

## Экспериментально-статические модели распределённых технологических объектов

*Цель статьи – совершенствование подхода к конфигурации программного обеспечения автоматизированных систем управления. Предмет исследования – экспериментально-статические модели, используемые в этом процессе. Работа выполнена с использованием теории отношений, графов и матриц. В результате получен универсальный метод машинного синтеза соответствующих моделей. Реализованный на его основе алгоритм позволяет автоматизировать процесс конфигурации систем управления распределёнными технологическими объектами. Ил. 2. Табл. 1. Библиогр.: 7 назв.*

**Ключевые слова:** конфигурация программного обеспечения, автоматизированная система управления, экспериментально-статическая модель, машинный синтез, алгоритм, распределённый технологический объект

*A purpose of the article is perfection of going near control CASS software configuration. The object of research is experimental-static models, in-use this process. Work is executed with the use of relations, counts and matrices theories. The universal method of the proper models machine synthesis is as a result got. The algorithm realized on his basis allows to automatize the configuration process of control systems by the up-diffused technological objects.*

**Keywords:** software configuration, control CAS, experimental-static model, machine synthesis, algorithm, up-diffused technological object

### Введение

Развитие современного металлургического производства сопровождается интенсивным внедрением автоматизированных систем управления технологическими процессами (АСУТП) во все его сферы. Особый класс задач представляет автоматизация централизованного управления распределёнными технологическими объектами (главным образом – по территориальному принципу). К таким объектам относятся как узлы основного производства (аглофабрики, конвертерные цеха, шахты и т.д.), так и промышленные энергетические, телекоммуникационные и транспортные системы и сети. Последние играют немаловажную роль в технологическом цикле металлургических, горнорудных и коксохимических предприятий.

Основой функционирования АСУТП являются математические модели производственных объектов и процессов. Большинство современных АСУТП базируется на программной логике, что обуславливает единство базового ядра программного обеспечения (ПО) самих систем и экспериментальных моделей для их исследования. Однако сложные распределённые объекты автоматизации единого ряда характеризуются различной конфигурацией конкретных технических решений. В связи с этим необходима адаптация ПО под определённый технический проект, что является отдельным этапом проектирования самой системы.

Традиционно эта задача решается с помощью файлов конфигурации, создаваемых в формате *xml*, *txt*, *cfg*, *dll*, *ini* и т.д. Непосредственное их составление требует высокой квалификации проектировщика, в т.ч. определённых знаний и навыков в области про-

граммирования. Для упрощения и ускорения данной процедуры разработаны множественные средства и системы автоматизированного проектирования (САПР) таких файлов, в основу которых закладываются экспериментально-статические модели (ЭСМ) соответствующих объектов.

Однако существенным недостатком большинства подобных САПР является отсутствие их унификации в отношении разнородных АСУТП. Другим недостатком является лимитирование размерностей проектируемых объектов в связи с ограниченными возможностями средств визуализации интерфейса с пользователем, нагромождением анализируемой информации и т.п.

Целью данной работы является разработка универсальных методов синтеза входных ЭСМ для АСУТП объектов больших размерностей на ЭВМ.

### Исходное представление объекта автоматизации

Распределённый технологический объект можно охарактеризовать множеством  $Y = A \cup U \cup Z$ , где  $A$ ,  $U$  и  $Z$  – соответственно, подмножества локальных объектов управления и контроля, их функциональных свойств и логических связей между ними. Данное множество можно построить таким образом, чтобы каждому элементу  $a_i \in A$  соответствовали единственный элемент  $z_{i,i+1} \in Z$ , связывающий его с элементом  $a_{i+1} \in A$ , а также подмножество, которое задаёт его свойства, воспроизводящие его функции. Исходя из этого, каждое из подмножеств  $Y$  задаёт отношения на других его подмножествах или внешних множествах [3]

$$\begin{cases} (\forall a_i \in A) \rightarrow (\exists! z_{i,i+1} \in Z) : a_i(z_{i,i+1})a_{i+1} \\ (\forall a_i \in A) \rightarrow (\exists \{u_j\} \in U) : a_i(u_j)\{f_{jij}\} \end{cases} \rightarrow \begin{cases} A \subset Z \times Z \wedge U \times U \\ U \subset A \times F \\ Z \subset A \times A \end{cases}, (1)$$

где  $F = \{f_{ij}^a\}$  – множество функций элемента  $a_j$  задающихся его свойствами  $\{u_{ij}^a\} \in U$ .

Множество  $F$  является характеристикой динамической составляющей АСУТП или её испытательной модели и однозначно определяется множеством  $U$ , на которое оно биективно отображается ( $F \leftrightarrow U$ ). Отношение  $U \subset A \times F$  в формуле (1), которое должно быть заложено в исходном коде прикладного ПО, связывает входную ЭСМ с самой системой. Поэтому входная модель может не включать множество  $F$ .

Исходя из выражения (1) множество  $Y$  может быть однозначно отображено связным, смешанным и взвешенным  $p \times q$  графом  $G_Y = (V, E)$ , для которого

$$(\forall v_i \in V, \forall e_j \in E) \leftrightarrow (\overline{u_{vi}^g}, \overline{u_{ej}^g}) \in U^G = U \cup \Delta U$$

, где  $\Delta U$  – дополнительное подмножество собственных свойств вершин  $v_i$  и рёбер  $e_j$  в составе графа  $G_Y$  для идентификации его элементов. Соответствие  $G_Y \leftrightarrow Y$  воспроизводится одним из следующих способов:

$$\left\{ \begin{array}{l} A \leftrightarrow V \\ Z \leftrightarrow E \\ U = U^G / \uparrow U \end{array} \right\} \vee \left\{ \begin{array}{l} A \leftrightarrow E \\ Z \leftrightarrow V \\ U = U^G / \Delta U \end{array} \right\} \vee \left\{ \begin{array}{l} A \leftrightarrow G_Y^h \\ Z \subset V \times E \\ U = U^G / \Delta U \end{array} \right\} \vee \left\{ \begin{array}{l} A_1 \leftrightarrow V_1 \wedge A_2 \leftrightarrow G_Y^p \\ Z \subset V \times E \\ U = U^G / \Delta U \end{array} \right\}, \quad (2)$$

где  $G_Y^h, G_Y^p \subset G_Y$  – деревья графа  $G_Y$ ;  $A_1, A_2 \subset A$  – подмножества, воспроизводимые соответственно вершинами  $V_1$  и деревьями  $G_Y^p$  графа  $G_Y$

$$(A_1 \cup A_2 = A, G_Y^p \supset V / V_1 = V_2).$$

Отношение  $Z \subset V \times E$  в формуле (2) определяет инцидентность между соответствующими вершинами и рёбрами графа  $G_Y$ . Учитывая, что деревья графов всегда ограничены вершинами, соответствие  $A_1 \leftrightarrow E_1 \wedge A_2 \leftrightarrow G_Y^p$  невозможно.

### Аналитическое представление ЭСМ

Увязка ЭСМ с прикладным ПО требует аналитической интерпретации графа  $G_Y$ . Подход, основанный на непосредственном применении топологических матриц, в данном случае не является приемлемым для объектов больших размерностей. Решение этой проблемы найдено в исходном формировании обычных блоков матрицы векторных элементов, которая воспроизводит как топологические, так и параметрические (весовые) свойства графической модели. Такой матрице присвоено название комплексной топологической матрицы (КТМ). Блоки создаются на основе компонент  $G_k \subset G_Y$ , образованных разрезами  $E' \subset E$  и  $E'' \subset E$ , смежными относительно опорных вершин  $v_i' \in V'$ :

$$(\forall v_i' \in V') \leftrightarrow (\exists! e_i' \in E') \wedge (\exists! e_i'' \in E'').$$

Каждая компонента  $G_k$  должна обладать структурной завершённостью, которая состоит в сохранении целостности каждого элемента  $a_i \in A$  при его воспроизведении деревьями. Разработанная процедура

формирования блоков КТМ следующая.

1. Определение направления анализа графа  $G_Y = (V, E)$  и нумерация его элементов таким образом, чтобы любая его компонента, полученная произвольным разрезом, имела непрерывную (сплошную) сквозную нумерацию вершин и ребер.

2. Унификация мощностей множеств, задаваемых векторами  $u_{ij}^g, u_{ej}^g$  ( $i = \overline{1, p}, j = \overline{1, q}$ ), путём дополнения их нулевыми элементами до мощности самого мощного вектора  $u_{max}^g$ :

$$\forall (\overline{u_{vi}^g}, \overline{u_{ej}^g}) \in U^G \rightarrow [\overline{u_{vi}^g}] = [\overline{u_{ej}^g}] = [\overline{u_{vi+i}^g}] = [\overline{u_{ej+i}^g}] = [\overline{u_{max}^g}] = r.$$

3. Определение множества упорядоченных подмножеств опорных вершин  $V' = \{V'_i\}$  для разбивки графа  $G_Y$  на компоненты следующим образом:

- нумерация  $t$  множеств  $V'_i \subset V'$  выполняется по направлению анализа графа;

- каждое подмножество  $V'_i \subset V'$  должно формировать по принципу смежности относительно своих элементов (вершин) два простых разреза  $E'_i$  и  $E''_i$ ;

- при условии  $A \leftrightarrow G_Y^h$  или  $A_2 \leftrightarrow G_Y^p$  опорные вершины  $v_i' \in V'$  выбираются таким образом, чтобы они ограждали соответствующие деревья.

4. Определение разрезов  $E'_i = \{e'_i\}$  и  $E''_i = \{e''_i\}$ , соответствующие элементы  $e'_i, e''_i$  которых смежны относительно вершин  $v_i'$  по следующим принципам:

- для опорных вершин множеств  $V'_x \subset V'_i \subset V'$  с нечётными номерами ребра  $e''_i$ , инцидентные вершинам  $v_i' \in V'_x$ , назначаются первыми в направлении анализа графа  $G_Y$  относительно опорных вершин  $v_i' \in V'_x$ , а ребра  $e'_i$  – вторыми;

- для опорных вершин множеств  $V'_y \subset V'_i$  с чётными номерами – наоборот, ребра  $e'_i$  назначаются первыми в направлении анализа графа  $G_Y$  относительно соответствующих опорных вершин  $v_i' \in V'_y$ , а ребра  $e''_i$  – вторыми.

Приведенные правила можно выразить следующими отношениями  $f^{t,1}, \phi_{1,2}^{t,1}$ :

$$\begin{aligned} f^t(E', E'') &= f^{-1}(E', E'') = [(e' \in E') \wedge (e'' \in E'')] \Downarrow [v_i' \in V'_i] \\ \phi^t(V'_i) &= \phi^{-1}(E'_i) = \begin{cases} (e' \in E'_i) \Downarrow (v_i' \in V'_i), & \text{если } t = x \\ (e' \in E'_i) \Uparrow (v_i' \in V'_i), & \text{если } t = y \end{cases} \\ \phi^t(V'_i) &= \phi^{-1}(E''_i) = \begin{cases} (e'' \in E''_i) \Uparrow (v_i' \in V'_i), & \text{если } t = x \\ (e'' \in E''_i) \Downarrow (v_i' \in V'_i), & \text{если } t = y \end{cases} \end{aligned} \quad (3)$$

где  $x, y$ , соответственно, чётные и нечётные значения номера (индекса)  $t$ ; « $\Downarrow$ » – условный символ смежности; « $\Downarrow$ » – условный символ инцидентности ребра «до» опорной вершины; « $\Uparrow$ » – условный символ инцидентности ребра «после» опорной вершины.

5. Разбивка графа  $G_Y$  на два упорядоченных множества компонент  $\{G_k^{E'}\}$  и  $\{G_k^{E''}\}$ , элементы которых обладают свойствами (3). Номер  $k$  присваивается им в порядке следования в направлении анализа графа, начиная с единицы.

6. Составление блоков КТМ для компонент  $G_u^{E'}$  и  $G_w^{E''}$ , где  $u$  и  $w$  – соответственно чётные и нечётные значения индекса  $k$ :

$$|M_{k_{tm},k}| = \begin{matrix} \overline{m_{11}^k} & \overline{m_{12}^k} & \dots & \overline{m_{1c}^k} \\ \overline{m_{21}^k} & \overline{m_{22}^k} & \dots & \overline{m_{2c}^k} \\ \dots & \dots & \dots & \dots \\ \overline{m_{d1}^k} & \overline{m_{d2}^k} & \dots & \overline{m_{dc}^k} \end{matrix}, \quad \begin{matrix} i = \overline{1, c}, p \geq r \\ j = \overline{1, d}, d \geq r \end{matrix} \quad (4)$$

где  $\overline{m_{ij}^k}$  – тополого-параметрические векторы;  $n_e$ ,  $n_v$  – номера, соответственно, ребер и вершин полной КТМ, которым отвечают строка и столбец, в котором размещен элемент  $\overline{m_{ij}^k}$ ;  $u_\chi^e$ ,  $u_\psi^v$  – весовые параметры соответствующему элементу  $\overline{m_{ij}^k}$  ребра и вершины (индексы  $\chi$  и  $\psi$  определяют номера этих элементов в весовых векторах  $\overline{u_{vi}^g}, \overline{u_{ej}^g}$ );  $\theta$  – топологическое свойство соответствующего элемента  $\overline{m_{ij}^k}$  ребра и вершины.

Значение элементов  $u_\chi^e$ ,  $u_\psi^v$  назначаются по следующему правилу:

$$u_\chi^e = \begin{cases} u_1^e, \text{ если } j=1+rh \\ u_2^e, \text{ если } j=2+rh \\ \dots \\ u_r^e, \text{ если } j=r+rh \end{cases}, \quad u_\psi^v = \begin{cases} u_1^v, \text{ если } i=1+rs & h=0,1,\dots, \lfloor \frac{c}{r} \rfloor \\ u_2^v, \text{ если } i=2+rs \\ \dots \\ u_s^v, \text{ если } i=r+rs & s=0,1,\dots, \lfloor \frac{d}{r} \rfloor \end{cases} \quad (5)$$

На основании формулы (5) минимальная размерность каждой матрицы  $|M_{k_{tm},k}|$  в формуле (4), обеспечивающая покрытие всех элементов векторов  $\overline{u_{vi}^g}, \overline{u_{ej}^g}$  для каждой вершины и ребра графа  $G_Y$ , должна составлять  $r \times r$ :  $c_{\min} = r$ ,  $d_{\min} = r$ .

Топологическое свойство  $\theta = 0 \vee \pm 1$  определяется типом матриц и отвечает прямым или опосредствованным отношениям связи элементов графа  $G_Y$ . Наиболее наглядным является использование свойства инцидентности.

Сформированные по формулам (4), (5) блоки воспроизводят множество  $Y \leftrightarrow G_Y$  и могут использоваться для составления программ синтеза ЭСМ на ЭВМ.

### Алгоритм машинного синтеза ЭСМ

Для машинного синтеза ЭСМ на базе блоков КТМ (4) необходимо обратное воспроизведение графа  $G_Y$  в памяти ЭВМ. Специфика его разбивки предусматривает наличие общих вершин  $\{V_x, V_y\}$  в компонентах  $G^{E'}$  и  $G^{E''}$ , откуда вытекает следующая форма сплошной матрицы  $|M_{k_{tm},k}|$  размерностью  $p \times q$

$$|M_{k_{tm},k}| = \begin{matrix} |M_{k_{tm},1}| & |M_{k_{tm},2}| & \dots & |M_{k_{tm},k_q}| \\ |0| & |f_{i,j}| & \dots & |0| \\ |0| & |0| & \dots & |f_{i,j}| \\ \dots & \dots & \dots & \dots \\ |0| & |0| & \dots & |f_{i,j}| \end{matrix} \quad (6)$$

где  $|M_{k_{tm},1}|, |M_{k_{tm},2}|, \dots, |M_{k_{tm},k_q}|$  – блоки КТМ с исключенными столбцами, которые имеют общие номера (индексы) с сопредельными блоками;  $|D_{1,2} M_{k_{tm},1}|, |D_{1,2} M_{k_{tm},1}|, |D_{1,2} M_{k_{tm},1}|, |D_{1,2} M_{k_{tm},1}|$  – блоки КТМ, которые имеют общие номера столбцов с сопредельными блоками;  $|0|$  – нуль-матрицы соответствующих размеров, которые дополняют матрицы компонент к размерности полной КТМ.

Согласно теории матриц выражение (8) представляется прямой суммой

$$|M_{k_{tm},k}| = |M_{k_{tm},1}| \oplus \left( \bigoplus_{k=2}^q |D_{1,2} M_{k_{tm},k}| \oplus |M_{k_{tm},k}| \right) + \left( \bigoplus_{k=1}^{k_q-1} |2_{k_{tm},k}| \right) \oplus |0|_{k_q} \quad (7)$$

где  $|0|_{k_q}$  – нуль-матрица, размер которой совпадает с матрицей  $|D_{1,2} M_{k_{tm},k}|, |M_{k_{tm},k}|$ .

После обратного синтеза КТМ  $|M_{k_{tm},k}|$  может, при необходимости (для унификации введения и обработки), претерпеть следующие изменения:

- заполнение позиций нуль-матриц элементами  $\overline{m_{ij}^k}$  по формулам (4), (5) с учетом изменения параметров  $h$  и  $s$  в формуле (4) по последовательностям  $h=0,1,\dots, \lfloor p/r \rfloor$  и  $s=0,1,\dots, \lfloor q/r \rfloor$  в связи с расширением КТМ до размера  $p \times q$ ;

- исключение параметров  $n_v, n_e \in \overline{m_{ij}^k}$  как таких, что теряют смысл в пределах полной КТМ  $|M_{k_{tm},k}|$ , которая имеет сплошную нумерацию строк и столбцов.

В результате позиции КТМ заполняются векторами  $\overline{m_{ij}^{k_{G_Y}}} = \overline{m_{ij}^k} / \{n_v, n_e\}$ , а полная КТМ определяется так:

$$|M_{k_{tm},k}| = |m_{ij}^{k_{G_Y}}|, \quad i = \overline{1, k_q}, \quad j = \overline{1, n_p}$$

Выражения (6), (7) представляют основу для составления алгоритмов и программ машинного синтеза ЭСМ для АСУТП каждого объекта.

При единстве базового ПО однотипных объектов, машинный синтез ЭСМ и её интеграция с ПО выполняется по следующей схеме (рис. 1):

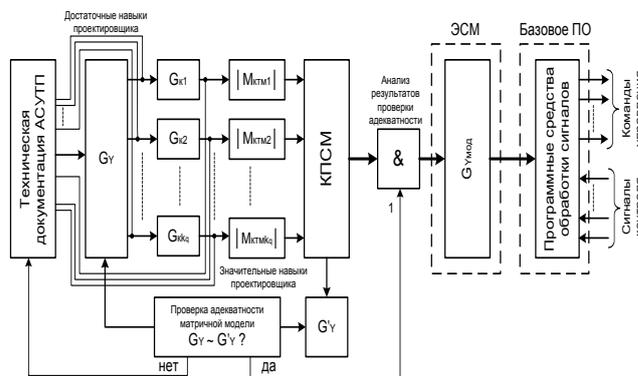


Рис. 1. Функциональная схема машинного синтеза ЭСМ

На основании анализа технической документации на АСУТП соответствующего объекта проектировщиком составляется граф  $G_Y$ , разбиваемый на компоненты  $G_k$ , для которых составляются матрицы  $|M_{k_{tm},k}|$ . При достаточных навыках проектировщика компоненты  $G_k$  могут быть сформированы непосредственно на основе анализа технической документации. При значительном опыте работы проектировщика блоки  $|M_{k_{tm},k}|$  могут быть сформированы в обход процедур составления графа  $G_Y$  и его компонент  $G_k$ .

На основе блоков  $|M_{k_{tm},k}|$  компьютерная программа синтеза ЭСМ (КПСМ), в алгоритм которой заложены методы обратного синтеза графической модели  $G_Y$  на основе формулы (7), формирует конфигурационный файл АСУТП или её испытательной модели с проверкой адекватности аналитической модели, представленной в виде упорядоченного множества  $\{|M_{k_{tm},k}\}$ . Адекватность достигается когда контрольный граф

$G'_Y$ , визуализирующий полную матрицу  $|M_{kmmG'_Y}|$  и обеспечивающий индикацию работы КПСМ, изоморфен исходному графу  $G_Y$ , а упорядоченные множества весовых коэффициентов их элементов равны [3]

$$G_Y \sim G'_Y \leftrightarrow \begin{cases} G_Y^\perp \sim G'^\perp_Y \\ (U^G \subset G_Y) = (U^{G'} \subset G'_Y) \end{cases}, \quad (8)$$

где  $G_Y^\perp, G'^\perp_Y$  – анатомические составляющие исходного и контрольного графов.

Проверка адекватности по критерию (8) может выполняться двумя способами:

- самим проектировщиком или другим лицом на основе анализа и сравнения графов  $G_Y$  и  $G'_Y$  (графа  $G'_Y$  и документации при достаточных навыках работы);

- специальной подпрограммой обработки графических и аналитических данных, в которой заложен алгоритм, реализующий выражение (8).

Нарушение условия (8) свидетельствует о допущении проектировщиком ошибок или неточностей на некотором этапе формирования модели.

Положительные результаты проверки адекватности, подтвержденные соответствующим блоком анализа, приводят к формированию модифицированного графа  $G_{Ymod}$  ( $G_Y \rightarrow G_{Ymod}$ ), заложенного в основу ЭСМ. Модификация графа  $G_Y$  в граф  $G_{Ymod}$  состоит в представлении его в специализированной форме, подставляемой в конфигурационный файл.

Реализация разработанного метода синтеза ЭСМ на основе исходных блоков матрицы  $|M_{kmmG_Y}|$  требует уточнения процедуры нахождения их прямых сумм по формуле (7). Дело заключается в том, что большинство литературных источников по теории матриц определяют операцию прямой суммы только по отношению к квадратным матрицам. Лишь некоторые работы в этом направлении, в частности, распространяют данную операцию на матрицы произвольной размерности. Однако, доступные пакеты разработки ПО не содержат в стандартной библиотеке функций операции прямой суммы матриц (массивов данных) произвольной формы. Это требует разработки метода опосредствованного её нахождения.

Для решения этой задачи можно воспользоваться свойствами обычной суммы матриц. С этой целью формулы (6), (7) можно объединить в таком выражении [4]

$$|M_{kmmG_Y}| = \begin{vmatrix} |M_{\dot{x}_1}| & |M_1| & |0| & |0| & |0| \\ |0| & |0| & |0| & |0| & |0| \\ \dots & \dots & \dots & \dots & \dots \\ |0| & |0| & |0| & |0| & |0| \end{vmatrix} + \begin{vmatrix} |0| & |0| & |0| & |0| & |0| \\ |0| & |r_1, \dot{x}_2| & |M_2| & |M_{\dot{x}_2}| & |M_0| \\ \dots & \dots & \dots & \dots & \dots \\ |0| & |0| & |0| & |0| & |0| \end{vmatrix} + \dots + \begin{vmatrix} |0| & |0| & |0| & |0| & |0| \\ |0| & |0| & |0| & |0| & |0| \\ \dots & \dots & \dots & \dots & \dots \\ |0| & |0| & |0| & |r_1, \dot{x}_{k_q}| & |M_{k_q}| \end{vmatrix} \quad (9)$$

На основании формул (4)–(9) составлен

алгоритм работы КПСМ (рис. 2)

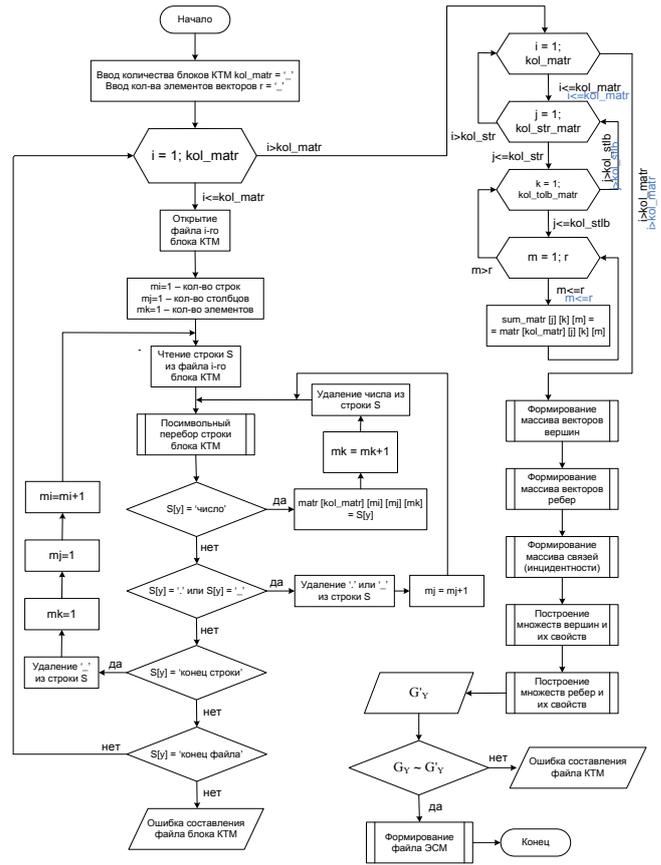


Рис. 2. Алгоритм машинного синтеза ЭСМ

Таблица. Основные опции панели пользователя КПСМ

Наименование опции	Выполняемая функция
Количество блоков КТМ	Установка количества операнд прямой суммы матриц
Индикация параметров	Отображение размерностей блоков КТМ и весовых векторов
Загрузка блоков КТМ	Считывание данных из текстовых файлов блоков КТМ
Синтез КТМ	Подсчёт прямых сумм по формуле (9)
Модификация графа	Формирование конфигурационного файла
Окно контрольного графа	Отображение окна индикации работы КПСМ
Вершины	Визуализация множества вершин и их весовых векторов
Ребра	Визуализация множества ребер и их весовых векторов

По приведенному алгоритму в среде Delphi 7.0 на языке Pascal разработана КПСМ, список опций панели управления которой приведен в таблице. В результате последовательного нажатия опциональных кнопок панели пользователя формируется граф  $G_{Ymod}$

в виде конфигурационного файла в нужном формате. Каждая матрица вводится в программу в виде входного текстового файла.

В зависимости от принципов реализации прикладного ПО могут применяться различные способы считывания данных из конфигурационного файла.

#### **Практическое применение и адекватность моделей**

В настоящее время разработанные метод и программа применяются для формирования компьютерных моделей, предназначенных для отладки и испытаний на функциональную безопасность и безотказность микропроцессорных систем централизованного управления стрелками и сигналами станций промышленного железнодорожного транспорта. Адекватность метода и создаваемых им моделей подтверждается успешной эксплуатацией систем управления станций «Передача-Донецк» ПрАО «Донецксталь – Metallургический завод» (с мая 2011 г.), «Полугорки» ПАО «Ясиновский КХЗ» (с ноября 2011 г.) и «Транзитная» ООО «Димитровпогрузтранс» (с декабря 2011 г.).

Поведение этих станционных АСУТП, зафиксированное в ходе эксплуатационных испытаний на этапе пуско-наладочных работ, а также сбора данных электронных протоколов работы в процессе эксплуатации, соответствует результатам лабораторного моделирования.

#### **Заключение**

Предложенный метод синтеза ЭСМ обладает следующими достоинствами: универсальность применения для АСУТП любых распределённых объектов, простота реализации с позиции квалификации проектировщика (которому достаточны знания о технологии работы объекта управления и его свойствах, кодируемых в виде весовых векторов), возможность локализации пространственных областей объекта на разных этапах проектирования, возможность формирования испытательных моделей только для определённой части крупного объекта.

В то же время, основным недостатком метода следует считать наличие минимального порога размерности блоков КТМ, что при наличии множественных свойств у элементов технологического объекта может привести к чрезмерной величине отдельных локальных областей. Устранение этого недостатка возможно путём суперпозиции локального и глобального объектов, которые могут интерпретироваться соответственно вложенной и внешней матрицами. Реализация такого подхода требует дополнительного исследования в этом направлении.

*Поступила 06.12.2012*



## **К 60-летию со дня рождения**

### **Александра Павловича Крячко**

10 мая исполняется 60 лет к.т.н., руководителю НПП «Днепрчерметавтоматика» Крячко Александру Павловичу.

Со времени окончания института до 1984 г. работал в Головном информационно-вычислительном центре Минчермета УССР.

С 1984 г. по 2003 г. работал в Научно-исследовательском и опытно-конструкторском институте автоматизации черной металлургии, где

прошел все стадии научного роста от заведующего группой до директора института (1997 г.).

В разное время Крячко А.П. был руководителем работ по созданию АСУП Белорусского и Молдавского металлургических заводов.

Будучи директором АО «НИИАчермет» в 1997-2003 гг. А.П. Крячко проявил исключительную настойчивость, целеустремленность и организаторские способности для сохранения и развития этой научно-технической организации.

А.П. Крячко - автор более 50 опубликованных научных работ, 6 авторских свидетельств и патентов.

В настоящее время он возглавляет Научно-производственное предприятие «Днепрчерметавтоматика», где продолжает активную работу по разработке и внедрению средств и систем автоматизации в черной металлургии. Является редактором раздела «Автоматизация» в журнале «Металлургическая и горнорудная промышленность».

***В день юбилея примите от нас пожелания Вам здоровья на долгие годы и счастья. Желаем творческой активности и личного благополучия, успехов в реализации задуманных планов.***

***Совет и редколлегия журнала.***