TEM Journal. Volume 8, Issue 1, Pages 144-149, ISSN 2217-8309, DOI: 10.18421/TEM81-20, February 2019.

Synthesis Method of Flash Graphs for Optimal Search of Manufacturing Subsystems Recognised on Basis of Composite Bayes Task Solution

V.N. Shepel`, N.V. Speshilova, V.A. Tripkosh, R.R. Rakhmatullin

Orenburg State University, Prospekt Pobedy, d. 13, 460018, Orenburg, Russia

Abstract **– The decision-making algorithm selection is one of the problems that modern economics, industrial management and situation identification process face. The Bayesian composite approach is considered to be the safest way for algorithm identification. However, the approach based on Bayesian composite rules is time-consuming and computationally intensive, that`s why it is important to develop the method, when combined with the Bayesian approach, will enable to scale back the computation.**

The article describes the synthesis method of flow graphs for optimal search within the member group (including the search within the industrial structure) and suggests this method as the main implementable algorithm for situation identification, based on Bayesian composite rule for decision-making.

Such scientific methods as mathematical techniques of graph theory, algorithm and combination methods, probability theory method, and methods of statistical analysis were used in the research.

Keywords **– Economics and industrial management, flow graph, situation identification, Bayesian composite task.**

DOI: 10.18421/TEM81-20
https://dx.doi.org/10.18421/TEM81-20

Corresponding author: N.V. Speshilova,
Orenburg State University, Prospekt Pobedy,
Orenburg, Russia
Email: spesilovanat@gmail.com

Received: 22 October 2018.
Accepted: 25 January 2019.
Published: 22 February 2019.

1. Introduction

The issue of pattern recognition is well-known: a lot of different recognition technologies have been developed [2]. The systems of automatic object recognition are used in geology, agronomy, medicine, but they are not implemented in the area of industrial management.

Some technologies and recognition systems (e.g. techniques based on logistic rules) have restrictive possibilities. Other recognition technologies (e.g. neural network) are too sophisticated and respectively expensive. The third group of techniques (e.g. techniques based on density evaluation of values distribution for class marks) are difficult to implement as they fail to give a prior description of the recognizable objects.

Hence it seems sensible to use the algorithms of recognizing objects in subsystems of the industrial enterprise [1],[3],[4],[5]. Such algorithms provide apriori information about classes of recognizable objects, consider the correlation between different shared classes of objects, minimizing the loss of information about them.

Among the known algorithms, the Bayesian composite rules are suggested as the most useful for the industrial management [3],[4],[5] for they provide the recognition validity and use the reliable apriori information about different shared classes. The disadvantage of the method is that it demands much calculation for recognition.

To avoid extra calculations, articles [9],[10] suggest an algorithm for recognizing situations for industrial information-management systems, which involves an effective method of reducing computation. The method presents the procedure for recognizing objects, based on a special structure called a flow graph.

The article describes the synthesis method of flow graphs for optimal search within the members of industrial enterprise`s subsystems.

The goal of the article is the development synthesis method of flow graphs for optimal search within the group members (including the search within the industrial structure) and suggests this method as the main implementable algorithm for situation identification, based on Bayesian composite rule for decision-making.

2. Instructions for the authors

During the synthesis of flow graphs, the selection procedure is based on the method of distinguishing a number of common characteristics among the objects of choice [6]. The composition choices of the groups (subsystems) $\{r\}$, chosen for recognition as the collection of variants, are the selection items. Hereinafter, for the sake of convenience, the composition patterns of the group will be presented as sequences of symbols. For example, in the composition pattern of the group *BACB* the item class names are identified as *A*, *B*, *C* and the symbol ordinal number corresponds to the item number in the group. The list of all composition patterns of the group is convenient to present in the "pattern-item" table. For example, Table 1. presents an arbitrary list of the group composition patterns of the size $N = 4$ for the alphabet of five classes *A, B, C, D, E.*

Table 1. Arbitrary list of the group composition patterns of the size N = 4 for the alphabet of five classes A, B, C, D, E.

Number of group composition pattern	Item № 1	Item № 2	Item № 3	Item № 4
1	*A*	*A*	*A*	*A*
2	*A*	*A*	*A*	*B*
3	*A*	*A*	*B*	*A*
4	*A*	*C*	*B*	*B*
5	*B*	*C*	*B*	*A*
6	*D*	*D*	*E*	*C*
7	*C*	*D*	*D*	*B*
8	*C*	*D*	*D*	*C*
9	*C*	*D*	*D*	*D*
10	*C*	*D*	*D*	*E*
11	*D*	*E*	*E*	*D*
12	*D*	*E*	*E*	*E*
13	*E*	*E*	*E*	*E*
14	*E*	*E*	*E*	*A*

The analysis of the data in the table enables to construct the flow graph for optimal search within the compositional patterns of the group presented for the recognition.

Before describing the proposed procedure for the synthesis of minimal flow graphs, we introduce some definitions.

The root in order *k* (by analogy with the morpheme "root of the word" in morphology) is called the ordered set of string characters, except for the character under the *k*-th number. For example, for the first line of Table 1., the second-order root is the A–AA character kit and the fourth-order root is the AAA– kit.

A kit of order *k* is a collection of lines with the coincident roots in order *k*. For example, lines 7-10 in Table 1. form a kit of the fourth order. The symbols on the *k*-th place in the set lines are meaningful symbols of these lines.

The main property of the *k* order kit is in the following: when obtaining any values of the characteristic $\{x_j\}$ during the recognition act, the maximum of the calculated sums for each *j*-th of *N* objects [9] in the kit lines

$$S_r = \sum_{j=1}^{N}\sum_{i=1}^{M}\delta_{ijr}\sum_{t=1}^{T}\sum_{q=1}^{Q_t}\delta_{tq}\cdot\log\lambda_{iqt} + \log P(r) \quad (1)$$

depends only on the number of the significant character.

In formula (1) the following notation is used:

M – the number of recognition classes in the alphabet;

i – recognition characteristic number where $i \in \{1,\dots,M\}$

t – recognition characteristic number from the dictionary that has got *T* characteristics;

λ_{iqt} – the conditional hit probability into *q* gradation (from possible Q_t) of *t* characteristic value for *i* - item class in histogrammic description of their distribution laws [8] ;

$\delta_{tq} = 1$, if *i*-characteristic value has got into *q* – gradation; 0, if it hasn`t got

$P(r)$ – a prior occurrence possibility of *r* – composition group;

$\log(\cdot)$ – Logarithm transformation of $(\cdot)$ to any base, e.g. to base *e*.

The volume of the *k* order kit is the number of lines included into the kit given.

A *k* order kit is considered complete if the number of lines in it is equal to the number of classes in the alphabet. Otherwise, the kit is considered incomplete.

A *k* order kit is identified if only the number of lines with the coincident roots of the *k* order in the pattern table is not less than two.

The minimal flow graph of the algorithm for enumerating the patters is constructed as follows.

The list of group choices (Table 1.) having been analyzed, all possible kits are identified and tabulated. Table 2. shows the example for the matter under discussion.

Table 2. Table of kits

Set category	Line number	Root	Significant character	Volume of kit
1st order kit	12, 13	*–EEE*	*D, E*	2
3st order kit	1, 3	*AA–A*	*A, B*	2
4st order kit	1, 2	*AAA–*	*A, B*	2
	7, 8, 9, 10	*CDD–*	*B, C, D, E*	4
	11, 12	*DEE–*	*D, E*	2
	13, 14	*EEE–*	*E, A*	2

Operation tables are filled in. To fill in the table, one of the full kits or one of the kits with the largest storage is chosen. The operation I is assigned to the kit I (henceforward the numbers of the operations will be symbolized by the Latin figures). The operation I → 7, 8, 9, 10 is for the example under discussion as the lines 7 ÷ 10 comprise the kit of the 4th order, which has got the largest storage (Table 3.).

Table 3. Table of the operation I

Number of operation → kit line	Solution	Next operation address	Lines to be analyzed
I → 7, 8, 9, 10	7	II	1, 2, 3, 4, 5, 14
	8	III	1, 3, 5, 6, 14
	9	IV	1, 3, 5, 11, 14
	10	V	1, 3, 5, 12, 13, 14

All kit line numbers are placed into column “solution” as any of them may be a maximum against statistics S_r.

For each solution next column of Table 3. is filled in by the line numbers of the source table, which significant character differs from the significant character of the line for other solutions. For example, the lines with other characters rather than characters *C, D, E* are placed into the space for the solution 7. The characters *C, D, E* are the significant symbols for the solutions 8, 9, 10. Lines with the numbers 1, 2, 3, 4, 5, 14 are placed into the space for the solution 7. Thus the choices with decision statistics less than the maximum are not analyzed.

Then the table of the operation II is filled in (Table 4.). First, all lines with full kit, then the lines due to decrease in storage, are chosen for each operation.

The spaces of the Table 4. and next tables are filled in by analogy. However, only lines for the operation under discussion are to be analyzed.

Table 4. Table of the II operation

Number of operation → kit line	Solution	Next operation address	Lines to be analyzed
II → 1, 2	1	VI	VI: 3, 5, 14
	2	VII	VII: 4
→ 1, 3	1		2, 14
	3		4, 5, 14

The number of the next operation is assigned to the kit which has got the least number of lines to be analyzed. For example, the notation «→ 1, 3» for the kits 1, 2 in the Table 4. means that the kit wasn`t assigned the number of operations due to the reason mentioned above.

The tables of the next operations (Tables 5-8) are filled in until a single line with the kit to be analyzed is left. The remaining lines are united into one operation under the notation, for example, *VI: 3, 5, 14.*

Table 5. Table of the operation III

Number of operation → kit line	Solution	Next operation address	Lines to be analyzed
III → 1, 3	1	VIII	VIII: 6, 14
	3	IX	IX: 5, 6, 14

Table 6.Table of the operation IV

Number of operation → kit line	Solution	Next operation address	Lines to be analyzed
IV → 1, 3	1	X	X: 11, 14
	3	XI	XI: 5,11, 14

Table 7. Table of the operation V

Number of operation → kit line	Solution	Next operation address	Lines to be analyzed
→ 1, 3	1		12, 13, 14
	3		5, 12, 13, 14
→ 12, 13	12		1, 3, 5
	13		1, 3, 5, 14
V → 13, 14	13	XII	XII: 12
	14	XIII	XIII: 1, 3, 5

Table 8. Table of the operation XIII

Number of operation → kit line	Solution	Next operation address	Lines to be analyzed
XIII → 1, 3	1		
	3	XIV	XIV: 5

The minimal flow graph for the optimal search within the members of the group is framed according to the tables.

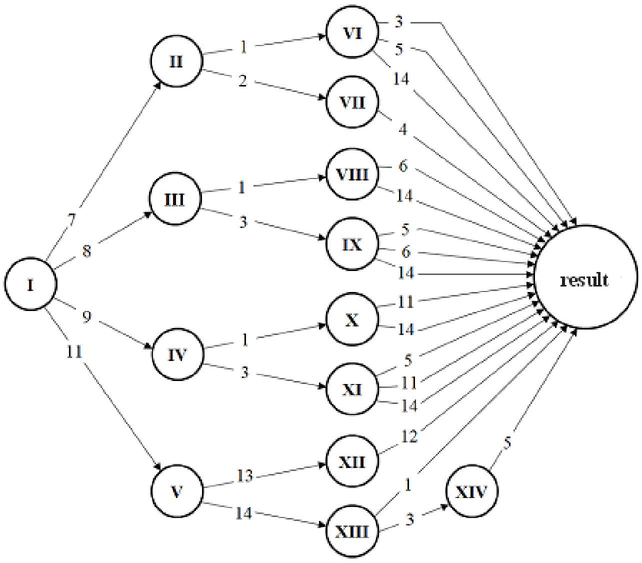
The node with the same number in the flow graph corresponds to each operation and the arc, outgoing from the node, corresponds to each solution. The arc enters the node, which number is identified in the graph “Next operation address”. For example, for Table 3. the initial node I arcs will lead to the nodes II, III, IV, V.

The development of the minimal flow graph starts from the node I and goes on till the node, corresponding to the last one from the operation table, is framed.

The development of the flow graph ends in the final node “Result”, where all arcs without next address meet.

The flow graph developed according to the Tables 3-8 is shown in the Picture 1.

It is the minimal flow graph for the optimal search within the group for the choice list of the Table 1.



Picture 1. The example of minimal flow graph for algorithm for the optimal search within the real list of subsystem composition.

The synthesized minimal flow is the compact and convenient way to describe the patterns for the choices in the list and for the optimal search to find the most probable one.

3. Results

For the example described, the minimal flow graph reduces the number of sum computation operations S_r from14 to 7 ÷ 9 and the number of comparison operators of the sums – to 8 ÷ 11. Reducing computation and the number of decision statistics S_r comparison operations will result in a greater efficiency for a larger list of choices of the real group composition, developed according to the characteristics of the specific recognition domain.

It should be noted that the flow graph has got two more nodes of two types beside the final node:

- the nodes whose outgoing arcs comprise a kit;
- the nodes whose arcs do not comprise a kit.

The nodes of the first type are called the efficient nodes, the nodes of the second type are called the non-efficient nodes and the node with incomplete kit, are called – efficient incomplete nodes.The flow graph method for optimal search [9] reduces the computation for statistics S_r. The procedure according to the minimal flow graph provides comparison in effective nodes of kit lines; the lines of k order kit differ from each other only by k member.

$$S_{g\,кр.} = \sum_{j=1,\, j\neq j^*}^{N} \sum_{i=1}^{M} \delta_{ijr^*} \cdot d_{ij} \;, \qquad (2)$$

where j^* – the ordinal number of the kit line significant character;

r^* – the number of one of the kit lines of the node g;

d_{ij} – statistics which is worth counting in the beginning when the characteristics of all group members enter the recognition module:

$$d_{ij} = \sum_{t=1}^{T} \sum_{q=1}^{Q_t} \delta_{tq} \cdot \log \lambda_{iqt} \qquad (3)$$

Here, considering formulae (2) and (3), statistics S_r for kit choices are calculated according to the formula:

$$S_r = S_{g\,кр.} + \log P(r) + d_{i^* j^*} \,, \qquad (4)$$

where i^* – number of group member, which takes place j^* in the choice r of the group member.

Considering the methods of abridged multiplication and addition, the suggested method of computing the decision sums S_r together with flow graph method of their search and comparison reduces effectively the large-scale computation of the composite Bayesian recognition algorithm.

4. Discussion

The decision scheme of member automatic recognition system together with the composite Bayesian algorithm with flow graph procedure for group member choice search can be implemented in hardware and software [7].

According to the minimal flow graph the system, when implemented in hardware, may be handled as a network of connected logical units. However, when we have a fixed recognition alphabet with class number M for member groups of different size we need several different flow graphs that make the decision scheme device more intricate. Besides, this scheme is unable to adapt to changing class alphabet and thus to the list of group members.

To implement the decision scheme, which is adaptive to the changing group lists, is appropriate doing it in software. The software based on the suggested recognition algorithm, processes data arrays which have the minimal flow graphs for the lists of group members of different size introduced in a particular way.

When generating the array, it is necessary that the number w_g, equal to the number of outgoing arcs, should be assigned to each node g of the flow graph.

Array indexes, which must correspond to the minimal flow graph node and its arcs, are determined according to the following conditions:

- if the node g (number w_g) is placed in the data under the number v then its outgoing arcs (member group choices) have the numbers $v + 1$, $v + 2$, …, $v + n$, …, $v + w_g$, where n is the ordinal number of the corresponding node arc, $n = \{1, \dots, w_g\}$;
- the array element number of the next node with ingoing arc n is determined by the addition $v + n + w_g$.

The example of data array generation, corresponding to minimal flow graph, is presented in Table 9.

Table 9. The example of data massive generation, corresponding to minimal flow graph

Node number	w_g → number of array element	Group member choice → number of array element				Number of array element with w_g of the next node → number of array element			
I	4 → v	*CDDB* → v+1	*CDDC* → v+2	*CDDD* → v+3	*CDDE* → v+4	v+9 → v+5	v+14 → v+6	v+19 → v+7	v+24 → v+8
II	2 → v+9	*AAAA* → v+10	*AAAB* → v+11			v+29 → v+12	v+36 → v+13		
III	2 → v+14	*AAAA* → v+15	*AABA* → v+16			v+39 → v+17	v+44 → v+18		
IV	2 → v+19	*AAAA* → v+20	*AABA* → v+21			v+51 → v+22	v+56 → v+23		
V	2 → v+24	*EEEE* → v+25	*EEEA* → v+26			v+63 → v+27	v+66 → v+28		
VI	3 → v+29	*AABA* → v+30	*BCBA* → v+31	*EEEA* → v+32		V → v+33	V → v+34	V → v+35	
VII	1 → v+36	*ACBB* → v+37				V → v+38			
VIII	2 → v+39	*BDEC* → v+40	*EEEA* → v+41			V → v+42	V → v+43		
IX	3 → v+44	*BCBA* → v+45	*BDEC* → v+46	*EEEA* → v+47		V → v+48	V → v+49	V → v+50	
X	2 → v+51	*DEED* → v+52	*EEEA* → v+53			V → v+54	V → v+55		
XI	3 → v+56	*BCBA* → v+57	*DEED* → v+58	*EEEA* → v+59		V → v+60	V → v+61	V → v+62	
XII	1 → v+63	*DEEE* → v+64				V → v+65			
XIII	2 → v+66	*AAAA* → v+67	*AABA* → v+68			V → v+69	V → v+70		
XIV	1 → v+71	*BCBA* → v+72				V → v+73			

The software implementation of recognition system suggests the following order of work. First the number of group members is determined and the corresponding v array element is read. In our example it is the number $w_I = 4$ and it means that the choices in the next four array elements should be read.

Then we integrate S_r according to the formula (4) and compare them.

After that, the ordinal number of the choice n, for which the S_r is maximal, is determined. According to the formula $v + n + w_{I,}$ the number of the array element with the w_{II} of the next node is determined and its value is read.

The same procedure is repeated until the element number of the final node is determined, for example V. This node is the output of the algorithm. The group member choice r with absolute maximum among the statistics S_r is written by the number V, thus "the solution" is developed.

The software program, that implements the suggested algorithm, is adaptive to the changing list of group choices. In case the list is changed the information complying with the other flow graph should be introduced and the data should be placed in the corresponding array in the manner given as aforesaid.

When the storage of choice lists in the form of the graphs is restricted, the quasiminimal flow graphs are recommended.

The quasiminimal flow graphs can be deduced by uniting identical nodes, i.e. nodes that have the outgoing arcs with the same number.

This approach may increase the route length, but it enables to reduce effectively the size of data arrays in flow graphs. For quasiminimal flow graphs, the algorithm of the recognition decision scheme for group members is the same as for the minimal graphs.

5. Conclusion

The suggested synthesis method of flow graphs for optimal search within the member group enables to implement Bayesian recognition algorithm of objects in the decision support systems to increase the economic and managerial efficiency.

References

[1]. Barabash, Yu. L. (1983). *Collective Statistic Decisions in Recogition*. Moscow: Radio and svyaz`.

[2]. Devroye, L., Gyorfi, L., & Lugosi, G. (1996). *A probabilistic theory of pattern recognation*. Berlin: Springer-Verlag.

[3]. Duda, R. O., & Hart, P. E. (1976). *Image Recognition and Scene Analysis*. Moscow: Mir.

[4]. Gorelik, A. L., Barabash, Yu. L., Krivisheev, O. V., & Epstein, S. S. (1990). *Selection and Recognition based on Location Information*. Moscow: Radio and svyaz'.

[5]. Krivosheev, O. V. (1988). Statistic Procedures of Object Recognition inGroups. In *Evaluation and Recognition of Stochastic Systems:materials of the reseach-to-practice conference*. Kiev.

[6]. Orlov, V. A. (1982). *Flow-graphs of Recognition Algorithm (to apply to geophysical tasks*. Moscow: Nauka.

[7]. Pearl, J. (2000). *Causality: Models, Reasoning, and Inference*. Cambridge: Cambridge University Press.

[8]. Shepel`, V. N., & Akimov, S. S. (2014). Modernization of Histogrammic Approach for Probability Distribution Law. *Bulletin of Orenburg State University, 9* (170), 179-181.

[9]. Shepel`, V. N., & Tripkosh, V. A. (2017). Recognition Algorithm of Industrial Situationsin Information and Management Systems on the basis of Composite Bayesian Task Solution. *Intellegence.Innovation.Investments, (12),* 97-101.

[10]. Tripkosh, V. A. (2016). Composite Flow Graph Algorithm for Recognitionof Situations in Information and Management Systems. *Innovation Technologies in Modern Science: materials of International research- to-practice conference*, Ufa.