Estimation of Operational Properties of Information Technology Usage and Dynamic Capabilities Indicators

Alexander S. Geyda
St.-Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences
St.-Petersburg, Russia
geida@iias.spb.su

Abstract — **the article outlines conceptual and corresponding formal models that provide means for estimation of dynamic capabilities indicators of systems and operational properties of information technology usage of these systems performance. Dynamic capability is the ability of a system to adapt to changes of the system's environment. The use of information technologies plays a central role in developing such capability of a system. Operational properties indicators of IT usage (such as efficiency, effectiveness, capabilities) defined as a kind of system operational properties indicators under conditions of changing environment in such a way that it is possible to estimate their values analytically. Such estimation is fulfilled through plotting the dependences of predicted values of operational properties of IT usage against variables and options of problems solved. To develop this type of models, the use of information technologies during system functioning is analyzed through an example of a technological system. General concepts and principles of modeling of information technology usage during operation of such systems are defined. An exemplary modeling of effects of technological information and related technological non-information operations of technological systems operation is provided. Based on concept models of operation of technological systems with regard to information technologies usage, set-theoretical models followed by functional models of technological systems operation using information technologies are introduced. An example of operational properties indicators estimation is considered based on ARIS diagramming tools usage.**

I. INTRODUCTION

Dynamic capabilities are usually defined [1] as the ability of a firm to integrate, build, and reconfigure internal and external competences to address rapidly changing environments. A more detailed definition of dynamic capabilities as a firm's "behavioral orientation to continuously integrate, reconfigure, renew, and recreate its resources and capabilities, focusing on upgrading and reconstructing its core capabilities in line with dynamic, changing environment to obtain and sustain competitive advantage" was given in [2]. A role of dynamic capabilities consists in "changing internal components of the firm and creating new changes" [3].

As we can see, these definitions describe the ability of a firm or an organization to change, adapt, compete, and perform in a changing environment. I define system dynamic capability as a systemological property. System dynamic capability is a system's ability to perceive its changing goals in its changing environment. This definition is similar to our previous definition of system's potential and other operational properties of systems and operational properties of information technology usage [4-8]. Other examples of models and methods for definition and estimation of such properties could be found in [9-27]. This ability to perceive system's changing goals in its changing environment requires a system to check system and environment states, and their relations, to learn, to produce information about actions needed for further execution and next, to perform such actions in order to change the system and its actions, to adapt and perceive changing goals in a changing environment. This ability manifests on a changing border of the system and its environment. For such ability, the system must be able to perform some information actions to check states of the system and its environment, to learn, and produce information about the required actions. Environment changes generate this need for information actions, which are performed as causal for non-information actions followed by information actions. Thus, an environment change makes IT usage necessary, which, in turn, causes IT effects and IT effects produce dynamic capability effects on the changing border of a system and an environment. This kind of an IT is always required for dynamic capability effects to be realized and environment change is required to generate a need for such IT usage for creation of dynamic capability effects. Therefore, when one talk about the operational properties of IT usage or dynamic capabilities one shall estimate the role of IT in creation of system dynamic capability effects in response to changing environment.

To describe the relations between information and non-information actions and dynamic capability effects during system functioning, concepts and principles (concept model) of IT application for dynamic capabilities effects realization are suggested. Through applying these concepts and principles, author reveal general patterns of IT application. The suggested conceptual model is provided for transition first to graph-theoretical, set-theoretical and then to functional model (to estimate probabilistic measure [9]) of IT usage for dynamic capabilities effects. It is based on patterns of non-information effects development with the use of information obtained.

General concepts and principles of IT usage for dynamic capabilities effects creation, or IT-enabled dynamic capabilities [28], are described in section two; modeling concepts, principles and patterns of such capabilities are described in section three. Examples of schemas for operational properties indicators estimation, including dynamic capabilities indicators, are introduced in section four. In section five, prototypes of software package for estimation of IT enabled dynamic capabilities indicators are described.

II. General concepts and principles of IT usage for dynamic capabilities effects

I shall describe the use of IT through a technological system example. The system is considered technological if its functioning is defined by technological documentation (e.g. manuals, descriptions, instructions) in the system. These include, for instance, systems that function to enforce manufacturing of unique products (e.g. in aerospace industry), and systems for implementation of state projects and targeted programs. General concepts required for development of IT application models in the context of technological systems dynamic capabilities include: IT, IT application, information, information use, system, system operation, purposeful changes in system operation, goal, outline of changes in system operation, benefit, technological information operation, technological non-information operation, system operation effects, effects of transition processes during functioning. Concepts are linked in a schema of purposeful changes of technological systems through the application of IT (Fig. 1). IT effects [3] are manifested in a technological system conditioned by changes in operation (for example, by transition processes from reaching one goal to reaching another). This change in operation becomes apparent in changes in non-information actions (their composition, properties and sequence). The changes in non-information actions are caused by the results of the information actions. Implementation of information actions is governed by necessary consideration of the environment impact on a technological system. As a result of the series of changes, personnel using a technological system obtain the effects different from those that would appear, should there have been no changes, that is, not considering the environment impact or the new technological system, conditioned by this impact. The operation implementation with new chosen parameters is explained by technological information operations implemented to take into account the impact of the environment on a technological system. These technological information operations provide for selection of next technological operations with better parameters (in effected conditions) depending on the changes in the states of a technological system and its environment. Best operational effects are achieved through consideration of these changes at execution of technological information operations. The use of different types of technological operations (hereinafter referred to as “TlOp”), e.g. information, non-information, in technological system functioning depending on verified technological system states and its environment is illustrated in Fig. 1. When TlOp sequences are implemented, first technological information operations (hereinafter referred to as “TIO”) are executed. These operations estimate changed states of the environment and system elements with regard to environment impact. Further, TIO liable for changed TlOp are executed (if necessary). Their ultimate goal is to obtain information about the technological system state and its environment and what should be changed in this regard. Then, technological non-information operations (hereinafter referred to as “TNIO”) connected with information operations by cause-effect relations are executed through practical implementation. The notions of information and IT, benefits of IT, benefits of information, information and non-information actions, TlOp, TIO, TNIO and other related notions were specified in [5]. Principles of technological system research and a number of related notions were introduced in [4,5]. General OP characteristics were defined in [6]. Let us specify the notions that are used further in the context of functional modeling of a technological system.

Technological information operation is an action to be executed according to the technological documentation, the goal of which is to provide needed information (for example, instructions) to perform other actions. Technological non-information operation is an action to be taken according to the technological documentation, the goal of which is to perform an exchange of material and energy (according to the instructions obtained). Technological information operations are executed according to a certain information technology. TIO (or, as a rule, a number of TIOs) aims at obtaining (creation) and transforming the information into such a form, where it could be used by a person or technical equipment to solve a task of choosing (for instance, choosing a mode of TNIO). During implementation of TIO and TNIO sequences, depending on the occurred events and states of the system elements and environment, which were revealed as a result, different TIO are executed. Then TIOs are used for choosing various TNIO resulting in occurrence of various events and states of the system. In this regard, the system and environment states do not recur during operation in reality, and sequences of TlOp, events and states (a loop in Fig. 1) should be expanded into structured sequences of events and states (outcome tree). As a result, numerous possible state sequences are obtained. They are connected by branches (events) depending on states of a system, environment and implemented sequences of TlOp (TIO and TNIO), and the events, which are revealed during TlOp execution.

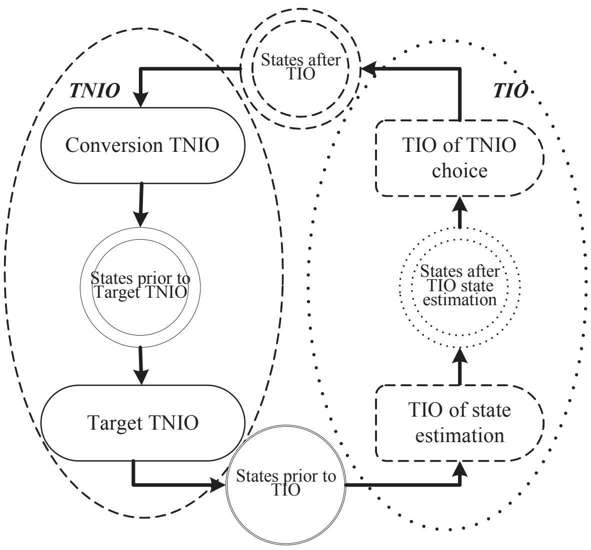

Fig. 1. A Loop of Different Types of TlOp Used during System Functioning

The system operation outcome is a sequence of conditioned states of the system and branches (events) between them caused by TlOp (both TIO and TNIO) and actions of the system environment. Let us denote a layer of possible chains of actions, events and obtained states by L_i. It depends on the environment state when i – th loop in Fig. 1 is performed. Chains of actions, events and states obtained due to sequences of such loops (L_1, L_2, L_3) are illustrated in Fig. 2. They depend on TIO and IT used for system functioning. During planning, the possible operation outcomes are reviewed, being a sequence of possible states and branches between them caused by TlOp (TIO and TNIO). Composition and characteristics of TlOp, which lead to possible operation outcomes, change as a result of TIO. TIO leads to various sequences of random events and states revealed as a result of changes in the environment states. These events and states form possible outcomes. Each possible outcome, except for various possibility measures of its implementation (depending on states of the system and environment, and implemented TlOp) complies with different effects (results with specified requirements) of operation and different operation efficiency.

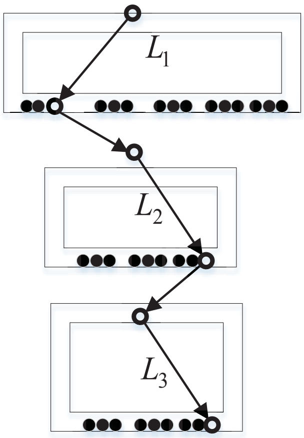

Fig. 2. Sequences of States in System Functioning

Operational properties of technological systems, namely system potential [4] or dynamic capability of such systems (with regard to IT application), describe future system parameters associated with its operational efficiency in changing environment. This property should be estimated based on the modeling of all possible future operation outcomes under all possible environment changes. System potential, or dynamic capability of a technological system, is a property that indicates whether a technological system is suitable to reach changing goals (actual and possible) in a changing environment. It would be rational to use the difference between technological systems with applied "new" and "old" IT as an indicator of IT enabled dynamic capability of the "new" IT compared with the one used previously.

Thus, this indicator can be used as an analytical estimation of an operational property indicator of IT usage. This indicator should be estimated based on analytic models developed through description of laws and manifestation patterns of effects, as a result of execution of TIO and TNIO sequences of various characteristics at different technological systems operation outcomes.

Use cases of such indicators includes choosing IT and TIO characteristics for optimal implementation of new IT, such as usage of distributed ledger technologies for various business processes, robotic technological process automation.

III. Concepts, principles and patterns of operational properties of IT usage modeling

Concepts applied during development of system functioning models with regard to transition actions of the system improvement, and principles applied during conceptual and formal modeling of technological system were defined in [5-8]. Let us consider general concepts, which require interpretation due to a suggested concept of IT application in the context of technological system functioning. Simplex of TlOp (simplex) is a sequence of the initial TIO (TIO required to initiate TNIO), TNIO and final TIO (TIO required to terminate TNIO). Reduced simplex (hereinafter referred to as "RS") is a simplex containing zero TNIO. There are several types of RS depending on the type of a state evaluation task they solve: if RS solves a task of general system state evaluation at the moment (to the moment) then it is type one RS. If RS solves a problem of state evaluation of one or several sites (i.e. "workplaces" that constitute the system as a whole) at the moment (to the moment) then it is type two RS. Depending on their specifics, different RS should be executed to evaluate the states of TS and environment as a result of execution of simplexes.

This rule is fixed by a principle of simplex linking through RS implementation. These RS are implemented differently depending on the results of execution of prior TlOp and environment states. RS targeted result consists in chosen composition and prescripts of further actions. This result should be used in consequent simplexes to achieve targeted results of TNIO. While different sequences of simplexes and RS should be executed differently (depending on various recorded states of the system and environment), different states are implemented as a result. Afterwards these states could lead to implementation of various simplexes and TS transition into next states, as a result. Creation of these sequences is given according to a principle of functional dependency of the system operation outcome from simplexes and states of the system and its environment. Nodes of an outcome tree are possible states achieved as a result of TlOp (TIO and TNIO by selected means), and tree edges stemming from the parent node are possible outcomes (transitions between states) resulting in TlOp implementation.

Such sequences of states and operations are then parameterized with possibilities of outcomes. A fragment of such parameterized graph-theoretical model is shown in Fig. 3. To keep the size of a model smaller, a principle of aggregation is suggested. It consists in aggregation of states achieved up to the moment of completion of certain types of reduced simplexes. Aggregation schema Σ_2 applied to a reduced simplex of type 2 is shown in Fig. 3. Tree branching at the system operation complies with one of the possible events chain if it is actualized. If the system state during operation is calculated on the basis of the state of several workplaces and several respective RS of type 2, the sub trees complying with possible states of workplaces and their combinations are connected into the branch. The outcome tree corresponds with all possible TS operation outcomes. Composition and characteristics of outcomes and the outcome tree depend on the TlOp composition and characteristics, and, as a result, on the used IT. In particular, possibility measure of possible outcome implementation (possibility measure of the outcome to become reality) depends on the composition and characteristics of TlOp (TIO and TNIO) and on the state of the environment during operation. Operation effects achieved as a consequence of certain outcome implementation depend on the composition and characteristics of TlOp (and the IT) and on the states of environment at operation. Knowing the possible outcome and characteristics of the effects, providing this outcome is real, one could calculate the system dynamic capabilities indicator (system potential).

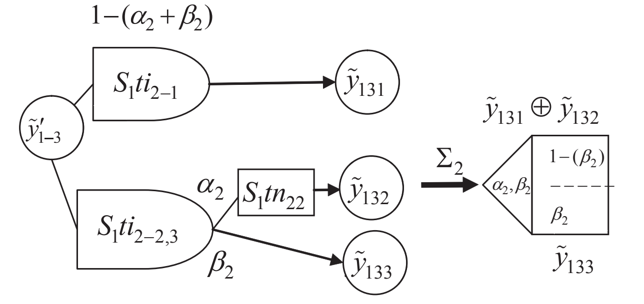

Fig. 3. Parameterized Model Fragment and Its Aggregation

Aggregation schema Σ_2 applied for reduced simplex of type 1 is shown in Fig. 3. Nested aggregation schema Σ_1 applied to the results of schema Σ_2 application is shown in Fig. 4.

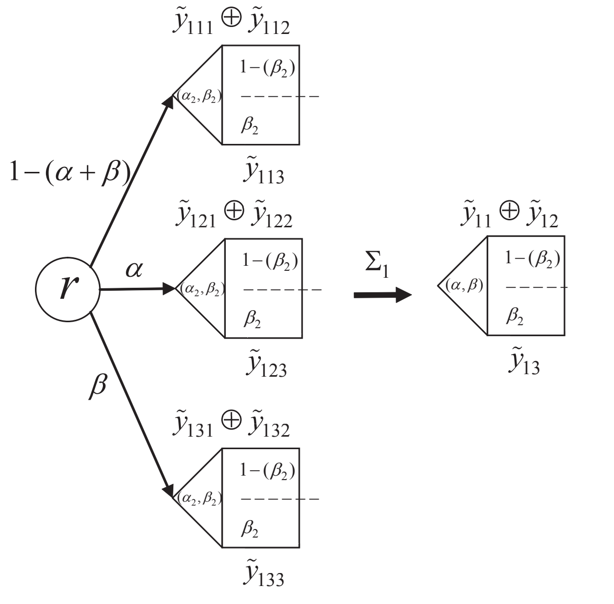

Fig. 4. Nested Aggregation of Parameterized Model

The role of IT usage during system functioning effects formation is illustrated in Fig. 5 through the example of RS type 1 execution and a resulting schema. $RS1_1$ to $RS1_N$ sequence is considered (upper part of figure). Each RS type 1 in a sequence checks results of operations fulfillment (from RS type 2 on corresponding workplaces, where TIO has just finished checking the operations results). Following this check, an initial state of the corresponding RS type 1 is formed based on the RS type 2 effects. Such effects are information effects of a check. Next, based on this initial information state (of the check type) RS type 1 is performed and a goal state of RS type 1 is formed due to the RS type 1 fulfillment. This goal state formed due to RS type 1 precepts information effects. Precepts obtained as a result of RS type 1 depend on the check results, IT and IT operations used to perform RS type 1. They use the results of the check, technological data and environment check data to calculate effects compliance during RS type 1 fulfillment. Based on an indicator of such compliance actual percepts are obtained. The percepts obtained during RS type 1 are then sent to RS type 2 and next to simplexes in order to start the corresponding TlOp.

As a result, TlOp workflow is changing. Therefore, producing of non-information effects is changing as a result. Thus, information effects appear because of possible changes (during checks) and then they cause changes in non-information effects through changed precepts. Once TlOp is finished, the corresponding TIO initiates the process of verification again. This cycle repeats again for $RS1_2$ and further until the last $RS1_N$ is fulfilled. To measure the results of system functioning with regard to IT usage, appropriate system dynamic capability indicators shall be suggested.

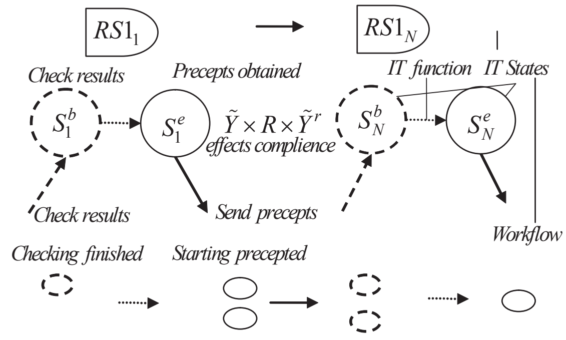

Fig. 5. Role of IT During System Functioning Effects Formation

An estimation of system dynamic capabilities indicators is proposed in the form of probabilistic or other correspondence measure estimation of effects predicted as required values. Such estimation is the basis of implementation of the loop of targeted changes. Estimation can be conducted either on the basis of analytical mathematical methods and models or through the generalization of one's experiences (heuristically). Models of states changes shown are linked together by graph theoretic nested tree model illustrated in Fig. 6.

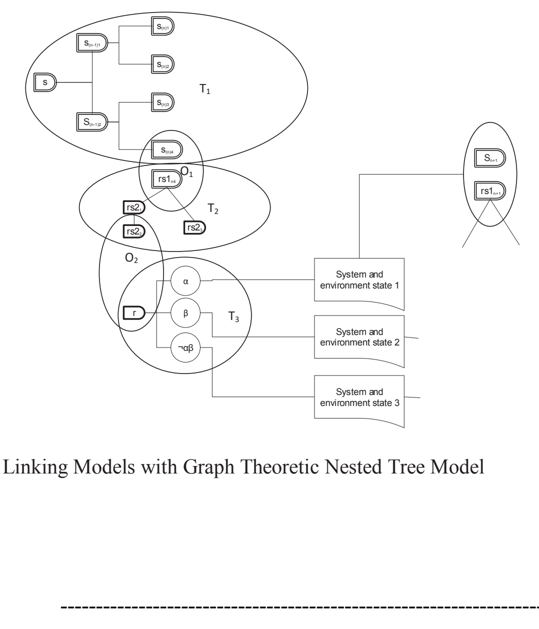

Fig. 6 Linking Models with Graph Theoretic Nested Tree Model

In this model, some trees that model the states of a system and its environment are linked together by nested trees T_1, T_2, T_3 .

The difference between the solutions of the considered problems based on an analytical evaluation of system dynamic capability indicators and those achieved heuristically consists in possibilities to built predictive mathematical models and to automate solutions of practical problems as mathematical problems of analytical estimation, analysis and synthesis (for example, as operation research or mathematical programming problems). Specifically, taking into account transition actions during the process of improvement of a system and its functioning, and the role of various information actions and IT technologies in this process, formulation and solution of practical problems of improvement of a system and its functioning, and IT usage as mathematical problems of system dynamic capability (potential) research become possible.

Typical system operational properties indicators, including schemas for estimation of dynamic capability indicators allowing for such research are described below.

IV. SCHEMAS FOR OPERATIONAL PROPERTIES ESTIMATION

A. A sequence of three schemas for OP indicators estimation

Let us introduce a sequence of three schemas for the estimation of system operational properties indicators, including operational property of system dynamic capability (OP). Each successive schema uses a previous one. The first, basic schema is aimed at the estimation of operation efficiency in the case were the goal of functioning does not change and system does not improve during functioning. According to the estimation following this schema, it is assumed that the decision made using an IT to improve actions during their implementation, along with the system and processes of their execution according to the goal perceived was made before functioning started. Thus, the functioning is not interrupted.

The second schema generalizes the first one to account for a plurality of possible functioning in order to reach different goals under different conditions of an environment. According to this schema, it is assumed that the possible improvements are determined in advance (with use of IT), there are multiple possible goals to achieve and certain transition improvement actions are determined (with IT use) before the start of operation.

Finally, the third schema summarizes the first two schemas in order to account for both possible transition actions selected before their application to achieve different goals and targeted transition actions selected and implemented during operation, depending on the prevailing conditions.

B. Basic schema for operational properties estimation

Let us introduce I_p as a value of measure on the set, p as a function defining this measure, Y-set of vectors of random characteristics of operational effects (characteristics of operational quality), R-set of the required component-wise relations between random values of effects of characteristics and their desired values, Y^r-set of vectors of characteristics values of the required functioning effects. Then, the estimation of I_p is set by the following schema:

$$p: Y \times R \times Y^r \rightarrow [0,1] \quad (1)$$

$$I_p = p(Y \times R \times Y^r) \quad (2)$$

I_p is the measure value of the possibility [29] indicating that the predicate in parentheses takes the value "true" or the value indicating that the random event corresponding to the predicate occurs. Thus, if $Y=y$ is a random variable defined on the axis of real numbers, R is the ratio "at most", y^r is the point at the real axis (the required limit value of the random variable y), then $p=F_y\ (x)$, $I_p=F_y\ (y^r)$ is the value of the function of distribution of the random variable y in the point y^r.

C. Schema of operational properties estimation given that the transition processes are known during goal changing

Let us introduce $Y^r(t)$ as a random process modeling any possible accidental changes of Y^r in time (for example, due to changes in goal of system operation), t_0 as the starting point of the system operation, $Y^r=Y^r(t_0)$, T as a goal duration of the system operation. Then, the schema of the estimation of operational properties would be of the following type:

$$p: Y \times R \times Y^r(t),\ t_0,\ T \rightarrow [0,1] \quad (3)$$

$$I_{op} = p\ (Y \times R \times Y^r(t)),\ t \in [t_0, T] \quad (4)$$

I_{op} is the measuring value of the possibility that the predicted values of system operation effects (under varying goals) comply with the desired values of effects in the corresponding way. Whereby:

$$I_{op}\ (t_0) = I_p \quad (5)$$

If the requirements are changed and corresponding changes can be determined before operation, it is necessary to plan a transition process (with the characteristic u) from one operation to another under the stated changes, which makes it possible to estimate the value of I_{op}. However, if characteristics u of transition actions cannot be determined in advance and depends upon the state of the system and the environment during operation, it is necessary to use the third estimation schema.

According to this schema, all transition actions are a sequence of changing actions depending upon the state s during the operation of the system and the environment. States and transition actions are determined with use of IT.

D. Schema of the estimation of OP given the processes of improvement depend on system states during its operation

The last OP research schema is used when sequences of transition actions described with the characteristics u are implemented during operation, depending on achieved states of the system and the environment s. Transition actions are selected in accordance with IT applied to change "goal" functioning and their prescripts. Transition actions effects manifested through workflow actions and prescripts selected for the "goal" functioning. "Goal" functioning is one transition fulfilled for.

At the same time, resources are spent for transition fulfillment. The constraint of $u\ (s)$ describes the characteristics of transition actions necessary for calculating transition effects and then, as a result, effects of the "goal" functioning. Characteristics of the sequence $uc=u_1... u_n$ of such transition actions out of the possible sequences U_c depend on the characteristics of the manifested sequences of states $sc=s_1... s_n$ out of possible sequences of states Sc. The schema of estimation of OP in this case is as follows:

$$p: Y (Uc (Sc), t) \times R \times Y^{*} (t) \rightarrow [0,1] \quad (6)$$

$$I_{ops=}p (Y (Uc (Sc), t) \times R \times Y^{*} (t), t \in [t_0, T], sc \in Sc, uc \in Uc \quad (7)$$

I_{ops} is the measuring value of the possibility that the predicted values of TSF effects (under varying goals and transition actions) meet the required values of effects in a corresponding way, in accordance with these goals. Possible sequences sc and uc depend on applied IT. At this, the previous estimation schema complies with $Uc=u$:

$$I_{ops} (Uc (Sc), t)=I_{ops} (u, t)=I_{op} (t) \quad (8)$$

V. Prototypes of Software for Estimation of Operational Property Indicators of IT Usage

Modeling of operational properties of IT usage requires creation of multiple system functioning models under multiple scenarios of environment functioning. Multiple models creation may be quite complex. Therefore, I propose to use diagrammatic means. Graph theoretic, diagrammatic models transformed into parametric through adding parameters and variables to graph theoretic models are built. Database of parameters and variables restrictions is used for this purpose. In the example considered, diagrammatic models were created with ARIS toolset modernized so as to use nested diagrams to reflect some relations through graph theoretic models.

Next, parameterized models are transformed into functional through adding formulas to ARIS models elements. Then, nested diagrammatic models are transformed into Microsoft Excel spreadsheets shown below.

Resulting spreadsheets constitute a program model of an IT enabled system dynamic capability estimation.

Examples of diagrammatic models are shown below. They are based on some common sub-process models (Fig. 7).

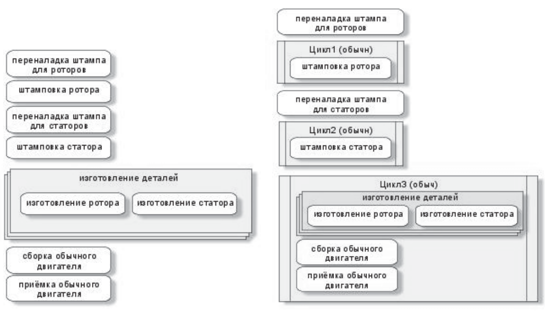

Fig. 7. Sub-Processes Used by Diagrammatic ARIS Models

Simplest models available were used. For example, only four scenarios of environment functioning are possible and there are four changing goals as a result. Diagrammatic model of functioning could be built for each goal. The use of an IT is modeled with relevant IT operations, resulting in a change of the course of functioning. Such operations require additional resources and time when a functioning goal is altered due to a change of environment (Fig. 8, Fig. 9).

Different model versions are considered. Version 1 (Fig. 8) differs from version 2 (Fig. 9) by respective TIO characteristics according to different IT used.

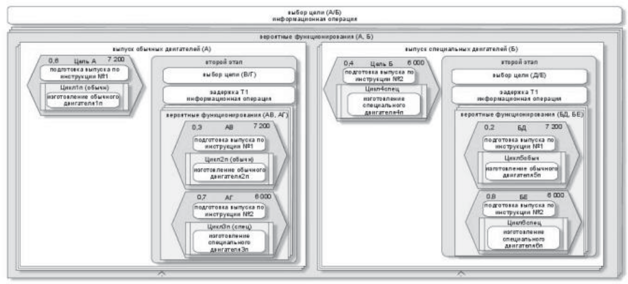

Fig. 8. Diagrammatic ARIS Model Version 1 to Estimate System Dynamic Capability Indicator of IT Usage

Next, an indicator of IT enabled dynamic capability is estimated as a probabilistic mix of system functioning efficiency with IT used for functioning changes according to four different scenarios of functioning change.

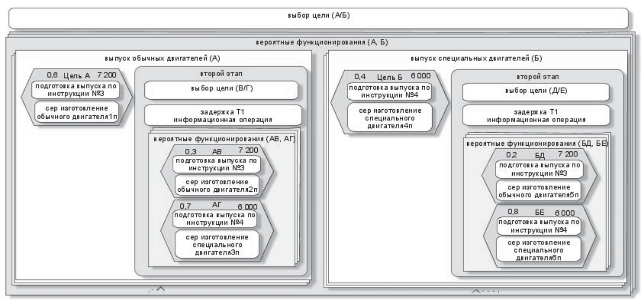

Fig. 9. Diagrammatic ARIS Model Version 2 to Estimate Operational Properties of IT Usage and Dynamic Capability Indicators

Resulting Microsoft Excel table example (Fig. 10) constitutes a program model for estimation of operational properties of IT usage and corresponding dynamic capability indicators. It was obtained automatically, using model-driven meta-modeling [30-35] and ARIS possibilities to generate a program code.

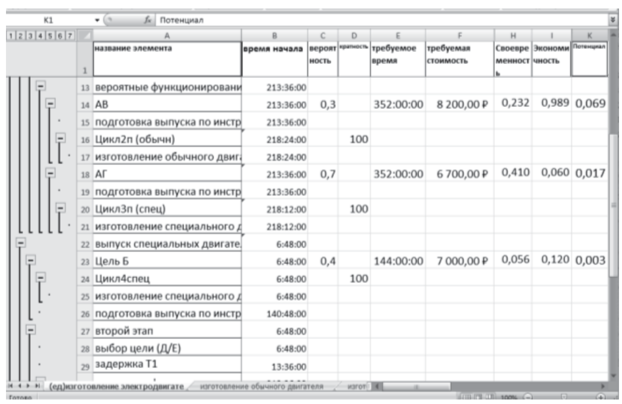

Fig. 10. Program Model to Estimate Operational Properties of IT Usage and Dynamic Capability Indicators

VI. Conclusion

The obtained results allow for evaluation of predicted values of systems operational properties of IT usage and dynamic capabilities indicators. They could help to analytically estimate IT operational properties, dynamic capabilities properties and other operational properties related to IT usage depending on variables and options in tasks solved. This could lead to a solution of contemporary problems of a research dedicated to the operational properties of IT usage, system dynamic capabilities as well as other operational properties using predictive analytical mathematical models and mathematical methods of research problem solving, for example, using mathematical programming and operation research mathematical models and methods. Examples of problems possible to decide include choosing IT and TIO characteristics for optimal implementation of new IT, such as optimal usage of distributed ledger technologies for business processes, robotic technological process automation optimization, and cyber-physical systems characteristics choosing.

Acknowledgment

Performed under support of the RFBR grant No. 16-08-00953

References

[1] D. Teece, G. Pisano, A. Shuen, "Dynamics capabilities and strategic management", *Strategic Management Journal,* Vol. 18 No. 7, pp. 509-533. (1997).
[2] C. Wang, P. Ahmed. "Dynamic capabilities: a review and research agenda", *International Journal of Management Reviews*, Vol. 9 No. 1, pp. 31-51. (2007).
[3] D. Teece. "Explicating dynamic capabilities: the nature and microfundations of (sustainable) enterprise performance", Strategic Management Journal, Vol. 28 No. 13, pp. 1319-1350. (2007).
[4] A. Geyda, I. Lysenko. "Tasks of study of the potential of socio-economic systems", *SPIIRAN proceedings*, 10, pp 63–84. (2009).
[5] A. Geyda and I. Lysenko. "Operational properties of agile systems and their functioning investigation problems: conceptual aspects" *J. of Appl. Informatics,* 12, 93-106 (2017)
[6] A. Geyda and I. Lysenko. "Schemas for the analytical estimation of the operational properties of agile systems", *SHS Web Conf.* 35, (2017)
[7] A. Geyda, I. Lysenko, and R. Yusupov. "Main concepts and principles for information technologies operational properties research", *SPIIRAN proceedings,* 42, 5–36 (2015)
[8] A. Geyda, Z. Ismailova, I. Klitny, I. Lysenko. "Research problems in operating and exchange properties of systems", *SPIIRAN proceedings*, 35, pp. 136–160 (2014)
[9] J.Taylor. "Decision Management Systems: A Practical Guide to Using Business Rules and Predictive Analytics", IBM Press, 312 P. (2011)
[10] T. Kendrick. "How to Manage Complex Programs", AMACOM, (2016)
[11] T. Dinsmore. "Disruptive Analytics: Charting Your Strategy for Next-Generation Business Analytics", Apress, 276 P. (2016)
[12] A. Downey. "Think Complexity. Complexity Science and Computational Modeling", *O'Reilly Media,* (March 2012)
[13] G. Cokins. "Performance Management: Myth or Reality?", Performance Management: Integrating Strategy Execution, Methodologies, Risk, and Analytics, Wiley, 274 P. (2009)
[14] G. Cokins. "Why is Modeling Foundational to Performance Management?", *Dashboard inside newsletter*, (March 2009)
[15] C. Hood, S.Wiedemann, S.Fichtinger, U. Pautz. "Requirements Management. The Interface between Requirements Development and All Other Systems Engineering Processes", Springer, 275 p. (2008)
[16] D. Hybertson. "Model-oriented systems engineering science: a unifying framework for traditional and complex systems", AUERBACH, 379 P. (2009)
[17] E. Aslaksen. "The System Concept and Its Application to Engineering", 266 P. (2013)
[18] E. Aslaksen. "Designing Complex Systems. Foundations of design in the functional domain", CRC Press, Complex and enterprise Systems Engineering series, Auerbach, 176 P. (2008)
[19] F. Franceschini, M. Galetto, D. Maisano. "Management by Measurement: Designing Key Indicators and Performance Measurement Systems", Springer, 242 P. (2007)
[20] G. Roedler, R. Schimmoller, D. Rhodes, C. Jones (Eds). "Systems engineering leading indicators guide", INCOSE Technical Product, INCOSE-TP-2005-001-03. Version 2.0, Massachusetts Institute of Technology, INCOSE, PSM, 146 P. (2010)
[21] G.Tanaka. "Digital Deflation: The Productivity Revolution and How It Will Ignite the Economy", McGraw-Hill, 418 P. (2003)
[22] Guide to the System Engineering Body of Knowledge, SEBoK v. 1.3.1. INCOSE, (2014)
[23] J.J. Simpson, M.J. Simpson. "Formal Systems Concepts. Formal, theoretical aspects of systems engineering. Comments on "Principles of Complex Systems for Systems Engineering", *Systems Engineering*. Vol. 13, No 2, p.p. 204–207 (2010)
[24] J. Elm, D. Goldenson, Kh. Emam, N. Donatelli, A. Neisa. "A Survey of Systems Engineering Effectiveness - Initial Results (with detailed survey response data)", NDIA SE Effectiveness Committee, Special report CMU/SEI-2008-SR-034, Acquisition Support Program. Carnegie-Mellon University, NDIA, 288 P. (2008)
[25] N. Patel. "Organization and Systems Design. Theory of Deferred Action", Palgrave McMillan, NY. 288 P. (2006)
[26] R. Stevens. "Engineering Mega-Systems: The Challenge of Systems Engineering in the Information Age", CRC Press, Complex and enterprise Systems Engineering series, 256 P. (2011)
[27] P. Mikalef, A. Pateli. "Information technology-enabled dynamic capabilities and their indirect effect on competitive performance: Findings from PLS-SEM and fsQCA," *Journal of Business Research,* Elsevier, vol. 70(C), p.p.1-16. (2017)
[28] P. Taticchi. "Business Performance Measurement and Management: New Contexts, Themes and Challenges", Springer Science & Business Media, 376 P. (2010)
[29] E. Zio and N. Pedroni. "Literature review of methods for representing uncertainty". (Toulouse, France: FONCSI), 61 P. (2013)
[30] E. Lee. "The Past, Present and Future of Cyber-Physical Systems: A Focus on Models", *Sensors,* Vol. 15. p.p. 4837-4869. (2015)
[31] B. Henderson-Sellers. "On the Mathematics of Modelling, Metamodelling, Ontologies and Modelling Languages", Springer Briefs in Computer Science, Springer, 118 P. (2012)
[32] T. Kendrick. "How to Manage Complex Programs", AMACOM, USA, 336 P. (2016)
[33] T. Debevoise, J. Taylor. "The MicroGuide to Process and Decision Modeling in BPMN/DMN: Building More Effective Processes by Integrating Process Modeling with Decision Modeling", CreateSpace Independent Publishing Platform, USA. 252 P. (2014)
[34] M. Lankhorst. "Enterprise Architecture at Work: Modelling, Communication and Analysis", The Enterprise Engineering Series, Springer. 352 P. (2013)
[35] A. Kleppe. "Software Language Engineering: Creating Domain-Specific Languages Using Metamodels", Addison-Wesley Professional, 240 P. (2008)