INTELLIGENT METHODS OF REVEALING FRAGMENTS IN BUSINESS PROCESSES

Nataliia Golian, Vira Golian, Olga Kalynychenko

***Abstract**: The Effective methods of intelligent analysis of business processes, in particular, methods of revealing fragments of such processes are developed. Besides, analyzing information extracted from journals of registering events of a business process (BP) to formalize the real behavior of a BP is carried out. Such data analysis is especially important in those cases when the occurring sequence of events is registered, i.e. executives have an opportunity to make a decision about the order of further process implementation.*

***Keywords**: business process, procedure, logical net, intelligent analysis.*

ACM Classification Keywords: I.*2 Artificial Intelligence – Knowledge Representation Formalisms and Methods*

Introduction

The term, which occurs the most frequently in this paper, is a "business process". According to literature it is orderly set of tasks that require one or more factors of production and generate result, which is focused on fulfilling customers' needs. Modeling and computer simulation have an increasingly important position amid tools are used by engineers and managers. This is the result of need to make quick and accurate decision in response to constantly changing environment. Furthermore, manufacturing systems are more complex than before. Through information technology that are applied in conjunction with software for modeling and simulation, issues difficult to solve due to the high complexity, can by deeply analysed. Modeling is to build a virtual model, which illustrates a real business process. Then, simulations are carried out on this model.

The purpose of this paper is an intelligent analysis of a business processes for formalizing its actual behavior.

The actuality of considering the structure and characteristics of business processes in the context of the present work is defined by the need for developing efficient methods of intelligent analysis of business processes, particularly, such processes.

At present there is a change-over to process management from a traditional functional one, which requires the formalization of existing business processes by constructing their hierarchical structure with the use of typical fragments of the processes. Hence, working out details of the structure and characteristics of business processes is a necessary condition of researching and developing methods of intelligent data analysis.

Constructing formal models of business processes requires considerable time and financial expanses as well as it is affected by a human factor, as it is often realized in concert with experts and executives, whose aims cannot coincide with the ones of the business processes under simulation and the enterprise common goals. This contradiction can lead to discrepancy between a real business process and its developed model. It can result in constructing inadequate models of business processes.

One of the main approaches to solving the given problem is realized on the basis of the methodology of intelligent analysis of business processes. The intelligent analysis is directed towards obtaining models of really implemented business processes on the basis of researching logbooks of events of such processes (files – logs).

Such data analysis is particularly important in those cases when an occurring sequence of events is being registered, however, the process is partially or completely non-formalized, i. e. executives have an opportunity to take decisions about the order of further process implementation, proceeding from the local information they have and in view of a BP. In fact, implicit relations between BP procedures are realized in such processes. The said relations are based upon the knowledge which is not involved in the process description, they can result in variations from its documentary behavior and at that are not practically identified on the basis of existing approaches in the field of intelligent analysis of business processes.

At present algorithms of constructing models of business processes on the basis of analyzing logbooks of events are developed [Agrawal, 1998; Aalst, 2003; Medeiros, 2003; Aalst, 2004]. At the same time the developed approaches do not allow to completely solve the problem of revealing implicit relations and, with their help, constructions of implicit choice in the structure of BPs.

Importance of simulation modeling and value analysis in the present-day world

One of the main trends that have been outlined lately in the field of information technologies is the increased interest for methodological and technological problems of using simulation modeling in the practice of researching and designing sophisticated systems in various application areas that is due to the following causes:

- by extending of the area of applications of simulation modeling first of all owing to such nontraditional directions as BPs, marketing, logistics, financial management, socio-economic processes, etc.
- by increasing of the level of manufacturability of simulation systems due to render features: graphical interface, animation as well as Case – technologies. Lately the unified language of visual representation of models of program systems – UML (Unified Modeling Language), developed by the famous American experts in the field of object – oriented programming Gradi Buch, Jimmy Rumbach and Ivar Jacobson has been widespread.
- By mass use of Internet – technologies for both support distance learning processes and realizing simulation experiments on the basis of modern network technologies. At present the websites of such famous experts in the field of simulation modeling as R. Sergeant, O. Balchi, R. Fujimoto and others are accessible to the broad researching public. Through websites one can get materials of such an important event as Winter Simulation Conference which is actually held by the International community of simulators. The Russian simulation portal gpss.ru regularly publishes the "hottest" information from foreign and Russian practice of simulation modeling.
- By development of opportunities of designing and studying sophisticated systems on the base of so-called models of virtual reality.

The reports submitted at the last three Winter Simulation Conferences (2000-2002) are evidence of the increased interest and demand for simulation modeling systems.

The first all – Russian theoretical and practical conference UMMOD-2003 was held in the city of Saint-Petersburgh (Russian Federation). Over 30 reports were devoted to research in different application areas (space manufacturing, logistics, medicine, ship building, transport, etc.). The reports presented are evidence of the scale and high level of projects being developed in Russia at present on the basis of simulation modeling methods.

Main information about simulation modeling and value analysis

Simulation modeling is a method of study allowing to analyze a system without changing it. It is possible due to the system under investigation being replaced by a simulating system, the information obtained is characteristic of the system investigated. Speaking about analysis of a company's activity the method allows simulating the implementation of BPs in such a way as if it occurred in reality and getting a real estimation of the duration of each process.

Value analysis is an instrument designed to estimate product (service) costing. Implementing value analysis allows to estimate cost price through management of processes directed at manufacturing a product or rendering a service. It is in this that the method of value analysis differs from traditional financial methods of calculation of expenditures within which the company's activity is estimated on the basis of functional operations but not by specific products (services) provided to a customer. The following point underlines value analysis: to produce a product (service) requires implementing a number of processes, consuming specified resources. Expenses on implementation of a process are calculated by transferring costs of resources on costs of process steps. The amount of expenditures on implementing all the processes with definite amendments constitutes product costing. If the traditional methods calculate expenses on a certain activity status by cost categories, value analysis shows costs of implementing all the process steps. Thus, the methods of value analysis allows to calculate expenses on manufacturing products (rendering services) most accurately as well as they present information for analyzing processes and their improvement.

Value analysis and simulation modeling stages include developing a model of processes, giving time parameters of finite (non-decomposed) processes. The resources are subdivided into time and material ones. The cost of a time resource is carried over to the cost of a material resource proportionally to that time which the resource takes to implement the process. The cost of a material resource is carried over to proportionally the quantity of repetitious of a process, purposes of resources on processes, performing a simulation of implementing the processes.

Analysis of the main trends in the area of present-day simulation modeling

One of the main trends in the area of developing and introducing modern systems of BP management is using simulation modeling as their integral part. Simulation systems, embedded into BP management, provide performing of such important tasks as project control, resource planning, control of business rules, investment forecast on the basis of analysis according to the scheme "what-if", training in new/reorganized BPs, script planning for emergency conditions. At that generally accepted is the point of view that simulation modeling must accompany BPs from the very starting stage of their formation, development and introduction.

Examples of such famous modeling systems as SIMUL8 (SIMUL8 Corporation), AutoMod (Brooks Automation), ProModel (ProModel Corporation) and WITNESS (Lanner Group Incorporated) can serve a confirmation of the mentioned trend.

SIMUL8 Corporation develops supplies to the market and supports simulation modeling systems oriented on performing tasks of business, government, education and organizations that face problems of managing flows of orders, clients, transport or production. Within a comparatively short period of time since the date of its foundation in 1994 the corporation has managed to include very many solid firms, namely: IBM, Bell Laboratories, Motorola, Ford Motor Company, Boeing Aircraft, British Airways, Virgin Atlantic, Hewlett-Packard Corporation, USA Air Force, British Steel and Nissan Motors in the list of its customers.

The systems of modeling AutoMod, ProModel and WITNESS have also found wide application in various application areas (manufacture, business, storehouses, logistics, transport, production of pharmaceuticals, reengineering in business) and have found such firms as General Electric, Intel, Siemens, Nokia, Motorola etc. as serious customers.

Two directions are distinguished in the VR-area of simulation modeling: the first one is related to videogame industry and the second direction is in the first place related to problems of researching and analyzing industrial processes on the basis of e-Manufacturing concept which received its development in the automobile manufacture industry of Germany in the late 1990s.

The main aim of using e-Manufacturing is progressing to such stage of modeling objects and processes which provides a possibility of studying in detail and optimizing all aspects of any productive process before starting its initiation.

It is natural that transferring to the e-Manufacturing regime especially of large-scale enterprises can only be gradual. Such concerns as Mercedes-Benz PkW, Opel, BMW, Audi, Toyota, Airbus (when manufacturing the airbus A-380 in Hamburg) are planning to introduce the idea of e-Manufacturing.

As a whole the concept of e-Manufacturing can be represented by the formula “Simulation + Virtual Reality”. Implementing the concept of e-Manufacturing requires having the following software support tools:

- Storage of text and graphic data represented in different formats;
- Simulation modeling of systems and processes investigated.
- Visualization of results of modeling by VR methods.

Only two firms – DELMIA and Technomatix – are ready to offer complete sets of mutually compatible software products for supporting e-Manufacturing concept at the European market of software products. The core of each system is a special data bank which represents three basic structures of industrial purpose – PPR (Product, Process, Resources). This bank is called an e-Manufacturing Server and PPR Hub by Technomatix and DELMIA respectively. Technomatix uses EM-Plant and DELMIA employs QUEST as simulators.

The Magdeburg Institute of Organization and Automatization of Industrial Production named after Fraunhofer – IIF –carries out a large scope of work on use of simulation modeling in industry (particularly, the EM-Plant systems) as well as on creating VR-models.

The institute has accumulated a considerable experience of developing both VRMI-models for representing industrial processes (including simulation modeling -based one) and special VR-models fully dipping a customer into virtual space. The latter allows him/her to do work on designing and testing machines and equipment. VR-models are used to instruct and train people – operators mastering new operations.

Thanks of the achievements of the institute in the area of developing and constructing virtual models a decision has been taken to construct the Virtual Development and Training Centre (VDTC) in Magdeburg. It is planned that the enterprises and organizations will be proposed a wide spectrum of services on bringing facilities and technology of developing VR-models to a commercial level, training of personnel operating sophisticated equipment, implementing repair and preventive work.

Problem statement

In really functioning BPs one can distinguish two types of choice of procedures which represent different kinds of relations between them: explicit and implicit ones.

The explicit relations [Desel, 2005] represent cause – and – effect relations between procedures. At that such procedures are usually available in pairs in a logbook of BP events. Example: For starting the implementation of procedure P2 it is necessary that procedure P1be completed.

Implicit relations represent indirect cause – and – effect relations between procedures. The interrelation between P1 and P2 is not seen immediately from the logbook of events. For instance, procedures P1 and P2 can be interconnected through a sequence of procedures <P3, P5>. It is this that defines the importance of formalizing typical constructions of implicit choice.

Hence, the problem is to obtain formal algebraic logical models of constructions of implicit choice on the base of analyzing the main peculiarities of typical sequences of procedures with implicit choice in BPs as well as errors of revealing such fragments with existing algorithms.

Implicit relations in BPs represent indirect cause – and – effect relations between procedures and possess the features of connectivity, reachability and do not have the feature of a sequence. In the case of implicit relations between the considered procedures of a BP there exists a chain of other procedures (indirect relation), which makes the revelation of such fragments more difficult.

All the above mentioned defines the importance of formalizing implicit relations between procedures.

The paper's task consists in obtaining formal models of implicit relations between BP procedures that would possess the following peculiarities:

- Representing parallel and sequential implementation of the current fragment of a BP and its other subprocesses;
- Covering the necessary and sufficient set of features of implicit relations allowing to single them out on the basis of analyzing the sequence of procedures of the BP implemented.

Algebraic logical models of implicit choice constructions

Define formally explicit and implicit relations between BP procedures by finite predicate algebra. Let us enter variables $x_{1,}, x_2, ..., x_n,$, denoting the states of procedures $P_{1,2} P_{2,} ..., P_n$. These variables are given on some finite set of possible values of procedure states. For example, $x_1 \in \{a, b, c\},$, where $x_1 = a$ means "procedure p_1 has not been implemented", $x_1 = b$ – "procedure p_1 is being implemented", $x_1 = c$ means "procedure p_1 has been implemented".

Let us enter predicates $x_{1,}, x_2, ..., x_n,$, $L_{1,}, L_2, ..., L_k,$, denoting pair wise relations (if they exists) between procedures $P_{1,2} P_{2,} ..., P_n$. For the example mentioned above the relation between procedures P_1 and P_2 will be described by a predicate $l_1 (x_{1,} x_2)$.

The constructions implementing implicit choice are characterized by a contradiction between a choice of some alternatives and the necessity of synchronizing chosen actions with those already being implemented within a BP. In other words, the implicit choice situation is characterized by a combination of synchronization and choice constructions, as shown in Fig.1.

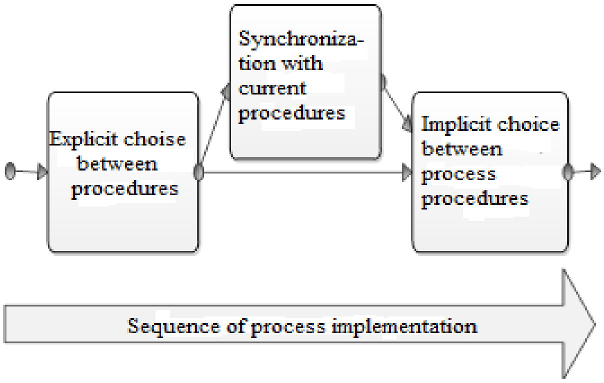

Fig. 1. Implicit choice situation between business process procedures

As it is seen from the figure, the implicit choice is defined by a choice of such and such procedures at the previous stages of a process and is implemented after synchronizing the results of the earlier choice with current procedures. Such synchronization is necessary so that all input conditions for final procedures, between which an implicit choice is made in the course of implementing a BP, may be satisfied. The problem of revealing implicit choice structures in intelligent analysis problems of BPs is defined by the fact that the existing algorithms whose mathematical base is formed by Petri nets, particularly WF-nets [Li, 2003] as their extension, usually cannot process such constructions.

Fig.2 shows a situation, according to which the final result of implementing the current BP fragment depends on an implicit choice between procedures P_4 and P_5.

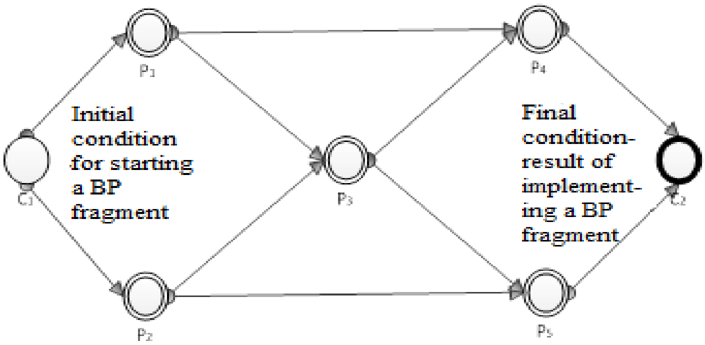

Fig. 2. Situation of implicit choice between P4 and P5

Implementing the given fragment starts in the case if initial condition C_1 is satisfied. Then an explicit choice between procedures P_1 and P_2 is implemented. The result of implementing procedures P_1 and P_2 must be synchronized with the result of procedure P_3, following which a choice between procedures P_4 and P_5 is made. As choosing one of the mentioned procedures is defined by a choice of P_1 and P_2 at an earlier stage of implementation, we obtain that the choice between P_4 and P_5 is implicit.

The operation of such a logical net will consist in a cyclic check of a set of all the relations between its BP procedures which exists in a modeled BP, which at each step of calculations will take the form of generation of a set of procedures that must be implemented to continue a BP.

Enter variables $x_{1?}, x_2, ..., x_5,$ denoting the states of procedures $P_{1,2} P_{2,} ..., P_5$. These variables can take values from the set $\{0,1\}$, which means implementation or non-fulfillment of the procedures respectively. To construct a logic net let us find a system of binary predicates $L_{1,2} L_{2,} ..., L_k$, that describes the logic of executing a fragment of BPs, shown in Fig.2. With this in mind enter temporary variable t, containing information about explicit and implicit choices of procedures. The variable t can take values from the set $\{0,1,2,3\}$, denoting implementation of procedures $P_{1,2} P_{2,}, P_1$ and $P_{3,2} P_{2,} and P_3$.

The system of predicates describing a logical net of the first typical implicit choice situation takes the form:

$$\begin{cases} L_1(x_1, x_3), \\ L_2(x_{1,} t,), \\ L_3(x_{2,} x_3), \\ L_4(x_2, t), \\ L_5(x_3, t), \\ L_6(t, x_4), \\ L_7(t, x_5). \end{cases} \quad (1)$$

The appropriate logic net is shown in Fig. 3.

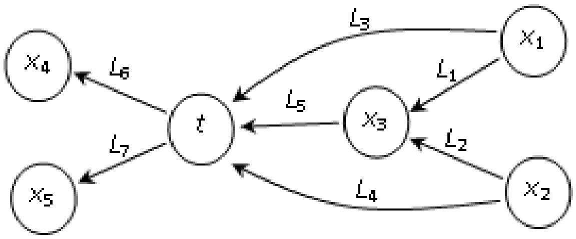

Fig. 3. Logical net of the first typical situation

The initial nodes of logic net are variables x_1, x_2 and the final ones are variables x_4, x_5. At the beginning of operation of a logical net all its variables, modeling BP – procedures, have the value of 0 – none of the procedures is not implemented. In the course of operation of the logical net, i. e. calculating a system of binary predicates which corresponds to this net, all or some variables *x1, x2,…, x5* of the logical net take the values of 1 – all the procedures have been implemented. The variable t will take the value, corresponding to a sequence of implementing procedures, which is realized in a BP.

Consider an example of operation of a constructed logical net in steps:

Step 0 (beginning of operation of a net): x1=0,…,x5=0, tis not defined.

Step 1: x1=1, t=2, the remaining nodes have no changes.

Step 2, halfstep 1 (net state) :x3=1.

Step 3 (completion of operation of a net) :x4=1.

It is obvious that, if the implementation of a procedure P2 is the initial action in the net, then completion of the net operation will result in x5=1. The second typical situation of implicit choice of a procedure sequence is shown in Fig. 4.

In the given situation there are two implicit choices – between procedures P_4 *and* P_5, as well as between procedures P_3 *and* P_5.

Sequence of this fragment is as follows. The initial condition C1 is an input for a single procedure P1. As a result, its performance can be parallel (in any order) performed the procedure P2 and P3 respectively. Further, there are several implementation options.

Option 1. In that case, if the procedure is executed before the procedure P2 P3 is the procedure P3, P4 and P5 can be performed in any order, independently of each other. Then you can choose from two parallel branches - either performed the procedure P6, or both of the procedure, P6 and P7.

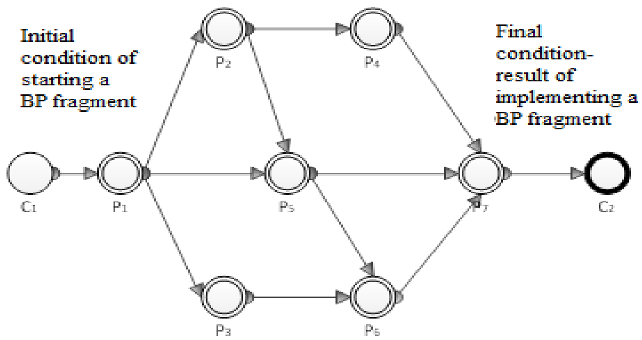

Fig. 4. Situation of implicit choice between procedures P_4 and P_5, P_3 and P_5.

Option 2. If the procedure is executed before the procedure P3 P2, then it becomes a further order of the hard-coded: P2 should be executed after the P4 and P5, P5 is performed after the P6, and then - P7.

Thus, the set of possible sequences of the procedures can be represented as tuples of the following:

{<P1, P2, P5, P6, P7>, <P1, P2, P3, P4, P6, P7>, <P1, P3, P2, P6, P4, P7>, <P1, P2, P4, P3, P6, P7>, <P1, P3, P6, P2, P4, P7>}.

The problem of identifying the fragment is as follows. Existing data mining algorithms do not recognize the connection between the procedures P1 and P5, P5 and P7, which can lead to delays and deadlocks in process models, in particular, if a pair of parallel processes P4 and P5 will be performed only one procedure P5. The direction of solving this problem is to identify the cause - effect relationship between the procedures P1 and P5, P5 and P7 on the basis of analysis of event log.

We construct algebraic-logical model of a typical situation, an implicit second choice of the sequence of procedures similar to the first type situation. We introduce the variables x1, x2, ..., x7, denoting the state of processes P1, ..., P7. These variables can take values from {0, 1}, respectively, which means the performance or failure to comply with procedures. To construct a logical network will find the system binary predicate L1, L2, ..., Lk, which describes the execution logic of business process fragment shown in Fig. 2.4. For this purpose, we introduce an intermediate variable t, which contains information about the sequence of explicit and implicit election procedures. The variable t can take values from {0, 1} by the formula: t = 0, if the procedure is executed before the procedure P2 P3; t = 1, if the procedure is executed after the procedure P2 P3. The system of binary predicates, that describes a logical net of the second typical situation of implicit choice, consists of 12 predicates $L_{1_?}, L_2, \ldots, L_{12}.$. The appropriate logic is shown in Fig. 5.

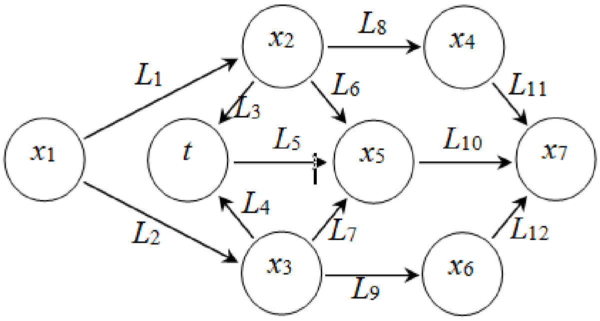

Fig. 5. Logical net of the second typical situation

The model of a BP in the form of a logical net, as shown above, generates at any moment of time a set of procedures that must be implemented to continue the BP. In fulfilling any next procedure the situation changes, which at once will result in the change of the state of a logical net and, respectively, the change of a set of procedures which must be realized to continue the BP.

Typical situation 3 of implicit choice of the sequence of procedures is shown in Fig. 6. There exists an implicit choice between procedures P3 and P5. The sequence of implementing the given fragment takes the following form. After realizing procedure P1 the nets operate practically in a sequential regime and do not have an advantage over the models of the same processes in the form of multi-place predicates.

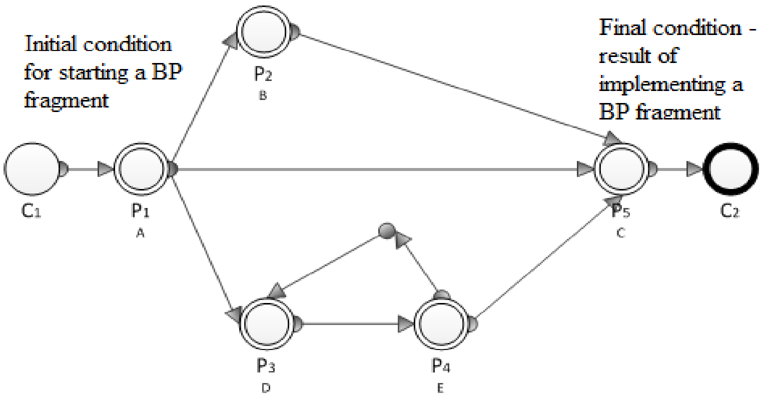

Fig.6. Situation of implicit choice between procedures P3 and P5

P5 can be executed after the loop P3 → P4, and the procedure P2.

Consequently, the possible sets of sequences of the procedures can be represented as tuples of the following:

{<P1, P2, P5>, <P1, P2, P3, P4, P5>,

<P1, P3, P2, P4, P5>, <P1, P3, P4, …, P3, P4, P2, P5>}.

The solution of the problem considered in this case involves finding the relationship between the procedures P1 and P5.

Construct a logical network of third typical situation an implicit choice of the sequence of procedures similar to those discussed above situations. We introduce the variables x1, x2, ..., x5, denoting the state of processes P1, ..., P5. We find the system binary predicate L1, L2, ..., Lk, which describes the execution logic of the movie business processes depicted in Fig. 2.6. For this purpose, we introduce an intermediate variable t, which contains information about the implicit choice between procedures, P3 and P5. The variable t can take values from {0, 1} by the formula: t = 0, if the procedure is executed before the procedure P3 P5; t = 1, if the procedure is executed after the procedure P3 P5.

The system of binary predicate that describes the logical network is a typical situation, an implicit second option consists of seven predicates L1, L2, ..., L7. The corresponding logical network is shown in Fig.7.

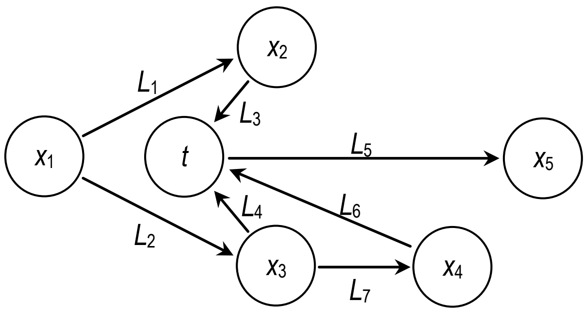

Fig. 7. Logical network of third typical situation

Consider the example of the logical network built on the cycles:

0 cycle (beginning of the network): x1 = a, the remaining nodes unchanged.

1 cycle: x2 = 1, the remaining nodes unchanged.

2 cycle: x3 = 1, t = 0, the remaining nodes unchanged.

3 cycle: x4 = 1, the remaining nodes unchanged.

4 cycle (the end of the network): x5 = 1.

Development of predicate models of implicit relations between procedures

Developing the method of identifying implicit choice situations requires formalizing of the main features of such situations, namely, formalizing of implicit relations of various type, which requires developing models of representing implicit relations between procedures.

This subsection is devoted to the development of predicate models of representing implicit relations between BP procedures on the basis of the algebraic logical model of a generalized implicit choice construction. The given model combines four schemes of interaction of an implicit choice construction and other BP fragments. Therefore four types of implicit relations between procedures will be further considered and formalized in the form of predicate models.

Realization of implicit choice models is based on a predicate model of representing an indirect relation between procedures in the logbook of BP events. Consider and conclusively formalize the implicit relations of all four types. Implicit relation of type 1: outputs of an outside subprocess – inputs of an analyzed fragment.

The given type of a relation is based upon the interaction of the form: output of an outside subprocess – inputs of final procedure Pi and the intermediate chain of procedures P2, … ,Pn (Fig. 8).

In accord with the given scheme of interaction let us formulate a set of conditions defining the relation of the given type:

- There is no explicit relation between initial P1 and final Pn procedures of the fragments under study.
- The initial procedure of fragment P1 in the model, obtained on the basis of analyzing a log of operations, has only one output that is an input of the procedure $P_j, j = \overline{2, n-1}$
- The procedure Pj has a second input which is common for the final procedure Pn.
- The input, common for procedures Pj and Pn, is the result of operation of procedure Pk that falls into a different situation of the given BP or a different subprocess. It should be noted that splitting in situations and subprocesses is included in the structure of a logbook of events, as there usually exists the column “Situation code”
- Procedure P5, outside relative to the analyzed fragment (generalizing the whole outside subprocess), is not related to initial procedure P1 in input-output;

- There is an indirect relation between procedures P1 and Pn.

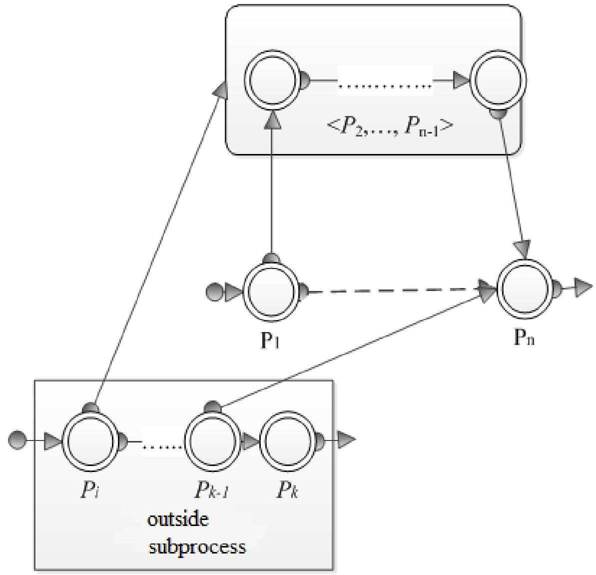

Fig.8. Implicit relation of type 1: outputs of the outside subprocess – inputs of the fragment under study

There is a type 1 implicit relation in meeting the considered six conditions between procedures P1 and Pn.

Summing up the above-mentioned, one can say that the given relation allows identifying an implicit choice construction in the BP model obtained as a result of analyzing a logbook of events. Revealing such a construction occurs when in the analyzed model fragment there is such an intermediate sequence of one or more procedures that an input of the given sequence simultaneously with an output of the final procedure of a fragment is defined by an output of an outside subprocess. Besides, the final fragment procedure has a second input, then there is an implicit relation between the initial and final procedures in the second input.

Type 2 implicit relation: prove the presence of the initial relation on the basis of an simplified scheme of interacting an implicit choice construction with other BP fragments on the base of a relation in input in Pn. The simplification consists in the fact that the sequence of procedures <P2…Pn> is replaced by the single procedure Pn, and the outside subprocess is replaced with the separate procedure P2. The given simplification does not effect on the heart of the proof, as representation of a part of the process as a sequence of procedures or as a common generalized procedure, realizing the whole subprocess, depends upon the degree of the model details worked out(Fig. 9)

Proceeding from the interactions represented in Fig.2, formulate a set of conditions defining the implicit relation of the given type:

- There is no explicit relation between initial P1 and final Pn procedures of the fragment under study.
- The initial procedure of fragment P1 in the model, obtained on the basis of analyzing the logbook of operations, has two outputs (or more – in the general case) that are inputs:
    - for the initial procedure $P_j, j = \overline{2, n-1}$ of the outside subprocess
    - for procedure P2 of the current BP fragment.
- There is an implicit relation in the sense of expression between procedures P1 and Pn.
- The results of implementing procedure Pn-1 are used as input ones for procedures Pk and Pn
- The final procedure of the outer subprocessPk is not connected by an indirect relation with initial procedure Pn.

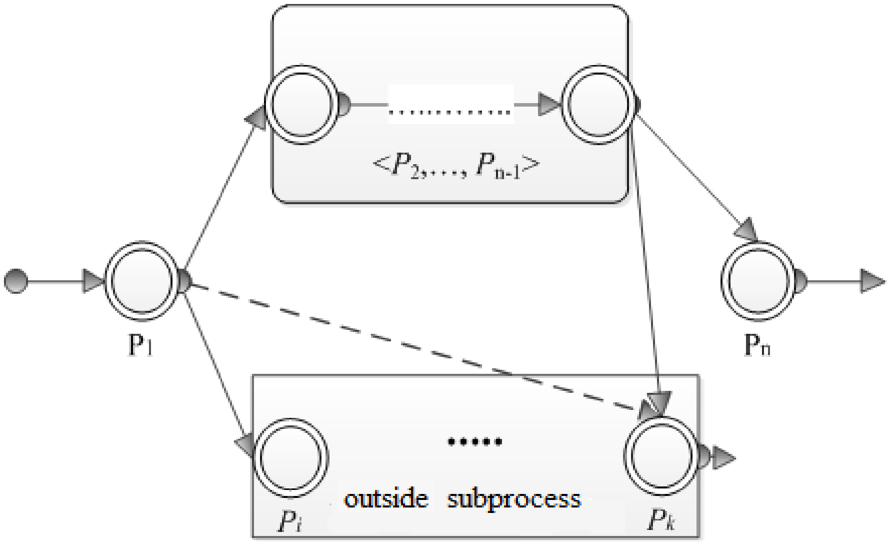

Fig.9. Implicit relation of type 2: outputs of analyzed fragments - inputs of an outside subprocess.

To be more exact, such a relation is not guaranteed in the general case. There is a type 2 implicit relation between procedures P1 and Pk when meeting the considered five conditions.

The set of the conditions formulated allows to identify an implicit choice construction in a BP model, obtained as a result of analyzing an event recording logbook, in the case if in the current model fragment there are two (in the general case more than two) variants of a process following that terminate with procedures Pk and Pn. Their implementation depends in which chain - <P2 …Pn> or on the outside subprocess <Pi ..Pk> the realization of a BP will follow after implementing the initial procedure. Then, if under the real execution of a process, reflected in an event recording logbook, procedure Pn has been implemented, it means that the chain <P2…Pn-1> has been realized.

Type 3 implicit relation: the outside subprocess is implemented concurrently affecting the sequence <P2…Pn-1> (Fig. 10).

On the basis of analyzing the presented scheme formulate a set of conditions defining the relations of the given type:

- Output of initial procedure P1 – input into the intermediate fragment <P2…Pn>;
- Output of the initial procedure of outside subprocess Pi;
- Input into the intermediate fragment <P2…Pn>;
- Output of the procedure Pn-1 - input into the final procedure of outside subprocess Pk;

Output of the last procedure of the intermediate sequence of the current fragment Pn-1 input into the final procedure of outside subprocess Pk.

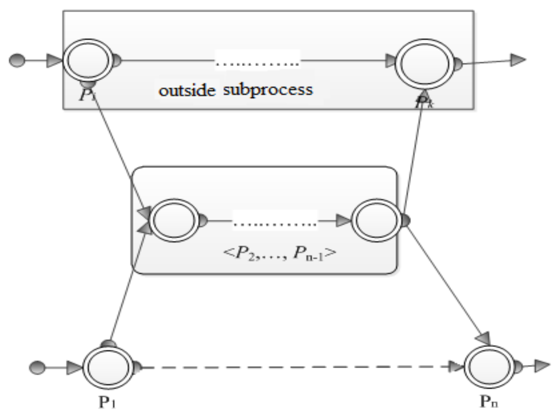

Fig.10. Type 3 of implicit relation is implemented concurrently

On the basis of analyzing the presented scheme let us formulate a set of conditions defining the relations of the given type:

- There is no explicit relation between initial P1 and final Pn procedures of the fragment under study.
- There is an indirect relation between initial P1 and final Pn procedures of the fragment under investigation;
- The first procedure P2 of the intermediate sequence <P2 … Pn> has two inputs:
  - from initial procedure P1 of the current fragment of the BP of an outside subprocess.
  - from initial procedure Pi of an outside subprocess.
- The results of implementing procedure Pn-1 are used as input ones for procedures Pk and Pn-1.
- The final procedure Pk of the outside subprocess is not indirectly related to initial procedure P1. To be more exact, such a relation is not guaranteed in the general case.

There is a type 3 implicit relation between procedures Pi and Pk when satisfying the considered five conditions.

The conditions, given above, allow to identify a type 3 implicit choice construction in that case if an outside subprocess is implemented concurrently with the current BP fragment, which results in the appearance of a construction with two input procedures and two output ones, defining two variants of a process following. Implementing this in that variant depends upon which chain - <P2 … Pn-1> or on outside subprocess <Pi … Pk> the BP will be realized after executing initial procedures. Then, if implementation of a procedure is represented in the event recording logbook, it means that execution of the chain <P2 … Pn-1> has occurred and, hence, there is an implicit relation between the procedures.

Type 4 implicit relation: the outside subprocess is implemented concurrently and with depending upon the sequence <P2 … Pn-1>.

The given relation is the detail of interactions shown in Fig. 2 and based upon the concurrent implementation of the fragment under consideration and the outside subprocess; the outside subprocess start is defined by a current fragment, and its completion effects the implementation of the current fragment (Fig. 11):

- Output of procedure P2– input into the initial procedure of outside subprocess Pi;
- Output of the final procedure of outside subprocess Pk - input into the final procedure of the intermediate sequence of the current fragment Pn-1.

On the basis of analyzing the scheme, shown above, formulate a set of conditions allowing to formalize the relations of the given type:

- There is no explicit relation between initial P1 and final Pn procedures of the fragment under study.
- There is an indirect relation through the sequence <P2 … Pn-1> between initial P1and final Pn procedures of the fragment under investigation.
- The first procedure P2 of the intermediate sequence has two outputs:
    - into the subsequent procedure of the current BP fragment;
    - into the initial procedure Pi of an outside subprocess.

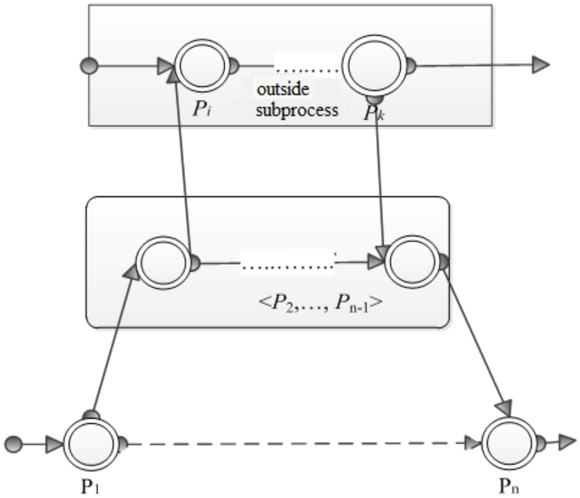

Fig.11. Type 4 of implicit relation: outside subprocess is implemented concurrently

- The results of implementing an outside subprocess procedure are used as input ones for procedure Pk;
- The final procedure of the outside subprocess is not indirectly connected with the initial procedure. To be more exact, such a relation is not guaranteed in the general case.

There is a type 4 implicit relation between procedures Pi and Pk when meeting the considered five conditions.

Predicate models of implicit relations between procedures which are obtained in the given subsection, represent different variants of concurrent and sequential execution of the current fragment of a BP, presenting the current situation, with other subprocesses.

The mentioned models allow formally representing implicit knowledge about interactions between procedures and, therefore, increasing the adequacy of the model obtained as a result of analyzing of a logbook of recording BP events.

Conclusion

The obtained algebraic logical models of implicit choice constructions represent logical nets in the form of a system of predicates representing interactions between the procedures of the construction under study. Such models allow to present implicit cause-and-effect relations between appropriate procedures which are not directly presented in a logbook of recording BP events.

Basing on examples of operation of constructed logical nets, one can see that only one predicate is calculated in the bigger part of steps, which states that these logical nets operate practically in a sequential regime and have no advantages over models of the same processes in the form of multi-place predicates. However, if one takes into account that only small key fragments of real BPs have been covered in the examples considered, and full models often contain a lot of procedures which can be implemented simultaneously, then the advantages of the models in the form of systems of binary equations become evident.

The practical aspect of the results obtained consists in the following. Implementing a logical net, realizing implicit choice constructions, ensures a possibility for obtaining a logbook of recording events with representation of implicit interactions between procedures which creates conditions for developing methods of revealing implicit choice constructions.

Bibliography

[Agrawal, 1998] R. Agrawal, D. Gunopulos, and F. Leymann.Mining Process Models from Workflow Logs. [Text]/ In Sixth International Conference on Extending Database Technology – 1998 – 469-483 pages.

[Aalst, 2003] W.M.P. van der Aalst, B.F. van Dongen, J. Herbst, L. Maruster, G. Schimm, and A.J.M.M. Weijters. [Text]/ Workflow Mining: A Survey of Issues and Approaches. Data and Knowledge Engineering, 47(2):237–2003 – 267 pages.

[Medeiros, 2003] A.K.A. de Medeiros, W.M.P. van der Aalst, and A.J.M.M. Weijters. Workflow Mining: Current Status and Future Directions. In R. Meersman, Z. Tari, and D.C. Schmidt, editors/ [Text]/ On The Move to Meaningful Internet Systems 2005: CoopIS, DOA, and ODBASE, volume 2888 of Lecture Notes in Computer Science, pages 389 -406. Springer-Verlag, Berlin, 2003.

[Aalst, 2004] W.M.P. van der Aalst, A.J.M.M. Weijters, and L. Maruster. Workflow Mining: Discovering Process Models from Event Logs. [Text]/ IEEE Transactions on Knowledge and Data Engineering –2004– 16(9):1128-1142 pages.

[Desel, 2005] J. Desel and J. Esparza. Free Choice Petri Nets [Text] / volume 40 of Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, Cambridge, UK, 1995.- 256 p.

[Li, 2003] M. Z. J.Q. Li, Y.S. Fan. Timing constraint workflow nets for workflow analysis. [Text] / IEEE Transactions on Systems, Man, and Cybernetics, part A: Systems and Humans, 33(2):179–193, March 2003.

Authors' Information

Olga Kalynychenko - *PhD, associate professor of Software Engineering department, Kharkov National University of Radio Electronics Ukraine; Lenin av., 14, 61166 Kharkov, Ukraine; e-mail: okalinichenko@mail.ru*

Vira Golian - *PhD, Software Engineering department, Kharkov National University of Radio Electronics Ukraine; Lenin av., 14, 61166 Kharkov, Ukraine; e-mail: veragolyan@yandex.ru*

Nataliia Golian – *PhD student of Software Engineering department, Kharkov National University of Radio Electronics Ukraine; Lenin av., 14, 61166 Kharkov, Ukraine; e-mail: veragolyan@yandex.ru*