DOI: https://doi.org/10.15276/hait.02.2021.3
UDC 004.738 + 004.94

Behavioral hidden testing of distributed information systems taking into account of energy

Oleksandr M. Martynyuk[1)]
ORCID: https://orcid.org/0000-0003-2366-1920; anmartynyuk@ukr.net
Oleksandr V. Drozd[1)]
ORCID: https://orcid.org/0000-0001-8305-2217; drozd@ukr.net
Sergiy A. Nesterenko[1)]
ORCID: https://orcid.org/0000-0002-3757-6594; sa_nesterenko@ukr.net
Vadym Yu. Skobtsov[2)]
ORCID: https://orcid.org/0000-0002-8546-0430; vasko_vasko@mail.ru
Thuong Van Bui [1)]
ORCID: https://orcid.org/0000-0002-9160-5982; govarava@gmail.com
[1] Odesa National Polytechnic University. 1, Shevchenko Ave. Odesa, 65044, Ukraine
[2)] United Institute of Informatics Problems of National Academy of Sciences. 7, Surganova Str. Minsk, 220012, Belarus

ABSTRACT

The introduction of new energy-consuming properties for positions and transitions into the checked properties of the extended reference Petri net, for which the deviations of the tested Petri net are determined and a testing model is developed, provides new diagnostic possibilities. Keeping the class of checked properties in the composition of deviations of incidence relations, correspondences and marking functions of positions and transitions for the checked and reference Petri nets, the new properties make it possible to record the appearance of critical temperature regimes that are a consequence of errors or directly leading to their appearance. This versatility of testing helps to increase its completeness, accuracy and efficiency. The energy-heavy testing model is based on verification of incidence, correspondence, and markup functions. Checking the markup functions when generating events in positions, performing actions in transitions, as well as the proposed checking of the energy consumption indicators accumulated in the monitor tokens, is performed when checking the incidence, correspondences. The features of the testing model include the input of generalized energy-loaded Petri nets recorders, accumulating information about energy consumption in the behavior of positions/transitions, topological components and subnets, the entire Petri net in the process of its functioning. The testing model is also distinguished by the recognition of the reference energy-loaded behavior when checking the Petri net based on behavioral identification and coincidence of subsets of positions/transitions, the determination of behavior, the use of check primitives and transactions. The behavioral testing model defines the formal conditions for behavioral testing procedures, including the analysis of the correctness of energy consumption. The dimensionality of the testing model was estimated using the representation of Petri net graphs, special graphs of attainable states, including Rabin-Scott automata, using list structures. These estimates define the limits of applicability of the formal testing model.

Keywords: Information System; Energy Behavior; Behavioral Testing; Petri Net; Identifier; Check Primitive

***For citation*: Martynyuk O. N., Drozd O. V., Nesterenko S. A., Skobtsov V. Yu., Bui Van Thuong. Behavioral Hidden Testing of Distributed Information Systems Taking Into Account of Energy. *Herald of Advanced Information Technology.* 2021; Vol.4 No.2: 135–145. DOI: https://doi.org/10.15276/hait.02.2021.3**

INTRODUCTION

For rapidly becoming more complex promising distributed information systems (DIS) [1], the properties of autonomy with an internal, hidden for control and monitoring nature of work, mobility and intelligence of components, dynamic cooperativity of their interaction become inherent [2, 3]. The explosive development of nanoelectronics makes it possible to achieve a significant reduction [4] in energy consumption for DIS hardware [5, 6]. The downside of the penetration of computer technologies into all spheres of human activity is the increase in the criticality of tasks [7] solved with their help, which significantly increases the requirements for the reliability of their functioning [8, 9]. The importance and absolute necessity of the design and operational efficiency of the DIS becomes obvious, in particular, achieved through the analysis of correctness [10], verification [11], testing and diagnostics [11, 12] of DIS projects and implementations, checking, in particular, the optimality values of their energy consumption, including in the modes of hidden functioning.

Numerous researches of efficiency [13, 14], reliability of work [15, 16], energy saving [17, 18] for existing technologies demonstrate the greatest influence [19] on the energy consumption of hardware DIS dynamic modes, primarily switching during operation, represented in electrical [18] and logic circuits [19, 20], logic [21, 22], schematic [23] and special structures [24], FPGA [25, 26].

Analysis of models and experimental results shows a direct polynomial dependence of the number

© Martynyuk O., Drozd O., Nesterenko S., Skobtsov V., Bui Thuong Van, 2021
This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/deed.uk)

of switching's represented at the logical and electrical levels of hardware of various computer systems, and estimates of the generalized power consumption determined for the system, structural, functional specifications of objects and processes of these systems [27, 28]. This, in particular, becomes obvious in connection with the direct connection between the operators of programming languages, the vertices of algorithms and transitions of models of the automaton class on the one hand and hardware implementations of bitwise arithmetic and microprogram instructions at the logical level on the other hand. Thus, it is obvious that it is possible to synthesize and verify design solutions and DIS implementation at the systemic, structural and functional level of representing objects [29, 30] and processes [31]. In particular, it is advisable to use for these purposes effective Petri nets [32], modern complex technologies [33, 34], showing hidden events and actions, which include the analysis of energy consumption, adding models and methods of operational current and inertial temperature observation [24, 25] and functional-temporary transactions [35, 36].

At the same time, the analysis of existing works, which consider the issues of control and monitoring of energy consumption, demonstrates the predominance of research in relation to hardware [18], implementation technologies [19, 20] and much less often – consideration of energy consumption at the system [36, 37], functional [38, 39], behavioral [40] levels.

As a result, it becomes possible to draw a conclusion about the relevance of the researhes of DIS behavioral check models with checking the correctness of energy consumption, supplemented by an analysis of internal, realized and manifested events and actions.

PURPOSE, PROBLEM STATEMENT

The purpose of this work is to determine the conditions for increasing the completeness and accuracy in the behavioral testing of DIS, extended by checking energy consumption, which is performed in experiments for checking and recognizing the functioning of extended Petri nets (EPNs) with fragments hidden for control and observation.

This goal determined the task of building a DIS behavioral check model based on the EPN, which has the features of accounting and subsequent analyzing energy consumption using chips with a registration mechanism, as well as hidden behavior. The proposed model makes it possible to find the conditions for monitoring the functioning of real DIS, taking into account its energy consumption when compared with the reference functioning.

DEVELOPMENT OF AN INPUT MODEL WITH ENERGY CONSUMPTION AND LATENT BEHAVIOR

The solution to this problem led to the development of an input for testing, a behavioral EPN model for DIS with fixing energy consumption [41], which shows that modeling EPN using it, representing asynchronous-event parallel DIS processes, makes it possible to record and analyze energy consumption in the system of parallel streams of chip-monitors at three levels: elementary (positions, transitions), component (subnets), general (EPN). When reaching the end positions or when cycling back to the starting positions in the case of reactive DIS, the chip-monitor system allows you to obtain the end values of energy consumption for elements, components and the entire EPN.

The developed model, keeping identifiers, check primitives and fragments, controlled and observable alphabets, chips-monitors of energy consumption [42], defines the external behavior of the EPN, its events and actions in the symbols of the input and output alphabets, as subsets of controlled events and observed actions from full sets of all events and actions. The development of the model also provides for the analysis of hidden, implicitly controlled events and implicitly observed actions and the corresponding energy consumption.

Thus, the input for the testing of the EPN $S(f)$ with the presented modifications in the latent behavioral properties and energy costs is defined as:

$$S(f)=(P, T, Ev, Ac, X, Y, Ep, Et, F, S, M_0), \quad (1)$$

where:

– $P, T, X, Y, Ep, Et, F, S, M_0$ – defined in [41], primary sets of positions, transitions, input and output alphabets, energy consumption of positions and transitions, incidence relations of positions and transitions, position correspondence, transitions, events, actions, energy costs, markings, these sets either have not changed, or have undergone some extensions and clarifications presented below;

– $Ev=\{ev, ev_2, \dots ev_{me}\}$, $Ac=\{ac_1, ac_2, \dots ac_{la}\}$ – complete sets of all events for positions and actions for transitions, respectively;

$$X=\{x_1, x_2, \dots x_{mx}, @\}\subseteq Ev, \quad Y=\{y_1, y_2, \dots y_{ly},$$

– $@\}\subseteq Ac$ – subsets of respectively controlled input events for positions and observed output actions for transitions, supplemented with the @ symbol – uncertainty symbol for a hidden (unknown) uncontrolled event, or unobservable action, or their bipartite chain with alternating hidden events and actions, the input of the chain is due to the asynchronous behavior of the EPN;

– $Ep=\{ep_1, ep_2,\dots ep_{np}\}$ is the set of boundary energy costs for the formation of events for positions from P, represented in the general case in the form of triplets $ep_{np}=(epi_{np}, ep^{\circ}_{np}, ept_{np})$ with components - upper boundary integer values of the current variables epi_{np},, temperature ep°_{np}, time ept_{np};

– $Et=\{et_1, et_2,\dots et_{nt}\}$ - the set of boundary energy costs of performing actions for transitions from T, also represented in the general case as triplets $et_{nt}=(eti_{nt}, et^{\circ}_{nt}, ett_{nt})$ with components - upper boundary integer values of alternating currents eti_{nt}, temperature et°_{nt}, time ett_{nt};

– $F:(B(P\times Ev\times Ep\times Qp)\rightarrow T)\cup((T\times Ac\times Et)\rightarrow B(P\times Qt))$ – extended conditional incidence relation for subsets of positions from the Boolean $\{p_1,\dots,p_{ip}\}\in B(P)$ and transitions from T, that is, for some transition $t\in T$ and, first, a subset of positions $\{p_1,\dots,p_{ip}\}\in B(P)$,, corresponding according to S for each position from $\{p_1,\dots,p_{ip}\}$ of twos of events and energy costs $\{(ev_1,ep_1),\dots, (ev_{ip},ep_{ip})\}\in B(Ev\times Ep)$, as well as multiplicities from $\{qp_1,\dots,qp_{ip}\}\subset Qp\subset N$ of incidence arcs for pairs “position-transition” $\{(p_1,t),\dots,(p_{ip},t)\}$ triples of the form $\{(\{(ev_1,ep_1,qp_1),\dots, (ev_{ip},ep_{ip},qp_{ip})\}\in B(Ev\times Ep\times Qp)$ for which $t\in F(\{(p_1,ev_1,ep_1,qp_1),\dots, (p_{ip},ev_{ip},ep_{ip},qp_{ip})\})$, secondly, subsets of positions $\{p_1',\dots,p_{ip}'\}\in B(P)$ corresponding according to S for some transition $t\in T$ of two actions and energy consumption $(ac,et)\in Ac\times Et$, a triplet $(t,ac,et)\in T\times Ac\times Et$ is formed, which, taking into account the multiplicities from $\{qt_1,\dots,qt_{it}\}\subset Qt\subset N$ incidence arcs for “transition-position” pairs $\{(t,p_1'),\dots,(t,p_{ip}')\}$ defines a complex two of the form $\{(t,ac,et),\{(p_1',qt_1),\dots,(p_{ip}',qt_{ip})\}\}\in F(t,ac,et)$;

– $S:(P\rightarrow Ev\times Ep)\cup(T\rightarrow Ac\times Et)$ – correspondence to the positions and transitions of its own internal events stored from the previous generation of events and stored from the previous execution of actions, extended by the values of energy consumption, included in F, that is, for some $t\in T$, $p\in P$, $ev\in Ev$, $ep\in Ep$, $ac\in Ac$, $et\in Et$, $(ev, ep)=S(p)$ and $(ac, et)=S(t)$ are executed;

– $M_0:P\rightarrow N$ – initial marking of positions taking into account the initial energy consumption of initialization, $M:P\rightarrow N$ - current marking of positions taking into account the current accumulated energy consumption, that is, for $p\in P$, $m_p\in N$, is executed $m_p=M(p)$, the readiness-excitation condition for some transition $t\in T$ has the form:

$$\begin{gathered}\forall(p,ev,ep,qp)\in F^{-1}(t)[M(p)=(p,ev',ep',qp')\&ev=\\=ev'\&ep\geq ep'\&qp\leq qp'],\end{gathered}\tag{2}$$

and the activity-triggering for some transition $t\in T$ with the receipt of a new marking M'' when the given condition of readiness-excitation is fulfilled is defined as:

$$\begin{gathered}(\forall(p,ev,ep,qp)\in F^{-1}(t)[M(p)=(p,ev',ep',qp')\&\\\&M''(p)=(p,ev'',ep'',qp'')\&ev'=ev''\&\\ep'\geq ep''\&\ \&qp''=qp'-qp])\ \&\\\&(\forall(p,qt)\in F((t,ac,et))[ac=ac''\&et\geq et''\&M(p)=\\=(p,ev',ep',qp')\&M''(p)=(p,ev'',ep'',qp'')\&ev'=\\=ev''\&ep'=ep''\&qp''=qp'+qp]).\end{gathered}\tag{3}$$

For the complete alphabets of events Ev and the labeling of energy inputs Ep in the positions P from the EPN $S(f)$, as well as for the positions P themselves, the extension to the subsets $\{ev_1,\dots,ev_{ip}\}\in B(Ev)$ and $\{ep_1,\dots,ep_{ip}\}\in B(Ep)$ due to the integration of parallel structures of positions, events, energy consumption for the input boolean $B(P)$ over positions P of the form $B(P\times Ev\times Ep)$. Integration is performed at the inputs and outputs of actions Ac in transitions T for $S(f)$. This is reflected above with respect to the incidence of F for subsets of positions $B(P)$ and transitions T.

On the basis of the expansion of the alphabets, the sets of all words of behavior W, W^{EvAc}, W^{EvEv}, W^{AcAc}, W^{AcEv} for $S(f)$ are determined, beginning and ending in different parallel events and actions. Let e be step zero, then:

$$\begin{gathered}W=W^{EvAc}\cup W^{EvEv}\cup W^{AcEv}\cup W^{AcAc},\\W^{EvAc}\subseteq(B(Ev)\times Ac)^*\cup\{e\},\\W^{EvEv}\subseteq((B(Ev)\times Ac)^*\cup\{e\})\times B(Ev),\\W^{AcAc}\subseteq Ac\times((B(Ev)\times Ac)^*\cup\{e\}),\\W^{AcEv}\subseteq Ac\times((B(Ev)\times Ac)^*\cup\{e\})\times B(Ev).\end{gathered}\tag{4}$$

In an EPN with energy consumption properties representing DIS, some of its events and actions may not be available for external control and monitoring. The subsets of controlled and observable (from the point of view of external analysis, without the @ symbol) input-output words W^{Ext}, W^{XY}, W^{XX}, W^{YY}, W^{YX}, beginning and ending for parallel events and actions, are defined as before in [41]:

$$\begin{gathered}W^{Ext}=W^{XY}\cup W^{XX}\cup W^{YX}\cup W^{YY},\\W^{XY}\supseteq(B(X)\times Y)^*\cup\{e\},\\W^{XX}\supseteq((B(X)\times Y)^*\cup\{e\})\times B(X),\\W^{YY\prime}\supseteq Y\times((B(X)\times Y)^*\cup\{e\}),\\W^{YX}\supseteq Y\times((B(X)\times Y)^*\cup\{e\})\times B(X).\end{gathered}\tag{5}$$

The presented primary sets from (1) and the set of words (2), (3) allow one to define the complete (2) and external (3) partially observable and controlled behavior. When solving problems of testing, it is through external behavior that it becomes possible to fully or partially establish the correspondence between the checked EPN $S(f)^\wedge$ and the reference EPN $S(f)$.

DEVELOPMENT OF A MODEL OF TESTING WITH VERIFICATION OF ENERGY INPUTS

The features of the developed modified model of behavioral testing include the selection of external, controlled input and observed output characters and words in identifiers of subsets of positions, in check primitives and fragments, inherited from the input model. Such words are defined as vectors of adjacent positions/events and transitions/actions with the registration of energy consumption in positions, transitions and token-monitors.

The class of properties of the EPN, the reference $S(f)$ and the checked $S(f)^\wedge$, assumed for check and taking into account energy consumption, is specified as the relations $F^\wedge$ and F, the correspondence between $S^\wedge$ and S for complete events and actions, as well as the function of marking positions $M^\wedge$ and M for checked $S(f)^\wedge$ and reference $S(f)$, that is, the class is preserved as in the model before modification [41]. Since in $F^\wedge$, F, $S^\wedge$, S, the energy consumption Ep of events Ev for P and the energy consumption Et of actions Ac for T are introduced [41], the class of properties being tested is extended in comparison with a simple Petri net. Consequently, the class of implicitly specified errors for the checked EPN $S(f)^\wedge$ is also preserved [42], firstly, in the structural part – in the differences between the incidence relations $F^\wedge$ and F, as well as the correspondences $S^\wedge$ and S, and secondly, the behavioral part – in the differences between the labeling functions $M_0{}^\wedge$, $M^\wedge$ and M_0, M. Entering energy costs Ep, Et into the properties being checked and, therefore, into the class of errors leads to an increase in the completeness and accuracy of control $S(f)^\wedge$.

The formal basis of recognition in the control model for external (3) – partially observable and controlled behavior is check or recognizing experiments [11], which make it possible to determine the correspondence of the checked EPN $S(f)^\wedge$ and the reference EPN $S(f)$.

Internal – unobservable and uncontrollable behavior $W^{In}=W \backslash W^{Ext}$ can be recognized indirectly - through its manifestation in external behavior, possibly delayed in event time, as the behavior of internal states in automata check or recognition experiments.

In the case of non-redundancy (minimality) of the EPN model $S(f)$, it is assumed that each internal position with an internal event from $Ev \backslash X$ or an internal transition with an internal action from $Ac \backslash Y$ also has its own specific, possibly postponed in event time manifestations in external behavior W^{Ext}.

As a result, it becomes possible to construct control or recognition experiments [11] to testing the complete behavior (2) of non-redundant EPN $S(f)$ provided that internal elements are determined and recognized – subsets of positions with internal events from $Ev \backslash X$ or transitions with internal actions from $Ac \backslash Y$.

First of all, in the internal behavior of W^{In}, one can select internal or partially internal one-step behavior primitive's $inPr$ to be checked with one entry to the transition and one exit for each of them of the form:

$$\begin{gathered}(\{(p_1,ev_1,ep_1),\dots,(p_{ip},ev_{ip},ep_{ip})\},(t_j\ ac,et),\\ \{(p_1{}',ev_1{}',ep_1{}'),\dots,\ (p_{ip}{}',ev_{ip}{}',ep_{ip}{}')\}) \in inPr\end{gathered} \tag{6}$$

such that

$$\begin{gathered}t_j \in F(\{(p_1,ev_1,ep_1),\dots,(p_{ip},ev_{ip},ep_{ip})\})\ \&\\ \&\{(p_1{}',ev_1{}',ep_1{}'),\dots,(p_{ip}{}',ev_{ip}{}',ep_{ip}{}')\} \in F(t_j,ac,et)\\ \&\&\ (ev_1,\dots,ev_{ip} \in Ev \backslash X \text{ or } ev_1{}',\dots,ev_{ip}{}' \in Ev \backslash X \text{ or}\\ ac \in Ac \backslash Y)\end{gathered} \tag{7}$$

On the basis of one-step primitives in internal behavior, it is possible to single out the basic internal structures of behavior – chains, trees, hammocks, cycles – in the general case – the set of basic Petri subnets $subS(f)=\cup_{i \in I} S(f)_i$ from the EPN $S(f)$, in which the events and actions of all their internal (not bordering on the rest of the EPN $S(f)$) positions and transitions are internal, that is, they belong to the sets $Ev \backslash X$ and $Ac \backslash Y$.

Such internal Petri subnets $S(f)_i$ can have sets of external input positions (base) $P_i^{in} \subseteq P_i$ and output positions (antibase) $P_i^{out} \subseteq P_i$. Some subsets of $P_i^{in}{}' \subseteq P_i^{in}$ of positions-inputs together with some, possibly empty, subset of internal positions $P_i{}' \subseteq P_i$ from $S(f)_i$ form the beginning of the path of execution of transitions through $S(f)_i$.

Similarly, some subsets of position-outputs $P_i^{out}{}' \subseteq P_i^{out}$ in combination with some possibly empty subset of internal positions $P_i{}'' \subseteq P_i$ from $S(f)_i$ form the endings of the paths of transitions through $S(f)_i$.

Any path to be verified internal for $S(f)_i$ of performing transitions in the internal alphabets $Ev \backslash X$ and $Ac \backslash Y$ from the external input of the path to $P_i^{in}{}' \cup P_i{}'=\{(p_{11},ev_{11},ep_{11}),\dots,(p_{1ip1},ev_{1ip1},ep_{1ip1})$ to its external output in $P_i^{out}{}' \cup P_i{}''= =\{(p_{k+11},ev_{k+11},ep_{k+11}),\dots,(p_{k+1ipk+1},ev_{k+1ipk+1},ep_{k+1ipk+1})$ $\}$ looks like

$$\begin{gathered}\{(\{(p_{11},ev_{11},ep_{11}),\dots,(p_{1ip1},ev_{1ip1},ep_{1ip1})\},((t_1,ac_1,et_1),\\ \{(\ ev_{21},ep_{21}),\dots,(ev_{2ip},ep_{2ip2})\},(t_2,ac_2,et_2),\dots,\\ \{(ev_{k1},ep_{k1}),\dots,(ev_{kipk1},ep_{kipk1})\},(t_k,ac_k,et_k),\\ \{(p_{k+11},ev_{k+11},ep_{k+11}),\dots,\\ (p_{k+1ipk+1},ev_{k+1ipk+1},ep_{k+1ipk+1})\}))\}.\end{gathered} \tag{8}$$

Then any internal Petri subnet $S(f)_i$, to be checked, which makes it possible to specify the set of such possible through paths for performing transitions (6), is reduced to a form that determines the internal macro-property $inpr_i$ and contains the indicated beginnings and ends of such paths:

$$\begin{gathered} inpr_i=((\{\{(p_{11},ev_{11},ep_{11}),\ldots,(p_{1ip1},ev_{1ip1},ep_{1ip1})\}, \\ \ldots,\ \{(p_{g1},ev_{g1},ep_{g1}),\ldots,(p_{gipg},ev_{gipg},ep_{gipg})\}\},S(f)_i, \\ \{\{(p_{11}',ev_{11}',ep_{11}'),\ldots, \\ (p_{1ip1'}',ev_{1ip1'}',ep_{1ip1'}')\}),\ldots, \\ \{(p_{h11}',ev_{h11}',ep_{h11}'),\ldots,(p_{hiph}',ev_{hiph}',ep_{hiph}')\}\}) \end{gathered} \tag{9}$$

Condensation of the EPN $S(f)$, as a special reduction that subtracts in the macro-position from the set $\mu P=\cup_{i\in I}\mu p_i$ all basic Petri subnets $S(f)_i\in subS(f)$ from $S(f)$, allows one to obtain a macro-EPN $\mu S(f)$, , in which all the usual positions and transitions remaining after inclusion in μP, are marked with external alphabets, respectively, of controlled events X and observable actions Y.

The energy-loaded part of the testing model is based on checking the static part of $F^{\wedge}$. That is, checking the functions $M^{\wedge}$, values of energy consumption $Ep^{\wedge}$ and $Et^{\wedge}$, values of energy consumption in the monitor tokens is performed when checking $F^{\wedge}$ [41].

The check taking into account the unobservable and uncontrollable behavior W^{In}, the energy consumption $Ep^{\wedge}$ and $Et^{\wedge}$ modifies the component model of behavioral control in the input representation [40] for the tested EPN $S(f)^{\wedge}$. The modified control model for $S(f)^{\wedge}$ is:

$$\begin{gathered} CS=(W'^{\wedge}, \{Pr, inPr, mPr\}, \{Ci, inCi\}, \{Cp, \\ inCp\}, Sg_{ca}, Ce_c), \end{gathered} \tag{10}$$

hear:

– $W'^{\wedge}$, Pr, mPr, Ci, Cp, Sg_{ca}, Ce_c – sets of correspondingly registered fragments of behavior, checked properties, three levels of migration of token-monitors, identifiers of positions/transitions, check primitives, signatures of operations of transformations of check analysis, strategy check analyzes are defined in [41], they have not changed or have undergone some extensions and refinements presented here below;

– $Pr=\{pr_{1u}, pr_{2u},\ldots, pr_{ku'}\}=\{Pr_X\cup Pr_Y\}$ is a set of external checked properties of the form:

$$\begin{gathered} Pr\subseteq(F:(B(P\times X\times Ep)\to T)\cup((T\times Y\times Et)\to B(P)))\cup \\ \cup(S: (P\to X\times Ep)\cup(T\to Y\times Et)); \end{gathered} \tag{11}$$

– $inPr=\cup_{i\in I}inpr_i$ – a set of internal verifiable macroproperties - verifiable properties of macro positions μP of condensation $\mu S(f)$ based on internal basic Petri subnets $S(f)_i\in subS(f)$ from $S(f)$;

– $Ci=\{ci_{1ti}, ci_{2ti},\ldots, ci_{kti}\}$ – sets of identifiers of positions/transitions for the macro-EPN $\mu S(f)$.

– $inCi=\{inci_{1ti}, inci_{2ti},\ldots, inci_{kti}\}$ - sets of macroidentifiers of macro positions μP with encapsulated relation F and correspondence S in these macro positions. Macro identifiers depend on the macro EPN $\mu S(f)$ and allow one to identify subsets of the reference macro positions μP in the recorded behavior $W^{\wedge}$ for $\mu S(f)$.. Thus, macro identifiers allow identifying internal behavior encapsulated in macro positions.

The identifiers $inci_{jkpp}{}^{\to}$, $inci_{jkp}{}^{\to}{}_{p}$, $inci_{jktt}{}^{\to}$, $ci_{jkt}{}^{\to}{}_{t}\in Ci$, as previously presented, are defined as twos of the form:

$$\begin{gathered} inci_{jkpp}{}^{\to}=(\mu p_{jtikp}, W_{jtikpp}{}^{\to}{}'), \\ W_{jtikpp}{}^{\to}{}'=\cup_{jtikip=1}^{kp}w_{jtikipp}{}^{\to}{}' \subset W_j', \\ inci_{jkp}{}^{\to}{}_{p}=(W_{jtikp}{}^{\to}{}_{p}', \mu p_{jtikp},), \\ W_{jtikp}{}^{\to}{}_{p}'=\cup_{jtikip=1}^{kp}w_{jtikipp}{}^{\to}{}' \subset W_j', \end{gathered} \tag{12}$$

here $\mu ci_{jkp\mu p}{}^{\to}$, $\mu ci_{jkp}{}^{\to}{}_{\mu p}$ are respectively the initial (for $\mu p^{\to}$) and final (for ${}^{\to}\mu p$) identifiers of the reference condensation positions $\mu S(f)$, uniquely incident to the corresponding macro positions μp_{jtikp}.

On the set $inCi$, the relations $\{\sigma, \eta, \tau, \nu\}$ of compatibility, incompatibility, indeterminacy and precedence are also valid, taking into account the incidence of macro-positions;

– $Cp=\{cp_1, cp_2,\ldots, cp_{k'}\}\subset((Pr^{\circ}Ci)\cup(Ci^{\circ}Pr))$ – a set of external check primitives for checking external checked properties of Pr in the macro EPN $\mu S(f)^{\wedge}$ for compliance with the reference macro-EPN $\mu S(f)$. The set of external check primitives is defined on the basis of external properties Pr of the form pr_{jpp}, pr_{jpt}, pr_{jtp}, $pr_{jtt}\in Pr$, , and identifiers Ci of the form $ci_{jkpp}{}^{\to}$, $ci_{jkp}{}^{\to}{}_{p}$, $ci_{jkpt}{}^{\to}$, $ci_{jkp}{}^{\to}{}_{t}\in Ci$. For example, the check primitives $cp^{\circ}{}_{jkppp}{}^{\to}$, $cp^{\circ}{}_{jkp}{}^{\to}{}_{pp}$, $cp^{\circ}{}_{jktpt}{}^{\to}$, $cp^{\circ}{}_{jkt}{}^{\to}{}_{pt}$, $cp^{\circ}{}_{jkptp}{}^{\to}$, $cp^{\circ}{}_{jkt}{}^{\to}{}_{tp}$, $cp^{\circ}{}_{jkttt}{}^{\to}$, $cp^{\circ}{}_{jkt}{}^{\to}{}_{tt}\in Cp$ look like twos [41]:

$$\begin{gathered} cp^{\circ}{}_{jkppp}{}^{\to}=(pr_{jpp}{}^{\circ}ci_{jkpp}{}^{\to}), \\ cp^{\circ}{}_{jkp}{}^{\to}{}_{pp}=(ci_{jkp}{}^{\to}{}_{p}{}^{\circ}pr_{jpp}),\ cp^{\circ}{}_{jktpt}{}^{\to}=(pr_{jpt}{}^{\circ}ci_{jktt}{}^{\to}), \\ cp^{\circ}{}_{jkp}{}^{\to}{}_{pt}=(ci_{jkp}{}^{\to}{}_{p}{}^{\circ}pr_{jpt}), \\ cp^{\circ}{}_{jkptp}{}^{\to}=(pr_{jtp}{}^{\circ}ci_{jkpp}{}^{\to}),\ cp^{\circ}{}_{jkt}{}^{\to}{}_{tp}=(ci_{jkt}{}^{\to}{}_{t}{}^{\circ}pr_{jtp}), \\ cp^{\circ}{}_{jkttt}{}^{\to}=(pr_{jtt}{}^{\circ}ci_{jtktt}{}^{\to}),\ cp^{\circ}{}_{jkt}{}^{\to}{}_{tt}=(ci_{jkt}{}^{\to}{}_{t}{}^{\circ}pr_{jtt}), \end{gathered} \tag{13}$$

here "°" is the designation of the DeMorgan semiconvolution (concatenation while keeping the common boundary element), taking into account the incidence of adjacent passages identified in the "°" operation, respectively, to transitions or positions.

For newly defined control $cp^{<}{}_{jkppp}{}^{\to}$, $cp^{<}{}_{jkp}{}^{\to}{}_{pp}$, $cp^{<}{}_{jktpt}{}^{\to}$, $cp^{<}{}_{jkt}{}^{\to}{}_{pt}$, $cp^{<}{}_{jkptp}{}^{\to}$, $cp^{<}{}_{jkt}{}^{\to}{}_{tp}$, $cp^{<}{}_{jkttt}{}^{\to}$, $cp^{<}{}_{jkt}{}^{\to}{}_{tt}\in Cp$ this kind of twos is inherent:

$$\begin{gathered}
cp^{<}_{jkppp}{}^{\rightarrow}=(pr_{jpp}\leq ci_{jkpp}{}^{\rightarrow}),\\
cp^{>}_{jkp}{}^{\rightarrow}{}_{pp}=(ci_{jkp}{}^{\rightarrow}{}_{p}\geq pr_{jpp}),\\
cp^{<}_{jktpt}{}^{\rightarrow}=(pr_{jpt}\leq ci_{jktt}{}^{\rightarrow}),\\
cp^{>}_{jkp}{}^{\rightarrow}{}_{pt}=(ci_{jkp}{}^{\rightarrow}{}_{p}\geq pr_{jpt}),\\
cp^{<}_{jkptp}{}^{\rightarrow}=(pr_{jtp}\leq ci_{jkpp}{}^{\rightarrow}),\\
cp^{>}_{jkt}{}^{\rightarrow}{}_{tp}=(ci_{jkt}{}^{\rightarrow}{}_{t}\geq pr_{jtp}),\\
cp^{<}_{jkttt}{}^{\rightarrow}=(pr_{jtt}\leq ci_{jtktt}{}^{\rightarrow}),\\
cp^{>}_{jkt}{}^{\rightarrow}{}_{tt}=(ci_{jkt}{}^{\rightarrow}{}_{t}\geq pr_{jtt}),
\end{gathered}\tag{14}$$

but here "$\leq$" is a designation of a vector-multiple inclusion, taking into account the incidence of adjacent initial or final, identified in the operation "$\leq$", respectively, transitions or positions;

– $inCp=\{incp_{1cp}, incp_{2cp},\ldots, incp_{kcp}\}$ – a set of atomic external control macro-primitives for checking the internal macro-properties of $inPr$ in the macro-EPN $\mu S(f)'$ for compliance with the reference macro-EPN $\mu S(f)$ with encapsulated by the relation F and the correspondence S in the internal macro-positions μP. The set of external check macro-primitives is determined based on the type of internal macro-properties of the form $inpr_{jpp}\in inPr$ and macro-identifiers $inci_{jkpp}{}^{\rightarrow}$, $inci_{jkp}{}^{\rightarrow}{}_{p}\in inCi$ with encapsulated relation F and corresponding S in these macro positions. So, for example, the check macro-primitives $incp^{\circ}_{jkppp}{}^{\rightarrow}$, $incp^{\circ}_{jkp}{}^{\rightarrow}{}_{pp}\in inCp$ and $incp^{<}_{jkppp}{}^{\rightarrow}$, $incp^{<}_{jkp}{}^{\rightarrow}{}_{pp}\in inCp$ have the form of twos:

$$\begin{gathered}
incp^{\circ}_{jkppp}{}^{\rightarrow}=(inpr_{jpp}{}^{\circ}inci_{jkpp}{}^{\rightarrow}),\\
incp^{\circ}_{jkp}{}^{\rightarrow}{}_{pp}=(inci_{jkp}{}^{\rightarrow}{}_{p}{}^{\circ}inpr_{jpp}),\\
incp^{<}_{jkppp}{}^{\rightarrow}=(inpr_{jpp}\leq inci_{jkpp}{}^{\rightarrow}),\\
incp^{>}_{jkp}{}^{\rightarrow}{}_{pp}=(inci_{jkp}{}^{\rightarrow}{}_{p}\geq inpr_{jpp}).
\end{gathered}\tag{15}$$

Earlier in [41, 42], it was noted that the assembly of adjacent primitives into control fragments, performed passively in the online testing or actively in the offline testing, makes it possible to check the behavioral DIS. It was also noted in [41, 42] that in the label hierarchy $mpr_i=\{(root_i, \{(node_{i1}, \{leaf_{i11}, leaf_{i12},\ldots, leaf_{i1L1},\}), (node_{i2}, \{leaf_{i21}, leaf_{i22},\ldots, leaf_{i2L1},\}),\ldots, (node_{iN1}, \{leaf_{iN11}, leaf_{iN12},\ldots, leaf_{i2N1L1},\})\})\}$ for end (leaf) and nodal labels is defined:

$$\begin{gathered}
Leaf_i=\{leaf_{i11},leaf_{i12},\ldots,leaf_{i1L1},\}\cup\\
\cup\{leaf_{i21},leaf_{i22},\ldots,leaf_{i2L1},\}\cup\ldots\cup\\
\{leaf_{iN11},leaf_{iN12},\ldots, leaf_{i2N1L1},\}\\
Node_i=\{node_{i1}\}\cup\{node_{i2}\}\cup\ldots\cup\{node_{iN1}\}.
\end{gathered}\tag{16}$$

The current leaf label $leaf_{ij}(p)\in Leaf_i$ of an arbitrary position $p\in P$, as well as the current leaf label $leaf_{ij}(t)\in Leaf_i$ of an arbitrary transition $t\in T$ for the corresponding state $S(f)$ is defined as:

$$\begin{gathered}
leaf_{ij}(p)=M(p)=pr_2(K(p,ev',ep'))=ep,\\
leaf_{ij}(t)=pr_2(K(t,ac',et'))=et.
\end{gathered}\tag{17}$$

The initial label $leaf_{ij}(p)_0\in Leaf_{i0}$ of an arbitrary position $p\in P$ in the initial state of $S(f)^i$ is defined as $leaf_{ij}(p)_0=M_0(p)$.

The node label $node_{ij}\in Node$ of an arbitrary topological element, the root label $root_i$ of an arbitrary hierarchy, the label, of the entire $S(f)$ in their initial or current state determines the accumulation of energy consumption based on the energy consumption of the lowest labels in the hierarchy [41]:

$$\begin{gathered}
node_{ij}=leaf_{ij1}+leaf_{ij2}+\ldots+leaf_{ijLj},\\
root_i=node_{i1}+node_{i2}+\ldots+node_{iNi}\\
PNEnergy=root_1+root_2+\ldots+root_R\,.
\end{gathered}\tag{18}$$

The development of an energy-loaded model makes it possible to determine the conditions of behavioral working and test control taking into account energy consumption with partial controllability and observability of events and actions of the $S(f)$ model.

Together with energy-loaded identifiers, primitives, fragments of the modified testing model, as in [41], in behavioral check procedures [42] based on Petri nets $S(f)$, binding, previously checked fragments are used, if necessary, to form connectivity when assembling non-adjacent primitives and fragments.

ESTIMATES OF THE DIMENSION OF THE CONTROL MODEL

The dimension of the model is estimated using the representation of the Petri net digraph $S(f)$ by list structures. Let $|P|=n_p$, $|T|=n_t$, $|M|=n_m$, $n=n_p+n_t+2n_m$ (here $2n_m$ are two fields with the index of the energy-loaded type label and their number$|Ev|=n_e$, $|X|=n_x$, $|Ac|=n_a$, $|Y|=n_y$, where $X\subseteq Ev$ and $Y\subseteq Ac$. A representation for transmission requires no more than n_t conditional memory cells, each of which contains no more than $2n_p+1_t+1_a+2_{Addr}$ or $2n_p+4$ conditional fields, requires no more than np conditional memory cells for a position, each of which contains no more than $2n_t+1_p+1_e+2_{Addr}$ or $2n_t+4$ conditional fields for position. Here i_p is a field with a position index, it is a field with a transition index, $2i_m$ are two fields with an index (energy load) and the number of label instances, i_e is a field with an event index, i_a is a field with an action index, $2i_{Addr}$ are two fields with the address of the next and previous cells in the list for a position or jump.

The upper bound for the number of conditional fields is:

$$\begin{gathered}
c_{S(f)}=n_t(2n_p+1_t+1_a+2_m+2_{Addr})+n_p(2n_t+1_p+1_e+\\
+2_m+2_{Addr})=\\
=4n_pn_t+(2_m+2_{Addr})(n_t+n_p)+1_tn_t+1_an_t+1_pn_p+1_en_p\cong\\
\cong 4n_pn_t+6(n_t+n_p).
\end{gathered}\tag{19}$$

The modified graph of reachable states (automata $A_{S(f)}$, representing the operation of the Petri net $S(f)$, multiple for positions and parallel for transitions, in the limiting case for each input or output set of positions of some transition contains n_t adjacent transitions, including itself, with their input-output sets with at most n_p positions, that is, together - $2n_tn_p$, similarly for each input or output set of transitions of a certain position contains np adjacent positions, including itself, with their input-output sets with at most nt transitions, that is together - 2ntnp. In the limiting case, for each set of parallel transitions (multi-transitions) and multiple positions (multi-positions), the automata $A_{S(f)}$ contains, respectively, 2^{nt} adjacent multi-transitions and 2^{np} adjacent multi-positions with their sets (with at most n_p positions and n_t transitions), that is:

$$\begin{gathered} 2n_p(2^{nt})\text{-}1=n_p2^{nt+1}\text{-}1, \\ 2n_t(2^{np})\text{-}1=n_t2^{np+1}\text{-}1. \end{gathered} \tag{20}$$

In the limiting case, for all transitions and positions, multi-transitions and multi-positions, the number of conditional cells and the maximum length when searching in them is determined as the sum of two corresponding terms:

$$\begin{gathered} cell_{AS(f)}=2n_tn_p(n_t)+2n_pn_t(n_p)=2n_pn_t(n_p+n_t), \\ cell_{AS(f)multi}= n_p2^{nt+1}\text{-}1+ n_t2^{np+1}\text{-}1= \\ =n_p2^{nt+1}+ n_t2^{np+1}\text{-}2. \end{gathered} \tag{21}$$

Accordingly, the longest search for all fields for one transition follows from $c_{AS(f)}$, taking into account the verification of all fields, including actions, is equal to $12n_p(n_t)$. The longest search for all fields for one position follows from $c_{AS(f)}$, taking into account all fields, including conditions, is $12n_t(n_p)$. Taking into account the presented index and address fields of positions/transitions, index fields of labels, events and actions, the last formulas of the limiting case for the number of conditional cells and the maximum length when searching in them take the form:

$$\begin{gathered} c_{AS(f)}= d_{AS(f)}= \\ 2*6n_tn_p(n_t)+2*6n_pn_t(n_p)=12n_pn_t(n_p+n_t), \\ c_{AS(f)multi}= d_{AS(f)multi}= 6n_t(n_p2^{nt+1}\text{-}1)+ \\ +6n_p(n_t2^{np+1}\text{-}1)= \\ =3n_t(n_p2^{nt+2}\text{-}2)+3n_p(n_t2^{np+2}\text{-}2). \end{gathered} \tag{22}$$

The presentation of the Rabin-Scott multilevel automaton $RS_{S(f)}$ – a special graph of reachable states for determining the set of identifiers – is limited by the number of identifiable positions / transitions and the presence of final categorical vertices that fix the formation of identifiers of positions / transitions. Without taking into account the limited number of subsets of positions, the upper bound on the number of fields in conditional cells of all levels for branching can be determined not more than $(2n_p+1)*(2^{nt}\text{-}1)$. Then, in the general case, the upper bound on the number of fields in conventional cells of all levels is determined by no more than:

$$\begin{gathered} c_{RSS(f)p\text{-}n}=min((2n_p+1)*min((2^{np}\text{-}1), n_t), \\ (2n_p+1)**(2^{nt}\text{-}1))= \\ =(2n_p+1)*min(min((2^{np}\text{-}1), n_t), \\ (2^{nt}\text{-}1)). \end{gathered} \tag{23}$$

The number of fields in conditional cells of branch-arcs is not more than:

$$c_{RSS(f)t\text{-}n}= \Sigma_{i=0}{}^{np}(3n_p*n_t{}^i). \tag{24}$$

The total assessment of the fields does not exceed

$$\begin{gathered} c_{RSS(f)pet\text{-}n}= (2n_p+1)*min(min((2^{np}\text{-}1), n_t), \\ (2^{nt}\text{-}1))+\Sigma_{i=0}{}^{np}(3n_p*n_t{}^i). \end{gathered} \tag{25}$$

In the Rabin-Scott automaton $RS_{S(f)}$, as a result of the reappearance of subsets/vectors of positions, converging branches (hammocks) and feedback loops are possible.

The estimate of the maximum number of conditional cells for the checked properties and identifiers is determined by the length of the linear indivisible sections of the reference behavior has the form no more than:

$$\begin{gathered} c_{PrS(f)}{}^{max}=2(2n_p+1+n_t)*n_t, \\ c_{CiS(f)}{}^{max}=2(2n_p+1+n_t)*n_t. \end{gathered} \tag{26}$$

The estimate of the number of conditional cells for any of the control primitives is determined by the sums of the estimates of the number of conditional cells for the checked property and identifier that make it up.

For simple and multiple checked properties and identifiers, the estimate of the maximum number of conditional cells of the check primitive is not more than:

$$\begin{gathered} c_{cpS(f)}{}^{max}=2((2n_p+1)*(n_t+1)+2n_t), \\ c_{CpS(f)}{}^{max}=4(2n_p+1+n_t)*n_t. \end{gathered} \tag{27}$$

These estimates represent the limits of applicability of the abstract energy-loaded model of behavioral control. The polynomial lowering of the estimates can be performed by applying the network and hierarchical decomposition of the input Petri net model $S(f)$.

CONCLUSION

The paper presents the development of a DIS behavioral model of testing based on extended Petri nets, which has features of properties of analyzing energy consumption, partial control over events and observability of actions.

By defining the conditions for behavioral control based on Petri nets, taking into account the energy consumption and the recognition of hidden events and actions, the modified model of testing allows you in addition to normal behavioral check to find distribute-combine and save energy consumption indicators for vertices, topological elements and subnets, for the entire Petri net.

This circumstance provides a basis for constructing behavioral check procedures for components of distributed information systems with incomplete controllability and observability, extended by both detailed and total verification of their energy consumption, which allows increasing the completeness and accuracy of testing in general.

REFERENCES

1. Van Steen Maarten & Tanenbaum Andrew S. “A Brief Introduction to Distributed Systems”. *Computing*. 2016; 98: 967–1009. DOI: https://doi.org/10.1007/s00607-016-0508-7.

2. Jerabek, M. “Hidden Object's Identity Algorithm for Distributed Information Systems”. *8th WSEAS International Conference on Data Networks, Communications, Computers*. November, 2009. 190–194.

3. Htoon, H. & Thet Thwin, M. M. “Mobile Agent for Distributed Information Retrieval System”. *5th International Conference on Electrical Engineering/Electronics, Computer. Telecommunications and Information Technology*. 2008. p. 69–172. DOI: https://doi.org/10.1109/ECTICON.2008.4600399.

4. Kelion, L. “Intel's Ivy Bridge Chips Launch Using '3D Tansistors' ”. *BBC*. 19 April 2013.

5. Patil, A. & Goti, A. ‘Green Computing in Communication Networks’. *International Journal of Computer Applications*. 2013; Vol. 68 No.23: 1–5. DOI: https://doi.org/ 10.5120/11716-7235.

6. Kazandjieva, M., Shah, C., Cheslack-Postava, E., Mistree, B. & Levis P. “System Architecture Support for Green Enterprise Computing”. *5th International Green Computing Conference (IGCC)*. 2014: 10. – Available from: https://sing.stanford.edu/site/publications/igcc14kazandjieva.pdf. – [Accessed: Feb, 2021].

7. Rierson, L. “Developing Safety-Critical Software: A Practical Guide for Aviation Software and DO-178C Compliance”. *Publ. CRC Press*. 2013. 610 p.

8. Sommerville, I. “Critical systems”. *Somerville’s Book Website*. 2014. Retrieved 18 April 2018. – Available from: https://sites.google.com/site/iansommerville/teaching-videos/critical-systems-videos. – [Accessed: Feb, 2021].

9. Koopman, P. “Critical Systems and Software Safety. Distributed Embedded Systems”. Carnegie Mellonm. November 16; 2015: 58 p. – Available from: https://users.ece.cmu.edu/~koopman/ece649/lectures/20_critical_systems.pdf. – [Accessed: Feb, 2021].

10. Sahlin, K. & Medvedev, P. “Error Correction Enables use of Oxford Nanopore Technology for Reference-Free Transcriptome Analysis”. *Nat Commun*, 2021. 12 p. DOI: https://doi.org/10.1038/s41467-020-20340-8.

11.Kudryavtsev, V. B., Grunskii, I. S. & Kozlovskii, V. A. “Analysis and Synthesis of Abstract Automata”. *Journal of Mathematical Sciences*. September 2010; Vol. 169 Issue 4: 481–532. DOI: https://doi.org/10.1007/s10958-010-0058-z.

12. Laycock, G. T. “The Theory and Practice of Specification Based Software Testing (dissertation)”. *Department of Computer Science* University of Sheffield. Retrieved January 2, 2018. – Available from: https://www.cs.le.ac.uk/people/glaycock/thesis.pdf. – [Accessed: Feb, 2021].

13.“Computer efficiency”. *McGraw-Hill Dictionary of Scientific & Technical Terms, 6E*. S.v. Retrieved June 14, 2021. – Available from: https://encyclopedia2.thefreedictionary.com/ computer+efficiency. – [Accessed: Feb, 2021].

14. Ryabko, B. “On the efficiency and capacity of computers”. *Applied Mathematics Letters*. 2012; Vol. 25 Issue 3: 398–400. DOI: https://doi.org/10.1016/j.aml.2011.09.021.

15. Shooman, M. L. “Reliability of Computer Systems and Networks: Fault Tolerance, Analysis, and Design”. Copyright 2002 John Wiley&Sons Inc.: 546. – Available from: http://inis.jinr.ru/sl/tot_ra/0/0/3/Shooman-Reliability.pdf. – [Accessed: Feb, 2021].

16. Rausand, M. & Hoyland, A. “System Reliability Theory: Models, Statistical Methods, and Applications”. Journal Wiley & Sons. 2004: 629. – Available from: https://notendur.hi.is/hthb1/%C3%81rei%C3%B0anleiki%20kerfa/System%20Reliability%20Theory%20Models%20and%20Statistical%20Methods.pdf . – [Accessed: Feb, 2021].

17. Laros, J. H., Pedretti, K., Kelly, S. M., Wei Shu, Ferreira, K., Van Dyke, J. & Vaughan, C. “Energy-Efficient High Performance Computing: Measurement and Tuning”. *Springer Science & Business Media.*

2012. – Available from: https://books.google.com.ua/books?id=Wq_XngsfAPQC&printsec=frontcover&hl=ru#v=onepage&q&f=false. – [Accessed: Feb 2021].

18. Akram, A. & Sawalha, L. “A Study of Performance and Power Consumption Differences Among Different ISAs”. *22nd Euromicro Conference on Digital System Design (DSD)*. 2019. p. 628–632. DOI: https://doi.org/10.1109/DSD.2019.00098.

19. Lee, W. “Tutorial: Design and Optimization of Power Delivery Networks”. *IEIE Transactions on Smart Processing and Computing*. 2016; Vol. 5 No. 10: 349–357. DOI: https://doi.org/10.5573/IEIESPC.2016.5.5.349.

20. Hameedah Sultan, Anjali Chauhan & Smruti R. Sarangi. “A Survey of Chip-Level Thermal Simulators”. *ACM Comput. Surv*. April 2019. p. 1–35. DOI: https://doi.org/10.1145/3309544.

21. Toft, J. K. & Nannarelli, A. “Energy Efficient FPGA Based Hardware Accelerators for Financial Applications”. *2014 NORCHIP*. 2014. p. 1–6. DOI: https://doi.org/10.1109/NORCHIP.2014.7004741.

22. Chen, X., Xu, J. & Yu, Z. “A Fast and Energy Efficient FPGA-Based System for Real-Time Object Tracking”. *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. 2017. p. 965–968. DOI: https://doi.org/10.1109/APSIPA.2017.8282162.

23. Prasanna, V. K. “Energy-Efficient Computations on FPGAs”. *Journal Supercomputers*. 2005. p.139–162. DOI: https://doi.org/10.1007/s11227-005-0289-9.

24. Drozd, O., Kuznietsov, M., Martynyuk, O. & Drozd, M. “A Method of the Hidden Faults Elimination in FPGA Projects for the Critical Applications”. *DESSERT 2018 - 9th IEEE International Conference on Dependable Systems, Services and Technology*. Kyiv: Ukraine. 2018. p. 231–234.

25. Drozd, A., Drozd, M., Martynyuk, O. & Kuznietsov, M. “Improving of a Circuit Checkability and Trustworthiness of Data Processing Results in LUT-based FPGA Components of Safety-Related Systems”. *CEUR Workshop Proceedings*, 2017; Vol. 1844: 654–661. – Available from: http://ceur-ws.org/Vol-1844/10000654.pdf. – [Accessed: Feb, 2021].

26. Drozd, O. Romankevich, V., Kuznietsov, M., Drozd, M. & Martynyuk, O. “Using Natural Version Redundancy of FPGA Projects in Area of Critical Applications”. *11th IEEE International Conference on Dependable Systems, Services and Technology (DESSERT’2020)*. Kyiv: Ukraine. May 14-16, 2020. p. 58–64. DOI: https://doi.org/10.1109/DESSERT50317.2020.91250506.

27. Ivanchenko, O., Kharchenko, V., Moroz, B., Kabak, L. & Konovalenko, S. “Risk Assessment of Critical Energy Infrastructure Considering Physical and Cyber Assets: Methodology and Models”. *2018 IEEE 4th International Symposium on Wireless Systems within the International Conferences on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS-SWS)*. 2018. p. 225-228. DOI: https://doi.org/10.1109/IDAACS-SWS.2018.8525594.

28. Tyurin, S. & Kamenskih, A. “Green Logic: Models, Methods and Algorithms”. *Green IT Engineering: Concepts, Models, Complex Systems Architectures. Studies in Systems, Decision and Control*. V. Kharchenko, Y. Kondratenko, J. Kacprzyk (eds). *Publ. Springer*. 2017; Vol.74: 69–86.

29.Cuadrado-Cordero, I., Orgerie, A.-C. & Menaud, J.-M. “Comparative Experimental Analysis of the Quality-of-Service and Energy-Efficiency of VMs and Containers' Consolidation for Cloud Applications”. Archived June 28, 2018, at *The Wayback Machine International Conference on Software, Telecommunications and Computer Networks* (SoftCOM 2017). Split: Croatia. Sep, 2017. p. 1–6.

30. Kazandjieva, M., Heller, B., Gnawali, O., Hofer, W, Levis, P. & Kozyrakis, C. “Software or Hardware: The Future of Green Enterprise Computing”. *Computer Science Technical Report CSTR*. 2011; Vol. 2: 13 p. – Available from: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.366.5840&rep=rep1&type=pdf. – [Accessed: Feb, 2021].

31.“Greenpeace “Clicking clean: who is winning the Race to Build a Green Internet””. (PDF) 2017. Archived from the original on July 6, 2018. – Available from: https://www.actu-environnement.com/media/pdf/news-28245-clicking-clean-2017.pdf. – [Accessed: Feb, 2021].

32.Kristensen, L. M. & Westergaard, M. “Automatic Structure-Based Code Generation from Coloured Petri Nets: A Proof of Concept. Formal Methods for Industrial Critical Systems”. *15th International Workshop, FMICS*. Lecture Notes in Computer Science. 2010. p. 215–230. DOI: https://doi.org/10.1007/978-3-642-15898-8_14.

33.Grobelna, I. “Formal Verification of Embedded Logic Controller Specification with Computer Deduction In Temporal Logic”. *Przeglad Elektrotechniczny*, 2011: Vol.87 Issue 12a: 47–50. – Available from: http://pe.org.pl/articles/2011/12a/11.pdf. – [Accessed: Feb, 2021].

34.Grobelna, I., Grobelny, M. & Adamski, M. “Model Checking of UML Activity Diagrams in Logic Controllers Design”. *Proceedings of the Ninth International Conference on Dependability and Complex*

Systems DepCoS-RELCOMEX. Advances in Intelligent Systems and Computing. Brunów: Poland. June 30–July 4, 2014. p. 233–242. DOI: https://doi.org/10.1007/978-3-319-07013-1_22.

35. Shuja, J., Ahmad, R.W., Gani, A. *et al.* "Greening Emerging IT Technologies: Techniques and Practices". *Journal Internet Serv Appl.* 2017. Vol. 8. DOI: https://doi.org/10.1186/s13174-017-0060-5

36. "The Utimate Green Technology: Creating Computers that use 10,000 Times less Energy". University of Alberta. *Science Daily*. – Available from: www.sciencedaily.com/releases/2017/02/170213083752.htm. – [Accessed: Feb, 2021].

37. Menezes, A. C., Cripps, Andrew & Buswell, R. A. & Wright, Jonathan & Bouchlaghem, D. "Estimating the Energy Consumption and Power Demand of Small Power equipment in Office Buildings". *Energy and Buildings*. 2014. p. 199–209. DOI: https://doi.org/10.1016/j.enbuild.2014.02.011.

38. Kovalev, I. S., Drozd, O. V., Rucinski, Andrzej, Drozd, M. O., Antoniuk, V. V., Sulima, Yu.Yu. "Development of Computer System Components in Critical Applications: Problems, their Origins and Solutions". *Herald of Advanced Information Technology*. *Publ. Nauka i Technica*. 2020; Vol.3 No.4: 252–262. Odesa: Ukraine. DOI: https://doi.org/10.15276/hait.04.2020.4.

39. Quariguasi-Frota-Neto, et al. "An Analysis of the Eco-Efficiency of Remanufactured Personal Computers and Mobile Phones". *Production and Operations Management Society*. 2012; Vol. 21 No 1: 101–114. DOI: https://doi.org/10.1111/j.1937-5956.2011.01234.x.

40. Kahhat, R. & Williams, E. "Materials Flow Analysis of E-Waste: Domestic Flows And Exports of Used Computers from the United States". *Resources, Conservation and Recycling*. 2012. p. 67–74. DOI: https://doi.org/10.1016/j.resconrec.2012.07.008.

41. Martynyuk, A. N., Bui, Van Thyong, Martynyuk, D. A. & Stepova, H. S. "Behavioral Model of Testing for Network Systems Energy" (in Russian). *Electrotechnic and Computer Systems Information Systems and Technology*. Odessa: Ukraine. 2018; No. 28(104): 208–214. – Available from: https://eltecs.op.edu.ua/index.php/journal/article/view/159/46. – [Accessed: Feb, 2021].

42. Martynyuk, O., Sugak, L., Stepova, H., Antoniuk, V. & Mileiko, I. "Models of Distributed Systems Testing Based on Energy Consumption in Behavior". *IntelITSIS'2021: 2nd International Workshop on Intelligent Information Technology and Systems of Information Security*. Khmelnytskyi: Ukraine. March 24-26, 2021. p. 481–493. – Available from: http://ceur-ws.org/Vol-2853/paper52.pdf. – [Accessed: Feb, 2021].

Conflicts of Interest: The authors declare no conflict of interest

Received 06.10.2020
Received after revision 03.03.2021
Accepted 15.03.2021

DOI: https://doi.org/10.15276/hait.02.2021.3
УДК 004.738 + 004.9

Поведінковий прихований контроль розподілених інформаційних систем з урахуванням енерговитрат

Олександр Миколайович Мартинюк[1)]
ORCID: https://orcid.org/0000-0003-2366-1920; anmartynyuk@ukr.net
Олександр Валентинович Дрозд[1)]
ORCID: https://orcid.org/0000-0001-8305-2217; drozd@ukr.net
Сергій Анатолієвич Нестеренко[1)]
ORCID: https://orcid.org/0000-0002-3757-6594; sa_nesterenko@ukr.net
Вадим Юрієвич Скобцов [2)]
ORCID: https://orcid.org/0000-0002-8546-0430; vasko_vasko@mail.ru
Тхіонг Ван Буі [1)]
ORCID: https://orcid.org/0000-0002-9160-5982; govarava@gmail.com
[1)] Одеський національний політехнічний університет, пр. Шевченка, 1. Одеса, 65044,Україна
[2)] Об'єднаний Інститут проблем інформатики Національної академії наук Білорусії, вул. Сурганова, 6. Мінськ, 220012, Білорусь

АНОТАЦІЯ

Введення додаткових нових енерговитратних властивостей для позицій і переходів у властивості, які перевіряються в розширеної еталонної мережі Петрі, для яких визначаються відхилення мережі Петрі і розробляється модель контролю, дає нові можливості діагностування. Зберігаючи клас перевіряються властивостей в складі відхилень відносин інцидентності, відповідностей і функцій розмітки позицій і переходів для перевіряємої і еталонної мереж Петрі, нові властивості дозволяють фіксувати появу критичних температурних режимів, які є наслідком помилок або безпосередньо ведуть до їх появи. Така різнобічність контролю сприяє підвищенню його повноти, точності і оперативності. Енерго-навантажена модель контролю заснована на базовій перевірці відношень інцидентності, відповідностей і функцій розмітки. Перевірка функцій розмітки при формуванні подій в позиціях, виконанні дій в переходах, а також запропонована перевірка показників енерговитрат, що накопичуються в фішках-моніторах, виконується при зазначеної перевірці відношень інцидентності, відповідностей і функцій розмітки. До особливостей моделі контролю відноситься введення узагальнених енерго-навантажених фішок-реєстраторів мереж Петрі, які накопичують інформацію про енерговитрати в поведінці елементів трьох рівнів - позицій/переходів, топологічних компонентів і підмереж, всієї мережі Петрі в процесі її функціонування. Модель контролю також відрізняється розпізнаванням еталонної енерго-навантаженої поведінки при перевірці мережі Петрі на основі поведінкової ідентифікації і ототожнення підмножин позицій і переходів, детермінізації поведінки, застосуванням контрольних примітивів і транзакцій. Поведінкова модель контролю визначає формальні умови для процедур поведінкового контролю, що включає аналіз коректності енергоспоживання. Розмірність моделі контролю оцінена за допомогою представлення графів мережі Петрі, спеціальних графів досяжних станів, в тому числі автоматів Рабина-Скотт, за допомогою спискових структур. Наведені оцінки визначають межі застосування формальної моделі контролю.

Ключові слова: Інформаційна система; енергетична поведінка; тестування поведінки; мережа Петрі; ідентифікатор; примітив перевірки

ABOUT THE AUTHORS

Oleksandr M. Martynyuk – PhD (Eng), Associate Professor of the Department of Computer Intellectual Systems and Networks. Odessa National Polytechnic University. 1, Shevchenko Ave. Odesa, 65044, Ukraine
ORCID: https://orcid.org/0000-0003-2366-1920; anmartynyuk@ukr.net
Research field: Behavioral Testing of Computer Systems; Formal Verification and Recognizing of Digital Systems; Artificial Intelligence

Олександр Миколайович Мартинюк – кандидат технічних наук, доцент кафедри Комп'ютерних інтелектуальних систем та мереж. Одеський національний політехнічний ун-т. пр. Шевченка, 1. Одеса, 65044, Україна

Olesandr Va. Drozd– Dr. Sci. (Eng), Professor, Professor of the Department of Computer Intellectual Systems and Networks Odessa National Polytechnic University. 1, Shevchenko Ave. Odesa, 65044, Ukraine
ORCID: https://orcid.org/0000-0001-8305-2217; drozd@ukr.net
Research field: Testing and Diagnosis of Computer Systems; Arithmetical Foundations of Computer Systems; Computer Systems and Components

Олександр Валентинович Дрозд – доктор технічних наук, професор кафедри Комп'ютерних інтелектуальних систем та мереж. Одеський національний політехнічний ун-т, пр. Шевченка, 1. Одеса, 65044, Україна

Sergiy A. Nesterenko – Dr. Sci. (Eng), Professor, Professor of the Department of Computer Intellectual Systems and Networks, Prorector. Odessa National Polytechnic University. 1, Shevchenko Ave. Odesa, 65044, Ukraine
ORCID: https://orcid.org/0000-0002-3757-6594; sa_nesterenko@ukr.net
Research field: Computer Networks; Artificial Intelligence; Microprocessor Systems; Neural Networks

Сергій Анатолієвич Нестеренко – доктор технічних наук, професор, професор кафедри Комп'ютерних інтелектуальних систем та мереж, проректор. Одеський національний політехнічний ун-т, пр. Шевченка, 1. Одеса, 65044, Україна

Vadym Yu. Skobtsov – PhD (Eng), Leading researcher, Associate Professor of United Institute of Informatics Problems. National Academy of Sciences. 7, Surganova Str. Minsk, 220012, Belarus
ORCID: https://orcid.org/0000-0002-8546-0430; smyk@gmail.com
Research field: Analyze, Testing and Diagnosis of Computer Systems; Artificial Intelligence

Вадім Юрійович Скобцов – кандидат технічних наук, провідний науковий співробітник, доцент Об'єднаного інституту проблем інформатики. Національна академія наук Білорусії, вул. Сурганова, 6. Мінськ, 220012, Білорусь

Thuong Van Bui – PhD Student of the Department of Computer Intellectual Systems and Networks. Odessa National Polytechnic University. 1, Shevchenko Ave. Odesa, 65044, Ukraine
ORCID: https://orcid.org/0000-0002-9160-5982; govarava@gmail.com.
Research field: Testing and Diagnosis of Computer Systems; "Green" Computer Systems and Components; Internet of Things

Тхіонг Ван Буі – аспірант кафедри Комп'ютерних інтелектуальних систем та мереж. Одеський національний політехнічний ун-т, пр. Шевченка, 1. Одеса, 65044, Україна