which the functional

$$J(DT) = \sum_{i=1}^{k} \min_{1 \le k \le m} \rho(T_i, y_k),\ T_i \in DT \qquad (6)$$

is minimum on the set $\Sigma(T)$ of its all possible decompositions:

$$J(D^*T) = \min_{DT \in \Sigma(T)} J(DT). \tag{7}$$

The approximating model is constructed using the relations:

$$R(T) = \{y_{k_1}, \ldots, y_{k_{d^{\mathrm{opt}}}}\}, \tag{8}$$

where k_i are found from the conditions

$$\rho(T_i, y_{k_i}) = \min_{1 \le k \le m} \{\rho(T_i, y_k)\}.$$

Here d^{opt} denotes the number of segments in the decomposition D^*T.

We shall say that model $R(T)$ is adequate if the functional $J(D^*T)$ does not exceed a certain number a which, generally speaking, depends on the length of the chain T, i.e.,

$$J(D^*T) \le a(\mathrm{L}(T)). \tag{9}$$

If the relation (9) is not satisfied one has to compile a new vocabulary and use it to construct another model. This can be done with the aid of the algorithm for compiling a vocabulary of reference segments described in Section 2.

Assume that the initial reference segment vocabulary Y has been somehow chosen and the decomposition D^*T of the structural sequence L making the functional (6) minimal on the set of all its possible decompositions has been found. The model $R(L)$ in the given vocabulary Y is thereby constructed. On the set of all segments $L_i \in D^*L$ we find a subset of segments, $L^* \subseteq D^*L$ which is a solution of the problem (3). Let us take this subset as the new vocabulary Y_n. For the sequence L we find a new decomposition D_n^*L in the new vocabulary Y_n that makes (6) minimal. Then we construct a new model $R_n(L) = \left\{y_{k_1}^n, \ldots, y_{k_{d_n^{\mathrm{opt}}}}^{\mathrm{n}}\right\}$, where $y_{k_i}^n$ is found in the same way as y_{k_i} [see (8)], etc.

Obviously, as a condition of this algorithm to stop one can take a situation in which the vocabulary cannot be modified for a given decomposition and the decomposition cannot be changed for given vocabulary. If after the algorithm stops, the model $R_n(L)$ is found to be inadequate, i.e.,

$$J(D_n^*L) > a(\mathrm{L}(L)),$$

the algorithm must run again with a new starting vocabulary Y.

The efficiency of such a composite algorithm essentially depends on the efficiency of its constituents. In fact, the algorithm solving (3) consists in successively calculating the values of functions $\pi(L_i, H_k)$ $\forall L_i \in H_k$ for known values of the functions $\pi(L_i, H_{k-1})$ using expression

$$\pi(L_i, H_k) = \pi(L_i, H_{k-1}) - \rho(L_i, L_{i_{k-1}}).$$

Thus, the algorithm operation is mostly associated with computing the distance matrix $\left\|\rho(L_i, L_j)\right\|$, which is proportional to $d^2 \cdot t_{\text{av}}$. The number d being the number of chains into which the sequence L is decomposed, and t_{av} is average time needed to compute the distance between two arbitrary chains not more than l indexes long, where l is twice the length of the maximum words in the vocabulary Y. The algorithm of finding a decomposition of a sequence requires a machine time of the order $\mathrm{T} = m \cdot l \cdot N \cdot t_{\text{av}}$ seconds [3], where N is the length of the structural sequence L, and m is the number of words in vocabulary Y. The algorithm is seen to be suitable for the analysis of sequences several hundreds symbols long and this agrees with the need of practical problems of path analysis.

4. Conclusion

The paper treats the problem of compiling a reference fragment vocabulary as a problem of finding the absolute maximum of the function $F(H)$ on the set of all subsets of the starting set of segments $L_i \in D^*L$ of the structural sequence L. To solve the latter problem, a single-step procedure is proposed for an exact solution based on the application of the method of monotonic functions. The described algorithm for setting up a vocabulary of reference fragments is simple and efficient from the point of view of the amount of necessary computations. Combining this algorithm with the algorithm for constructing a model approximating the analyzed measurements path in a given vocabulary [3], makes it possible to construct a composite algorithm, which simultaneously identifies the vocabulary and decomposes the structural sequence. The composite algorithm is constructed as a chain of successive applications of its component algorithms.

Literature cited

1. Ya. A. Kogan, A. A. Korshunov, and I. B. Muchnik, “Automatic analysis of the measurement path of dynamic operation of computing systems,” in: Materials of the Seminar on the Development and Use of Data Processing Systems in the Unified Computer System, *Znanie, MDNTP*, Moscow (1980), pp. 136-140.
2. V. V. Mottl’ and I. B. Muchnik, “Lingustic analysis of experimental curves,” *TПÉR*, *67*, No. 5, 12, (1979).
3. Ya. A. Kogan, A. A. Korshunov, and I. B. Muchnik, “Algorithm for constructing a model approximating the event path describing the operation of computing systems, “*Avtomat. Telemekh*., No. 4, 160 (1980).
4. A. A. Dorofeyuk, “An automatic classification algorithms,” (Review), *Avtomat. Telemekh*., No. 12, 78 (1971).
5. J. E. Mullat, “Extremal subsystems of a monotonic systems. I,” *Avtomat. Telemekh*., No. 5, 130 (1976), http://www.datalaundering.com/download/extrem01.pdf .
6. E. N. Kuznetsov, “Analysis of a relation matrix by constructing on it a monotonic system,” *Avtomat. Telemekh.*, No. 7, 128 (1980), http://www.datalaundering.com/download/couplem.pdf .