

Научная статья
УДК 004.94
doi: 10.17223/19988605/62/3

Матрично-топологический анализ компонентных цепей

Вячеслав Михайлович Дмитриев¹, Тарас Викторович Ганджа²,
Максим Игоревич Кочергин³

^{1, 2, 3} Томский государственный университет систем управления и радиоэлектроники, Томск, Россия

¹ dmitriewvm@gmail.com

² gtv@main.tusur.ru

³ maksim.i.kochergin@tusur.ru

Аннотация. Предложен теоретико-множественный формализм метода компонентных цепей, позволяющий строить матрично-топологическое описание объекта моделирования. Введены основные понятия, определяющие компонент, его модель, а также состав и содержание связей компонента с системой. Определена роль матричных моделей в методах компонентного моделирования. Даны канонические формы для моделей компонентов и отражена их иерархия. Введенные формальные понятия компонента и компонентной цепи, а также законов сохранения позволяют построить ее матрично-топологическую модель.

Ключевые слова: математическое моделирование; компонентные цепи; компонент; топология цепи.

Для цитирования: Дмитриев В.М., Ганджа Т.В., Кочергин М.И. Матрично-топологический анализ компонентных цепей // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2023. № 62. С. 25–35. doi: 10.17223/19988605/62/3

Original article
doi: 10.17223/19988605/62/3

Matrix topological analysis of component circuits

Vyacheslav M. Dmitriev¹, Taras V. Gandzha², Maxim I. Kochergin³

^{1, 2, 3} Tomsk State University of Control Systems and Radioelectronics, Tomsk, Russian Federation

¹ dmitriewvm@gmail.com

² gtv@main.tusur.ru

³ maksim.i.kochergin@tusur.ru

Abstract. A set theoretic formalism of the component circuits method is proposed, which allows to construct a matrix-topological description of the simulation object. The paper introduces basic concepts defining the component, its model, as well as the composition and content of the component's links with the system. The role of matrix models in the methods of component modeling is determined. Canonical forms for component models are given and their hierarchy is reflected. Introduced formal concepts of a component and a component circuits, as well as conservation laws allow building its matrix topological model.

Keywords: mathematical modeling; component circuits; component; circuit topology.

For citation: Dmitriev, V.M., Gandzha, T.V., Kochergin, M.I. (2023) Matrix topological analysis of component circuits. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie, vychislitel'naja tehnika i informatika – Tomsk State University Journal of Control and Computer Science*. 62. pp. 25–35. doi: 10.17223/19988605/62/3

Введение

За последние десятилетия создано довольно много систем имитационного и компьютерного моделирования, перечень которых приведен на официальном сайте Национального общества имитационного моделирования [1]. Рассмотрим некоторые из них.

Система Rand Model Designer [2] (ранее MvStudium [3]) позволяет строить модели многокомпонентных гибридных (дискретно-непрерывных) и непрерывных систем и проводить на их основе активные вычислительные эксперименты. Математические модели структурных блоков задаются в символьном виде. Система AnyLogic поддерживает такие подходы к разработке моделей, как процессно-ориентированный (дискретно-событийный), агентный и системно-динамический, а также разные их комбинации. Указанная среда применяется довольно широко в бизнес-моделировании компаний и в образовании [4]. Simulink [5] – это система, входящая в математический пакет Matlab, осуществляющая блочное моделирование с ориентированными и неориентированными связями у блоков. Simulink имеет обширную библиотеку моделей блоков из различных областей и программное обеспечение для анализа системно-событийной динамики, предназначенное для построения имитационных моделей, анализа и оптимизации динамических систем [6]. Для моделирования схем аналоговой, цифровой и силовой электроники может применяться среда Multisim [7]. Программный комплекс SimInTech [8] (ранее MBТУ – Моделирование в технических устройствах [9]) имеет акцент на моделирование объектов, представленных структурными схемами САУ. Он используется в качестве инструмента для проектирования систем следящих приводов, автоматического управления и роботов-манипуляторов, а также ядерных и тепловых энергетических установок и пр. Одна из ранних отечественных систем – среда многоуровневого моделирования СМ МАРС [10], которая предназначена для автоматизации натуральных и вычислительных экспериментов над сложными управляемыми системами. В большинстве своем названные системы базируются на подходах компонентного моделирования [11].

В данной работе предлагается полный формализм метода компонентных цепей (МКЦ), лежащий в основе СМ МАРС, который ранее не рассматривался. Он приведен для цепей с однородными потоками в связях и установленного заранее класса модели объекта.

1. Математическая модель фирмы

Для разработки общих алгоритмов в МКЦ необходимо формализовать основные понятия к вводимым определениям, согласуя их с требованиями конструктивности, универсальности и программной ориентированности.

Пусть компонент M с n связями (выводами) B_0, \dots, B_{n-1} входит в систему, описываемую Q физическими переменными V^1, V^2, \dots, V^Q . Каждой связи $B_i, i = 0, 1, \dots, n - 1$, соответствует индексное множество $I^i = \{I_1^i, I_2^i, \dots, I_{q_i}^i\}, q_i \leq Q$, задающее типы переменных, действующее на связи B_i . Моделью компонента M будем называть систему уравнений (соотношений), связывающих переменные, определяемые индексным множеством $\bigcup_{i=0}^{n-1} I^i$, то есть переменные $V[I^0], V[I^1], \dots, V[I^m]$.

2. Матричные модели в методах компонентного моделирования

Систему уравнений, описывающую поведение компонента, будем называть моделью компонент, или M -моделью. Эта модель обычно расписывается в виде ряда формул и подлежит либо ручной обработке, либо обработке с применением специальных генераторов модели [12]. Будем расписывать M -модели в форме:

$$\text{ММ [}<\text{название модели}>]: \{F_1, F_2, \dots, F_Q\}, \quad (1)$$

где $\{F_1, F_2, \dots, F_Q\}$ – уравнения модели.

При реализации алгоритмов численного анализа, когда модель M получается в результате выполнения некоторого вычислительного процесса, необходимо ввести понятие вычислительной модели (ВМ), которая задается в форме:

$$\text{ВМ [}<\text{название модели}>]: \{k, I^0, I^1, \dots, I^{n-1}, B_1, B_2, \dots, B_k\}, \quad (2)$$

где k – число связей компонента M ; I^0, I^1, \dots, I^{n-1} – индексы переменных, действующие на связях компонента; B_1, B_2, \dots, B_k – вычислительные блоки, соответствующие уравнениям F_1, F_2, \dots, F_Q . Кроме внешних параметров (переменных) компонент характеризуется и внутренними параметрами (конструктивные, технологические, геометрические и др.).

Величина k , задающая число уравнений в M -модели (1) или число вычислительных блоков в B -модели (2) в общем случае не определяется однозначно. Она зависит от типа уравнений и числа переменных в связях.

Для построения вычислительных алгоритмов анализа цепей необходимо зафиксировать основные канонические формы, которые может принимать M -модель компонента.

3. Канонические формы для моделей компонентов и их иерархия

Модель компонента M с внешними параметрами $V_\alpha, \alpha \in \bigcup_{i=0}^{n-1} I^i$ есть система уравнений, которая может принимать следующие канонические формы.

А. Система линейных алгебраических уравнений

$$\sum_{\alpha} a_{\alpha} V_{\alpha} = C_V, \quad (3)$$

где a_{α} и C_V – переменные коэффициенты, определяемые внутренней структурой компонентов.

Б. Система линейных алгебраических уравнений

$$\sum_{\alpha} b_{\alpha} V_{\alpha} = C_V, \quad (4)$$

где b_{α} – постоянные коэффициенты; C_V – переменные правые части, определяемые входными воздействиями.

В. Система уравнений

$$V_{\alpha} = C_V, \quad (5)$$

где C_V – переменные значения V_{α} .

Г. Система уравнений

$$\sum_{\alpha} b_{\alpha} V_{\alpha} = C_C, \quad (6)$$

где b_{α} и C_C – постоянные коэффициенты.

Д. Система уравнений

$$V_{\alpha} = C_C, \quad (7)$$

где C_C – постоянные значения V_{α} .

Линейные модели (3)–(7) задаются непосредственно в качестве моделей компонентов с привлечением аналитических преобразований [13]. Коэффициенты (3)–(7) можно также получить в результате работы Б-блоков у В-моделей.

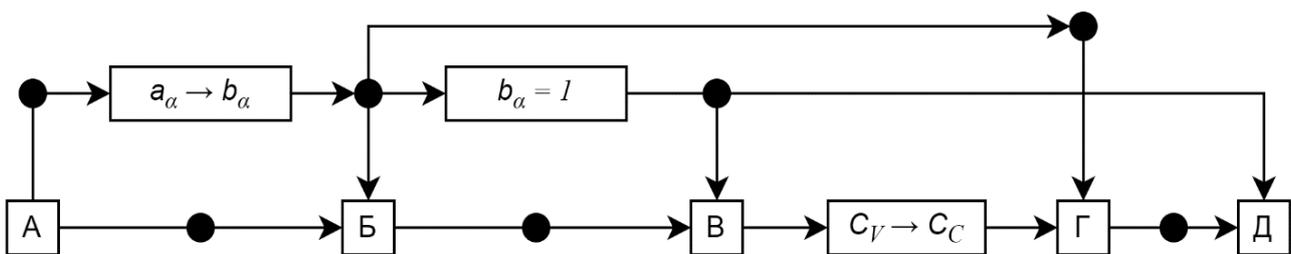


Рис. 1. Иерархическая структура моделей
Fig. 1. Hierarchical structure of models

Очевидно, что модели А, Б, В, Г, Д связаны в иерархическую структуру (рис. 1).

4. Примеры канонических форм моделей

Приведем примеры канонических форм моделей компонентов [14]:

– динамическая нелинейная модель диода (А-форма)

$$C_b \left(\frac{du_1}{dt} - \frac{du_2}{dt} \right) = i_1 - Si \left(e^{\frac{u_1 - u_2}{V_i}} - 1 \right) - \frac{1}{Rb} (u_1 - u_2 - Rmi_1);$$

– модель емкости (Б-форма)

$$\frac{d(u_1 - u_2)}{dt} = \frac{1}{C} i_1;$$

– модель синусоидального источника (В-форма)

$$u_1 - u_2 = A \sin(\omega t + \varphi);$$

– модель резистора (Г-форма)

$$u_1 - u_2 - Ri_1 = 0;$$

– модель постоянного источника (Д-форма)

$$u_1 - u_2 = E.$$

5. Формальное понятие компонента и компонентной цепи

На основании понятия компонента и его модели вводится понятие компонентной цепи (КЦ). Компонентной цепью будем называть произвольную совокупность компонентов, связи которых, именуемые ветвями цепи B_0, \dots, B_{n-1} , объединены в точках, называемых узлами N_0, \dots, N_{n-1} . Каждому узлу цепи соответствуют законы сохранения энергии.

Очевидно, что в зависимости от природы цепи узловой закон сохранения может принимать самые различные формы, однако для построения вычислительных алгоритмов форма законов сохранения должна быть зафиксирована.

Решением модели цепи, состоящей из компонентов M_1, \dots, M_i , будем называть совокупность переменных, действующих на ветвях цепи (8)

$$V = \{ \bigcup (\bigcup V_{M_j}^{I_{M_j}^i}) \}, \quad (8)$$

где $I_{M_j}^i$ – индексы переменных, действующих на i -й связи компонента M_j , В решении V переменные $V[\alpha]$ и $V[\beta]$ будем называть одноименными, если $\alpha = \beta$.

Пусть узел N_i , ($i = 1, \dots, k-1$) объединяет ветви, на которых действуют переменные $V_1[\alpha_1], V_2[\alpha_1], \dots, V_{i_1}[\alpha_1], V_1[\alpha_2], V_2[\alpha_2], \dots, V_{i_2}[\alpha_2], \dots, V_1[\alpha_q], V_2[\alpha_q], \dots, V_{i_q}[\alpha_q]$.

Полагаем, что цепи допускают закон сохранения, заданный в виде равенства одноименных переменных (9) или (10):

$$V_1[\alpha_j] = V_2[\alpha_j] = \dots = V_{i_j}[\alpha_j], \quad j = 1, \dots, q, \quad (9)$$

$$\sum_{k=1}^{i_j} a_k^j V_k[\alpha_j] = 0, \quad j = 1, \dots, q. \quad (10)$$

Можно потребовать, чтобы закон сохранения принял вид:

$$\sum_{k=1}^{i_j} V_k[\alpha_j] = 0, \quad j = 1, \dots, q, \quad (11)$$

т.е. $a_k^j = 1$. Однако для некоторых классов цепей представление вида (11) недопустимо, так как связи компонентов (ветви цепи) могут иметь естественную или общепринятую классификацию.

Очевидно, что при узловых законах вида (9) переменные ветвей цепи можно заменить на узловые переменные. Линейная форма (10) соответствует закону сохранения для переменной $V[\alpha_j]$.

Для разработки методик моделирования и анализа необходимо формальное описание компонентной цепи. Полагаем, что компонентам цепи присвоены номера $1, \dots, l$ (компоненты имеют идентификаторы M_1, M_2, \dots, M_l). Связи компонента $M_i, i = 1, \dots, l$, нумеруются в произвольном и затем фиксируемом порядке. Существует три способа задания компонентной цепи: списковый, графический и аналитический. Рассмотрим здесь первый способ.

Будем полагать, что цепь задана, если имеется список, строка которого имеет вид:

$$M_i, \text{sign}_i^j B_i^j, N_i^j, \quad (12)$$

где $i=1, \dots, n_j, j=1, \dots, l$.

Такое представление именуем далее списковым, или С-цепью.

Знак sign_i^j , где $i=1, \dots, n_j, j=1, \dots, l$, задает ориентацию ветви B_i^j . Очевидно, что при формировании узлового закона сохранения (10) a_k^j соответствуют закону сохранения при определенной ветви. В силу однородности (10) способ установления знака безразличен.

Очевидно, что в (12) $B_i^j, N_i^j, j=1, \dots, l; i=1, \dots, n_j$ задают соответственно ветвь, образованную i -й связью компонента M_j , и узел, в который ведет эта ветвь. Перечисление ветвей в (12) осуществляется в соответствии с фиксированной нумерацией связей компонентов.

Для интерпретации результатов анализа цепей, представление решения V в виде (8) обычно оказывается неудобным. Воспользуемся тем, что ветви цепи имеют сплошную нумерацию B_1, B_2, \dots, B_m , и сгруппируем одноименные переменные:

$$V = V_1[1], \dots, V_m[1], V_1[q]. \quad (13)$$

Очевидно, что цепь является однородной, если выполняется условие

$$I_1^1 = \dots = I_1^{n_1} = \dots = I_1^l = \dots = I_1^{n_l}. \quad (14)$$

Итак, цепь называется однородной, если на каждой ее ветви действует один и тот же набор физических переменных. В противном случае цепь называется неоднородной. Решение неоднородной цепи представляется в виде (8) либо в виде (13). При этом ряд элементов в (13) в таком случае будет отсутствовать.

6. Структура матрично-топологической модели цепи

Моделью цепи будем называть совокупность моделей ее компонентов M_1, M_2, \dots, M_l , внешним параметрам которых поставлены в соответствие элементы решения V согласно С-цепи (12) и узловых законов сохранения (9) и (10).

Для практического построения модели цепи необходимо задать ее класс. Полагаем, что класс цепи задан, если каждому типу физических переменных поставлен в соответствие узловой закон сохранения, т.е. задано множество пар $\Pi_i = (V^i, P^i), i=1, \dots, q, P_i \in \{P_1, P_2\}$. Через P_1 обозначен закон равенства одноименных переменных (9), через P_2 – закон сохранения (10). Очевидно, что всякая цепь может входить в качестве компонента или подцепи в другую цепь.

Рассмотрим матричное представление канонических форм после применения к ним процедур алгебраизации и линеаризации, предназначенное для алгоритмов численного анализа. Матрицей A канонической формы модели цепи будем называть матрицу

$$A = \begin{pmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,q} & a_{1,q+1} \\ a_{2,1} & a_{2,2} & & a_{2,q} & a_{2,q+1} \\ \vdots & & \ddots & & \vdots \\ a_{p,1} & a_{p,1} & & a_{p,q} & a_{p,q+1} \end{pmatrix}. \quad (15)$$

Полагаем, что канонические формы А, Б, В, Г, Д задаются матрицей $[a_{ij}], i = 1, \dots, q + 1; j = 1, \dots, p$ размерности $(q + 1) \times p$. Столбец $(q + 1)$ соответствует вектору правых частей со своими знаками. Другая форма уравнения (15) имеет вид:

$$\begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,q} \\ a_{2,1} & a_{2,2} & & a_{2,q} \\ \vdots & & \ddots & \\ a_{p,1} & a_{p,1} & & a_{p,q} \end{pmatrix} \cdot V_{\text{вн}} = \begin{pmatrix} a_{1,q+1} \\ a_{2,q+1} \\ \vdots \\ a_{p,q+1} \end{pmatrix}, \quad (16)$$

где $V_{\text{вн}}$ – вектор переменных цепи V^α , $\alpha \in \bigcup I^i (i=0, \dots, n-1)$ моделируемого компонента. Вектор $V_{\text{вн}}$ получается из V^α упорядочением в соответствии с фиксированной нумерацией связей и естественным порядком индексов переменных.

Представим (16) в виде:

$$A_1 \cdot V_{\text{вн}} = A_2. \quad (17)$$

Очевидно, что $A = [A_1, A_2]$. К размерности матрицы A предъявляются требования существования и единственности решения цепи, включающих в себя моделируемые компоненты. Рассмотрим основные структурные блоки модели цепи. Узловые законы связей P_1 и P_2 также представляются в матричной форме:

$$PV = 0; \quad P = [P_1 \quad P_2]^t. \quad (18)$$

Матрица P_1 соответствует узловому закону равенства одноименных переменных P_1 , матрица P_2 – закону сохранения P_2 . Элементы матрицы P , очевидно, удовлетворяют условию $P_i^j \in \{0, 1, -1\}$. Для построения (18) формула (9) переписывается в виде:

$$V_1[\alpha_j] - V_2[\alpha_j] = 0, \dots, V_{i,j-1}[\alpha_j] = V_{i,j}[\alpha_j] = 0 \quad j = 1, \dots, q. \quad (19)$$

Матрица P_1 расписывается из коэффициентов (19) для каждого узла цепи. Исходной информацией для построения P_1 служит С-цепь и $\bigcup_{i=1}^q \Pi_i$. При практической реализации метода компонентных цепей матрица P_1 не строится. Совокупность переменных с $\Pi_i = (V^i, P_i)$, относящихся к ветвям, которые ведут в один узел, заменяется согласно системе равенств (9) узловой переменной. В этом случае решение V представляется в виде:

$$V = \bigcup_{i=1}^q V_i, \begin{cases} \Pi_i = (V^i, P_1) & \text{если } \Pi_i = (V^i, P_1), \\ V_i = (V_0^i, \dots, V_{k-1}^i) & \text{если } \Pi_i = (V^i, P_2). \end{cases} \quad (20)$$

Матрица P_2 строится на основе (10). Для каждого из узлов расписывается строка P_2 путем циклического просмотра С-цепи. Ненулевые коэффициенты P_2 рассылаются следующим образом:

$$P_{i,j} = \begin{cases} 1, & \text{sign}_p^q = +1, \quad q = 1, \dots, l; \quad p = 1, \dots, n_i, \\ -1, & \text{sign}_p^q = -1, \quad i = N_q^p, \quad j = B_q^{jp}. \end{cases} \quad (21)$$

С-цепь просматривается для каждой переменной V_i , для которой $\Pi_i = (V^i, P_2)$.

В целом модель компонентной цепи включает в себя матрицы P_1 , P_2 и A -матрицы для каждого компонента цепи.

Матричное уравнение (22) в котором Φ -матрица включает в себя матрицы P_1 и P_2 , а также матрицу A_1 (17) для каждого компонента цепи, будем называть основным уравнением метода компонентных цепей

$$\Phi \cdot V = W. \quad (22)$$

Элементы матрицы A_1 рассылаются в Φ -матрицу в соответствии с С-цепью (12) и установленным порядком переменных в V (13). Вектор W содержит нули в позициях, соответствующих строкам P_1 и P_2 . Элементы W , соответствующие матрицам A_1 , равны элементам матриц A_2 (17). Уравнение (23) и является моделью компонентной цепи с заданными моделями компонентов и узловыми законами сохранения

$$\begin{pmatrix} A_1 \\ P_1 \quad P_2 \end{pmatrix} \cdot V = W. \quad (23)$$

7. Графическая интерпретация компонентной цепи

Будем полагать, что компонентная цепь (КЦ) задана, если имеется список следующего вида:

$$\begin{aligned}
 1: & K_1, \text{sign}_1^1 B_1^1, \dots, \text{sign}_{n_1}^1 B_{n_1}^1, N_1^1, \dots, N_{n_1}^1, \\
 & \vdots \\
 l: & K_l, \text{sign}_1^l B_1^l, \dots, \text{sign}_{n_l}^l B_{n_l}^l, N_l^l, \dots, N_{n_l}^l.
 \end{aligned}
 \tag{24}$$

Знак sign_i^j $j=1, \dots, l; i=1, \dots, n_j$ задает ориентацию соответствующей ветви B_i^j . Полагаем, что связь ведет в компонент, если $\text{sign}_i^j = +1$ (положительное направление ветви), и от компонента, если $\text{sign}_i^j = -1$ (отрицательное направление) (рис. 2).

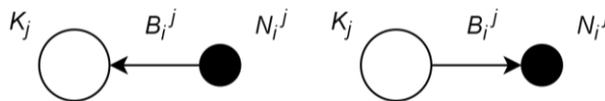


Рис. 2. Варианты ориентации связей
Fig. 2. Link orientation options

Перейдем к графической интерпретации понятия КЦ в виде структурного (бихроматического) графа.

G-графом будем называть граф $G = \{K, N, B, Q\}$, где

- 1) $K = \{K_1, K_2, \dots, K_{l-1}\}$ – множество вершин компонентов графа;
- 2) $N = \{N_1, N_2, \dots, N_{k-1}\}$ – множество вершин – узлов графа;
- 3) $B = \{B_1, B_2, \dots, B_{m-1}\}$ – множество, элементы которого образованы связями компонентов.

Функция Q ставит в соответствие каждой связи $B_j \in B$ упорядоченную пару вершин $(K_i, N_i) \neq (N_i, K_i)$. Ориентированные таким образом связи образуют множество дуг G-графа. Он, в свою очередь, состоит из подграфов. Обозначим их как g – графы компонента. Тогда граф $G = \{\bigcup g_j, N, Q'\}$ – ориентированный граф, где $\bigcup g_j$ – множество, элементы которого образованы объединением g -графов; $N = \{N_1, N_2, \dots, N_{k-1}\}$ – множество вершин – узлов графа; Q' – функция, ставящая в соответствие каждому g -графу упорядоченный набор вершин-узлов.

Для изображения G-графа введем следующие графические обозначения: K и N – множества белых (компоненты) и черных (узлы) вершин. На рис. 3 приведена простая цепь, содержащая компоненты K_1, K_2, K_3, K_4 с узлами N_0, \dots, N_3 с различным числом связей (см. рис. 3, a) и соответствующий цепи G-граф (см. рис. 3, b).

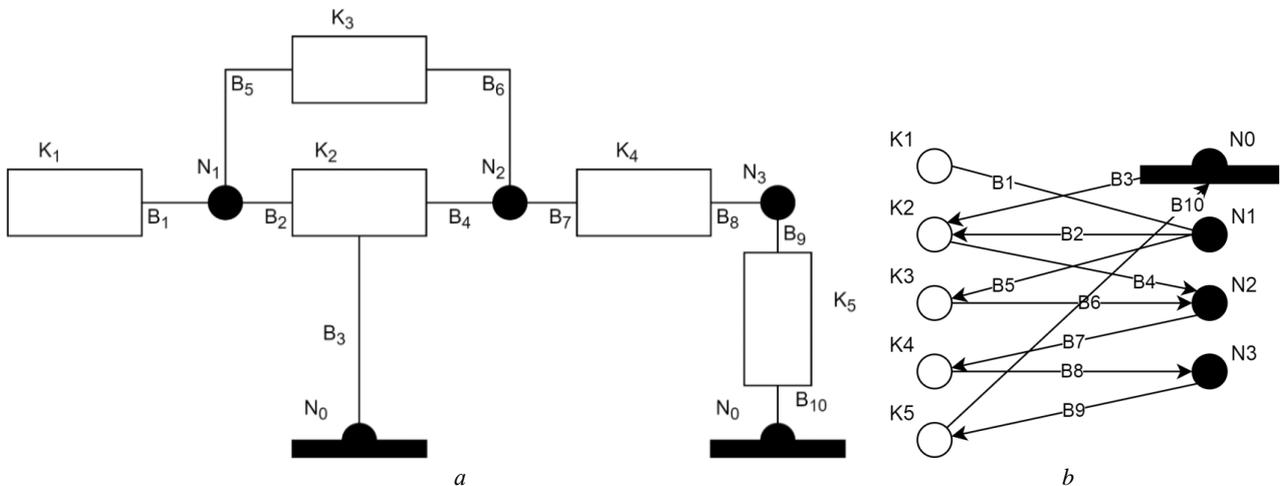


Рис. 3. Компонентная цепь: a – непосредственно компонентная цепь, b – бихроматический граф цепи
Fig. 3. Component chain. a) is directly component chain, b) is the bichromatic graph of the chain

Оперируя приведенными матрицами смежности и инцидентности, интерпретируем свойства графа цепи (см. рис. 3) для определения возможных топологических ошибок при формировании компонентной цепи.

Введем основные топологические обозначения: $\bigcup B_i$ – совокупность ветвей цепи C_k ; $\bigcup N_j$ – совокупность узлов цепи C_k ; $\bigcup_{j \in K_j} N_j$ – совокупность узлов компонента; $\bigcup_{j \in K_j} B_j$ – совокупность связей компонента.

В этих обозначениях можно представить различные типы топологических ошибок (рис. 4):

- а) висячий узел $\bigcup_{i \in N_j} B_i = 1$ (ранг узла; см. рис. 4, а);
- б) неприсоединенный компонент $\bigcup_{i \in N_j} B_i = 0$ (см. рис. 4, б);
- в) узел без ветвей $\bigcup_{i \in N_j} B_i = 0$ (см. рис. 4, в);
- г) петля из связей $(B_i^k, B_j^k) \in N_j$ (см. рис. 4, д).

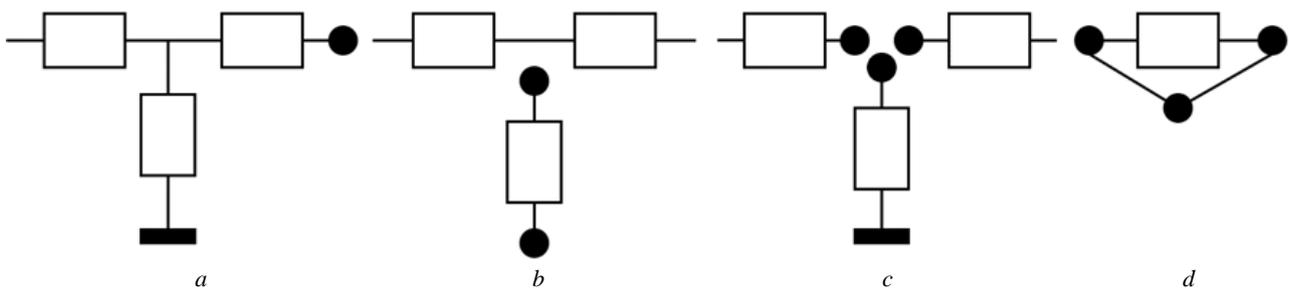


Рис. 4 Типы топологических ошибок
Fig. 4. Types of topological errors

Эти ошибки соответствуют следующим вариантам, применительно к матрицам:

Вариант 1. В матрице G_2 появляется столбец с одним ненулевым элементом.

Вариант 2. В матрице S (смежности) появляется нулевая строка.

Вариант 3. В матрице G_2 появляется столбец только с нулевыми элементами.

Вариант 4. В матрице G_2 появляется столбец с двумя ненулевыми элементами для ветвей одного и того же компонента.

Для данных вариантов могут быть составлены соответствующие алгоритмы контроля топологических ошибок в компонентной цепи.

Заключение

Предложенный в работе теоретико-множественный формализм метода компонентных цепей позволяет строить матрично-топологическое описание объекта моделирования. Введенные понятия определяют компонент, его модель, а также состав и содержание связей компонента с системой. Даны канонические формы для моделей компонентов (модель емкости – Б-форма, модель синусоидального источника – В-форма, модель резистора – Г-форма, модель постоянного источника – Д-форма), позволяющие строить модели систем различной природы, и отражена их иерархия. Введенные формальные понятия компонента и компонентной цепи, а также законов сохранения позволяют построить матрично-топологическую модель компонентной цепи. Приведенная графическая интерпретация компонентной цепи позволяет установить однозначное соответствие между непосредственно компонентной цепью (как моделью некоторой системы) и ее бихроматическим графом, состоящим из узлов и компонентов, соединенных связями. Приведенные четыре случая топологических ошибок в компонентной цепи позволяют выявить их автоматически посредством анализа матриц смежности и инцидентности графа цепи.

Список источников

1. Программные продукты. URL: <http://simulation.su/static/ru-soft.html> (дата обращения: 09.09.2022).
2. Isakov A.A., Kolesov Y.B., Senichenkov Y.B. A new tool for visual modeling-Rand Model Designer 7 // IFAC-Papers On Line. 2015. V. 28 (1). P. 661–662.
3. Senichenkov Yu.B., Kolesov Yu.B. Physical modeling in MvStudium // Differential Equations and Control Processes. 2011. V. 2. P. 34–40.
4. Yakimov I.M., Trusfus M.V., Mokshin V.V., Kirpichnikov A.P. AnyLogic, extendsim and simulink overview comparison of structural and simulation modelling systems // RPC 2018 Proc. of the 3rd Russian-Pacific Conference on Computer Technology and Applications. 2018. Art. 8482152.
5. Deng H., Zhu L., Wang J., He C., Liu J. Hydraulic system simulation of hybrid hydraulic module based on Simulink // Proc. of SPIE – The International Society for Optical Engineering. 2022. V. 12259. Art. 122593M. 9 p.
6. Дьяконов В.П. Simulink 5/6/7. М. : Изд-во ДМК, 2008. 781 с.
7. Peng Z. Comparison and discussion of the functions of logisim and multisim // Journal of Physics: Conference Series. 2021. V. 2030 (1). Art. 012055.
8. Колотыркин И.П., Петухов В.Н. Создание моделей сложных технических систем в среде SimInTech // Компьютерное моделирование в железнодорожном транспорте: динамика, прочность, износ : ст. тез. конф. Брянск : Брянск. гос. техн. ун-т, 2018. С. 45–47.
9. Шабаетв Е.А. Анализ и синтез систем автоматического регулирования на основе программного комплекса «МВТУ» // Научная молодежь: пути и перспективы развития агроинженерной науки: студенческий сб. науч. тр. Зерноград : Азово-Черноморская инж. акад., 2005. С. 13–24.
10. Дмитриев В.М., Шутенков А.В., Зайченко Т.Н., Ганджа Т.В. MARS – среда моделирования технических устройств и систем. Томск : В-спектр, 2011. 277 с.
11. Бенькович Е.С., Колесов Ю.Б., Сениченков Ю.Б. Практическое моделирование динамических систем. СПб. : БХВ – Петербург, 2002. 707 с.
12. Арайс Е.А., Дмитриев В.М. Моделирование неоднородных цепей и систем на ЭВМ. М. : Радио и связь, 1982. 159 с.
13. Арайс Е.А., Сибиряков Г.В. Авто-Аналитик. Новосибирск : Изд-во Высш. и сред. спец. образования РСФСР. 1973. 284 с.
14. Дмитриев В.М., Зайченко Т.Н., Гарганеев А.Г., Шурыгин Ю.А. Автоматизация функционального проектирования электромеханических систем и устройств преобразовательной техники. Томск : Изд-во Том. ун-та, 2000. 292 с.
15. Оре О. Графы и их применение. М. : Мир, 1965. 174 с.

References

1. National Society for Simulation. (n.d.) *Programmnye produkty* [Software]. [Online] Available from: <http://simulation.su/static/ru-soft.html> (Accessed: 9th September 2022).
2. Isakov A.A., Kolesov, Y.B. & Senichenkov, Y.B. (2015) A new tool for visual modeling-Rand Model Designer 7. *IFAC-Papers On Line*. 28(1). pp. 661–662.
3. Senichenkov, Yu.B. & Kolesov, Yu.B. (2011) Physical modeling in MvStudium. *Differential Equations and Control Processes*. 2. pp. 34–40.
4. Yakimov, I.M., Trusfus, M.V., Mokshin, V.V. & Kirpichnikov, A.P. (2018) Any logic, extendsim and simulink overview comparison of structural and simulation modelling systems. *RPC 2018 – Proceedings of the 3rd Russian-Pacific Conference on Computer Technology and Applications*. Art. 8482152. DOI: 10.1109/RPC.2018.8482152
5. Deng, H., Zhu, L., Wang, J., He, C. & Liu, J. (2022) Hydraulic system simulation of hybrid hydraulic module based on Simulink. *Proceedings of SPIE – The International Society for Optical Engineering*. 12259. Art. 122593M. DOI: 10.1117/12.2638851
6. Dyakonov, V.P. (2008) *Simulink 5/6/7*. Moscow: DMK.
7. Peng, Z. (2021) Comparison and discussion of the functions of logisim and multisim. *Journal of Physics: Conference Series*. 2030(1). Art. 012055. DOI: 10.1088/1742-6596/2030/1/012055
8. Kolotyrykin, I.P. & Petukhov, V.N. (2018) Sozdanie modeley slozhnykh tekhnicheskikh sistem v srede SimInTech [Creation of models of complex technical systems in the SimInTech environment]. *Komp'yuternoe mo-delirovanie v zheleznodorozhnom transporte: dinamika, prochnost', iznos* [Computer Modeling in Railway Transport: Dynamics, Strength, Wear]. Proc. of the Conference. Bryansk: Bryansk State Technival University. pp. 45–47.
9. Shabaev, E.A. (2005) Analiz i sintez sistem avtomaticheskogo regulirovaniya na osnove programmno kompleksa “MVTU” [Analysis and synthesis of automatic control systems based on the software complex “MiTD”]. In: *Nauchnaya molodezh': puti i perspektivy razvitiya agroinzhenernoy nauki* [Scientific youth: ways and prospects for the development of agroengineering science]. Zernograd Azovo-Chernomorsk Engineering Academy. pp. 13–24.
10. Dmitriev, V.M., Shutenkov, A.V., Zaychenko, T.N. & Gandzha, T.V. (2011) *MARS – sreda modelirovaniya tekhnicheskikh ustroystv i sistem* [MARS – environment for modeling technical devices and systems]. Tomsk: V-Spekt. r.
11. Benkovich, E.S., Kolesov, Yu.B. & Senichenkov, Yu.B. (2002) *Prakticheskoe modelirovanie dinamicheskikh sistem* [Practical modeling of dynamic systems]. St. Petersburg: BKhV – Peterburg.

12. Arays, E.A. & Dmitriev, V.M. (1982) *Modelirovanie neodnorodnykh tsepey i sistem na EVM* [Modeling of heterogeneous circuits and systems on a computer]. Moscow: Radio i svyaz'.
13. Arays, E.A. & Sibiryakov, G.V. (1973) *Avto-Analitik* [Auto-Analyst]. Novosibirsk: Higher and Secondary Vocational Education of the RSFSR.
14. Dmitriev, V.M., Zaychenko, T.N., Garganeev, A.G. & Shurygin, Yu.A. (2000) *Avtomatizatsiya funktsional'nogo proektirovaniya elektromekhanicheskikh sistem i ustroystv preobrazovatel'noy tekhniki* [Automation of functional design of electromechanical systems and equipment conversion devices]. Tomsk: Tomsk State University.
15. Ore, O. (1965) *Grafy i ikh primeneniye* [Graphs and Their Application]. Moscow: Mir.

Информация об авторах:

Дмитриев Вячеслав Михайлович – профессор, доктор технических наук, профессор кафедры компьютерных систем в управлении и проектировании Томского государственного университета систем управления и радиоэлектроники (Томск, Россия). E-mail: dmitriewvm@gmail.com

Ганджа Тарас Викторович – доцент, доктор технических наук, профессор кафедры компьютерных систем в управлении и проектировании Томского государственного университета систем управления и радиоэлектроники (Томск, Россия). E-mail: gtv@main.tusur.ru

Кочергин Максим Игоревич – кандидат технических наук, доцент кафедры компьютерных систем в управлении и проектировании Томского государственного университета систем управления и радиоэлектроники (Томск, Россия). E-mail: maksim.i.kochergin@tusur.ru

Вклад авторов: все авторы сделали эквивалентный вклад в подготовку публикации. Авторы заявляют об отсутствии конфликта интересов.

Information about the authors:

Dmitriev Vyacheslav M. (Professor, Doctor of Technical Sciences, Professor of the Department of Computer Systems in Control and Design of Tomsk State University of Control Systems and Radioelectronics, Tomsk, Russian Federation). E-mail: dmitriewvm@gmail.com

Gandzha Taras V. (Associate Professor, Doctor of Technical Sciences, Professor of the Department of Computer Systems in Control and Design, Tomsk State University of Control Systems and Radioelectronics, Tomsk, Russian Federation). E-mail: gtv@main.tusur.ru

Kochergin Maxim I. (Candidate of Technical Sciences, Associate Professor, Department of Computer Systems in Control and Design, Tomsk State University of Control Systems and Radioelectronics, Tomsk, Russian Federation). E-mail: maksim.i.kochergin@tusur.ru

Contribution of the authors: the authors contributed equally to this article. The authors declare no conflicts of interests.

Поступила в редакцию 22.09.2022; принята к публикации 01.03.2023

Received 22.09.2022; accepted for publication 01.03.2023