

## МАТЕМАТИЧЕСКОЕ И КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ

УДК 004.45

### МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ТРАНСФОРМАЦИИ ФУНКЦИОНАЛЬНЫХ СТРУКТУР СЛОЖНЫХ ТЕХНИЧЕСКИХ СИСТЕМ

В. А. Бочаров, МГТУ им. Н.Э. Баумана, Москва, Россия;  
e-mail: bocharov.v@bmstu.ru

*Статья посвящена развитию средств и методов системной инженерии, повышающих эффективность автоматизации разработки. Одно из направлений развития – создание языковых и математических моделей, покрывающих неиспользованные свойства систем, в числе которых семантика связей между элементами построения функции и накладываемыми на неё ограничениями. В статье приводится описание нового подхода к построению математической модели системы на основе связей, обеспечивающего тождественные и эквивалентные трансформации её функциональной структуры, что необходимо для формирования технических решений в процессах разработки.*

**Ключевые слова:** системная инженерия, функциональность, математическая модель, сложная техническая система, разработка, автоматизация, трансформация

**DOI:** 10.21667/1995-4565-2025-94-98-105

#### Введение

Данное исследование посвящено решению задачи декомпозиции и интеграции функциональной структуры сложных технических систем (далее – СТС) в процессах их разработки [1]. К данному классу систем относятся системы, у которых нет однозначного соответствия между внутренним строением и внешними свойствами и у которых оно не определяется комбинаторными методами. Суть этой задачи – в замещении исходного представления системы как «чёрного ящика» с неизвестным содержанием на представление с добавленными знаниями о содержании этой системы с учётом следующих обстоятельств. Во-первых, всякая функция принимает данные, отправляет данные, несёт в себе определённый метод – формулу или алгоритм и реализует этот метод некоторым средством – механизмом. Во-вторых, конечная цель достигается, как правило, не одним действием, а несколькими взаимосвязанными. В-третьих, способов реализации метода может быть более одного, и его выбор на одном этапе разработки влияет на постановку задачи разработки последующего. В-четвёртых, всякая функция потребляет ресурсы, из-за чего результаты работы системы можно условно разделить на целевые и сопутствующие – вспомогательные и обеспечивающие. По отношению одних к другим можно судить об эффективности системы. Добавим к этим обстоятельствам ещё одно – непостоянство, изменчивость системы, обусловленные конечностью физических свойств, износом и усталостью материала, исчерпанием расходуемого ресурса и другими факторами. Сам процесс поиска и выбора технических решений сопряжён с неоднозначностью такого выбора и потребностью в метриках, объективно характеризующих вклад решения в конечный результат. В итоге получаем многомерную, параметрическую модель предмета исследования и разработки, решать которую традиционными методами системной инженерии затруднительно, и требуются новые подходы. Попробуем определить такой способ описания функциональности системы, чтобы переход от одного уровня детализации функциональности системы к другому представлял собой тождественную или эквивалентную

трансформацию функциональной структуры, которую можно задать, вычислить и проверить, используя математические методы.

### Образ функциональной структуры системы

Работа системы  $S$  выражается конечным списком фактов:

$S: \{ A1, A2, \dots, An \}$ , где каждый факт выражает предусмотренный результат работы системы, определяемый различиями и/или изменениями, порождаемыми системой. В таком своём качестве каждый факт можно считать параметром одноместного предиката  $P: \{ A \}$ . Таким образом, можно сказать, что система со стороны внешнего наблюдателя – это список фактов, возникающих при её работе, которые могут быть разделены на целевые или сопутствующие (побочные), и оцененных как неизбежные, вероятные или непредсказуемые. Кроме вектора  $S$ , у системы также имеется вербальное определение его предназначения, которое выглядит, как выражение:  $A1$ : «Преобразовать цифровой сигнал  $\varphi$  в аналоговый  $\psi$ ». Оно является источником определения исходных данных, результатов и управляющих воздействий, т. е. портов функции.

Развернув единичный факт работы системы на два множества исходных и результирующих фактов, получим двухместный предикат  $P: \{ X, Y \}$ , где  $X$  – множество аргументов,  $Y$  – множество значений, в котором заключена проекция множества  $X$  на множество  $Y$ . Таким образом, осуществляем переход от факта к конечной разнице, которую он представляет.

Добавив третье множество отношений между аргументами и значениями, получим 3-местный предикат  $P: \{ X, Y, Z \}$ , где  $Z$  – множество методов, обеспечивающих получение заданных значений функции из заданных аргументов, получим трёхместный предикат, отображающий проекцию трёх множеств – аргументов, значений и методов, а также логических граничных условий, о которых будет сказано далее.

Добавив четвёртое множество – средства, реализующих методы, получаем 4-местный предикат:  $P: \{ X, Y, Z, M \}$ , где  $M$  – механизм, в котором заключён способ реализации метода  $Z$ . В таком виде вектор  $P$  уже представляет схему внутреннего строения функции в объёме спецификации IDEF0. Однако мы учитываем ещё дополнительные свойства, вытекающие из закономерностей отношений между аргументами, значениями, методами и средствами:

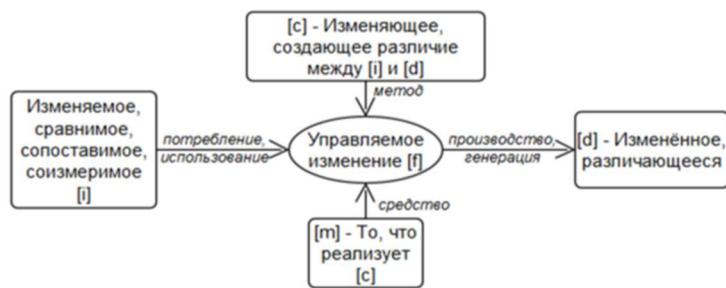


Рисунок 1 – Внутренние семантические связи функции  
Figure 1 – Internal semantic connections of the function

Эти отношения становятся источниками граничных условий для определений функции, которые, в конечном итоге, обеспечат корректность математической модели при её построении.

Добавив логические граничные условия существования функции, основанные на логике предикатов, и которые не предусматривала методология IDEF0 [2, 3], получим:

- $\exists (X, Y, Z)$  – декларация о том, аргумент, значение и метод существуют (заданы);
- $\forall Z (Z \rightarrow X) \cap (Z \rightarrow Y)$  – утверждение о том, что для всех имеющихся методов аргумент и значение являются потомками метода;
- $\nexists Z (X = \emptyset) \cap (Y = \emptyset)$  – условие отсутствия случаев, когда используются или производятся пустые множества. Это обеспечивает отсутствие в составе системы пустых или недействительных элементов.

На этих отношениях строятся константы, аксиомы, постулаты и прочие инварианты модели создаваемой системы.

*Примечание.* Данная статья посвящена функциональному проектированию, основы которого определены в технологии IDEF0. Методологии IDEF3, IDEF1X, IDEF5 и другие также могут быть проанализированы на предмет внутренних семантических связей (определенны условия существования системы в терминах этих связей) и использованы при решении задач анализа и синтеза систем.

### Математическая модель системы

Математическая модель системы использует понятия, выражаемые перечислением и упорядочением его свойств, и указания на методы вывода одних свойств из других. При этом способ организации множества свойств берётся в соответствии с принципами и методами систематизации и классификации по В.В. Омельченко [4].

Математическая модель функциональности системы, предназначенная для решения задач тождественной или эквивалентной декомпозиций функций в процессах их разработки, будет иметь следующие определения.

1. Вводим базовое определение функции общего вида как упорядоченного по ролям набора портов и соединений:

$$F: \{ i: \{ \text{list} \}, d: \{ \text{list} \}, c: \{ \text{list} \}, m: \{ \text{list} \}, J(f, t): \{ \text{list} \} \}, \quad (1)$$

где  $i$  (*input*) – исходные данные;  $d$  (*destination*) – результаты работы функции;  $c$  (*control*) – управление отношением  $i$  и  $d$ ;  $m$  (*mechanism*) – механизм, реализующий управление  $c$ .

2. Накладываем на базовое определение функциональности (1) граничное условие, реализующее закон семантической связности для элементов определения функций, который гласит, что: а) значение всегда выводимо из аргумента; б) отношение аргументов и значений выводимо из условия (управления):

$$\begin{aligned} \forall F \exists (i, d, c); \\ (d \leftarrow i) \cap ((i, d) \leftarrow c). \end{aligned} \quad (2)$$

3. Выражаем тождество или эквивалентность функций при их декомпозиции и композиции как взаимное соответствие множества функций и множеств их элементов, сгруппированных по ролям:

$$F : \{ F1, F2, \dots \} \equiv F' : \{ \{i\}, \{d\}, \{c\}, \{m\} \}. \quad (3)$$

4. Выражаем специализацию (различимость) портов функций, используя отношение «общее – особенное»:

$$\begin{aligned} I' \subseteq (I1, I2, \dots), D' \subseteq (D1, D2, \dots), \\ C' \subseteq (C1, C2, \dots), M' \subseteq (M1, M2, \dots). \end{aligned} \quad (4)$$

*Примечание.* Специализация может иметь иерархическую вложенность, а глубина детализации функции соответствует числу ступеней вложенности.

5. Выражаем логическое граничное условие для межфункциональных связей (соединений), которое гласит, что «отправителем может быть только порт выходных данных» (Query 1):

$$\begin{aligned} \text{forall } F, J(f, t), \\ Q1 = ((f \in D') \cap (f \neq \emptyset)) \cap (t \in \text{XOR}(I', C', M') \cap (t \neq \emptyset)). \end{aligned} \quad (5)$$

6. Выражаем логическое граничное условие для межфункциональных связей (соединений), которое вводит запрет на наличие портов «ниоткуда» и «в никуда» (Query 2):

$$\begin{aligned} \text{forall } F, J(f, t), \\ Q2 = \neg\{J(f, \emptyset), J(\emptyset, t), J(f \notin D'), \\ J((t \notin I') \cap (t \notin C') \cap (t \notin M'))\}. \end{aligned} \quad (6)$$

7. Вводим оценку факта выполнения всех объявленных условий целостности (состоительности)  $Q$  математической модели (1), которая перемножает значения для всех пяти обозначенных условий и принимает значения true или false:

$$Q = \{(2)*(3)*(4)*(5)\}. \quad (7)$$

8. Выполняем анализ сложности функции  $\Psi$ , складывая количество портов каждого типа:

$$\psi = \{\Sigma(I), \Sigma(D), \Sigma(C), \Sigma(M)\}. \quad (8)$$

Применение математической модели функциональности КФС в программном приложении требует наличия словаря (типологии) элементов функции и отношений между ними, а для её решения требуются вычисления, использующие операции теории множеств и логики предикатов.

### Трансформации функциональных структур

Опираясь на математическую модель в данных шести определениях, мы можем выполнять самую ответственную операцию процессов разработки сложных технических систем – тождественную или эквивалентную трансформацию функциональных структур, на которой основаны детализация требований и выбор технических решений [5]. Декомпозиция и интеграция функций в составе функциональной модели системы выглядят как разделение исходной составной задачи на набор параллельных и последовательных составляющих задач, решение их по-отдельности и объединение результатов решения в единое целое, как показано на рисунках 2, 3 и 4. Параллельная декомпозиция и интеграция функции имеют вид:

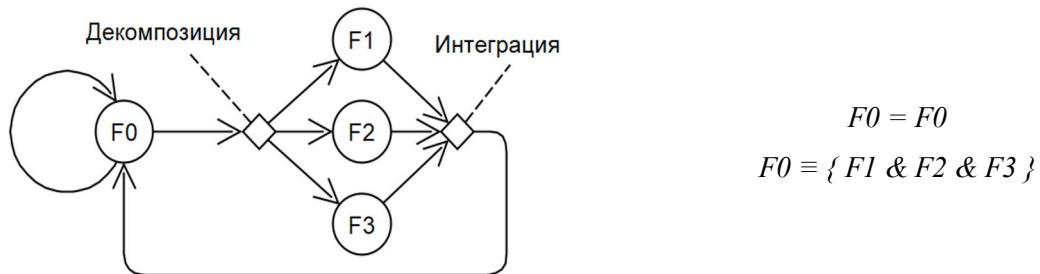


Рисунок 2 – Параллельная декомпозиция и интеграция функции  
Figure 2 – Parallel function decomposition and integration

Последовательная декомпозиция и интеграция представляют собой последовательность исполнения функций, в которой каждое последующее действие использует результат выполнения предыдущего. На рисунке 3 показана функция  $F0$ , тождественная самой себе (как «чёрный ящик») и эквивалентная совместной работе трёх последовательных составляющих функций:

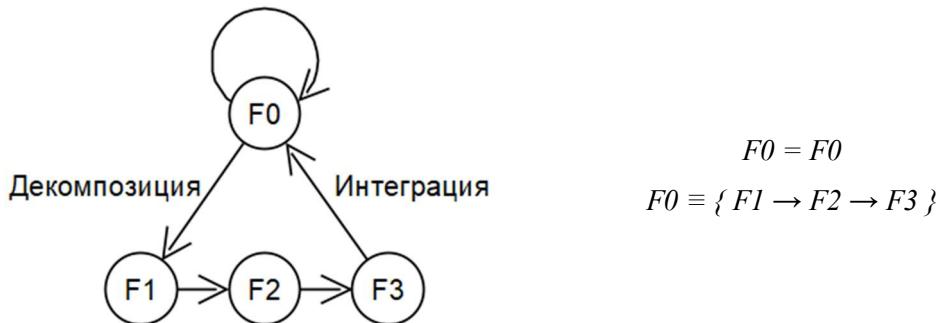


Рисунок 3 – Последовательная декомпозиция и интеграция функции  
Figure 3 – Sequential decomposition and integration of functions

В более общем виде функциональная структура системы может быть представлена в виде ориентированного ациклического графа. Функции  $F1, F2 \dots F7$  на рисунке 4 представлены их номерами:

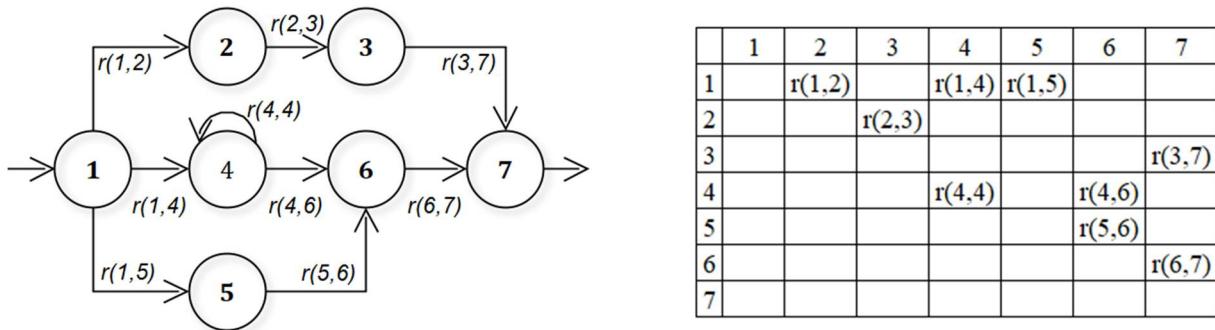


Рисунок 4 – Функциональная структура в виде ориентированного графа и его табличного представления

Figure 4 – Functional structure in the form of a directed graph and its tabular representation

На языке логики предикатов такая функциональная структура представляется как:

$$S : \left\{ \text{nxt}\left( f1, \left( \text{nxt}\left( \text{par}\left( \text{nxt}\left( f2, f3 \right), \text{nxt}\left( f4, f5 \right), f6 \right), f7 \right) \right) \right) \right\}, \quad (9)$$

где *nxt* – «последовательная» связь (next); *par* – «параллельная» связь (parallel).

Такую математическую модель можно использовать для решения задач разработки – выбора и принятия технических решений, проверки совместимости портов функций по типам и единицам измерения, оценки сложности функциональной структуры системы и других.

Эти задачи могут решаться с помощью инструментов вычисления логики первого порядка, с использованием соответствующего математического аппарата [8].

Алгоритм процесса эквивалентной декомпозиции функциональной структуры выглядит, как показано на рисунке 5.

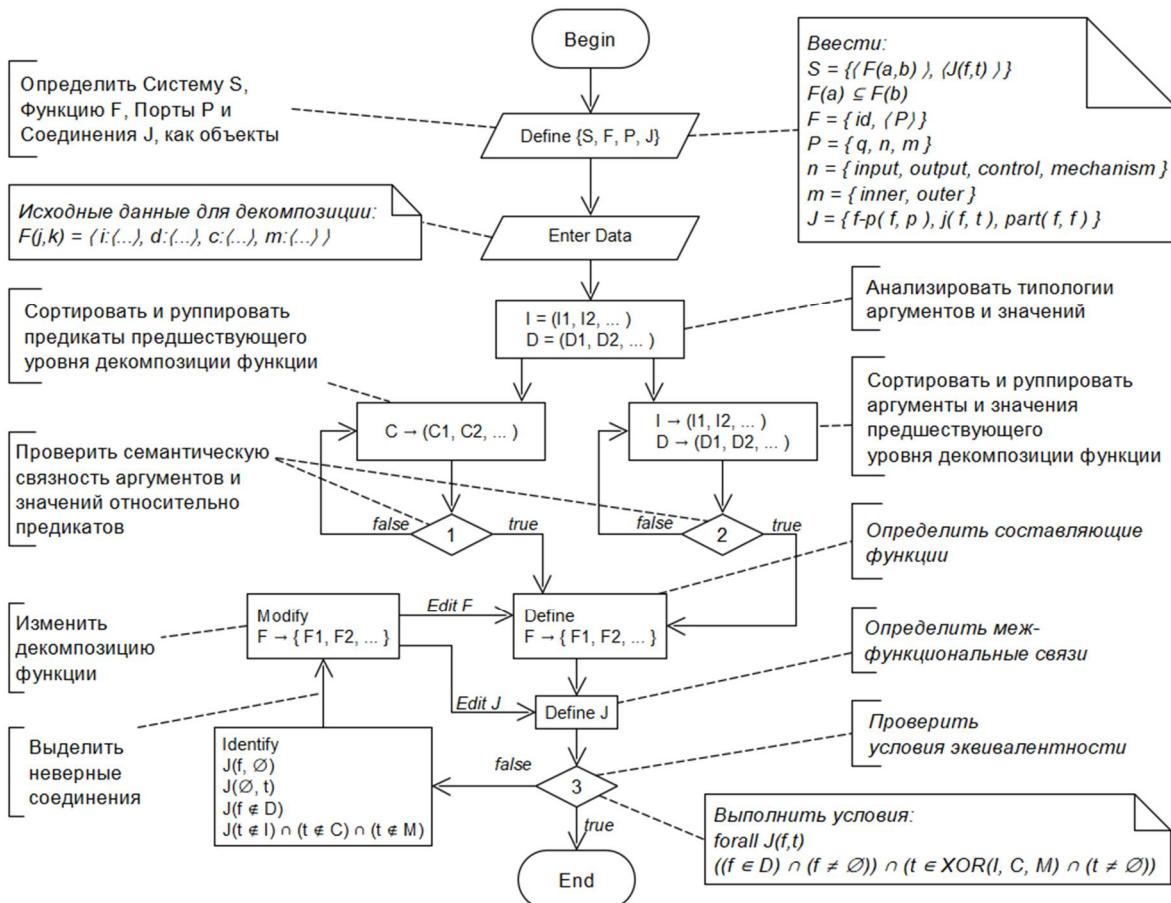


Рисунок 5 – Алгоритм эквивалентной декомпозиции

Figure 5 – Equivalent decomposition algorithm

Результатом вычисления будет тождественная или эквивалентная декомпозиция функциональной структуры, в которой соблюдены условия существования функции, включены принятые технические решения и известны пределы, до которых нужно декомпозировать модель, чтобы определить все механизмы системы. У основного результата также может быть производный полезный результат, потребляемый остальными процессами проектирования [9], например оценка целесообразности решений или прогноз отказоустойчивости и отказобезопасности.

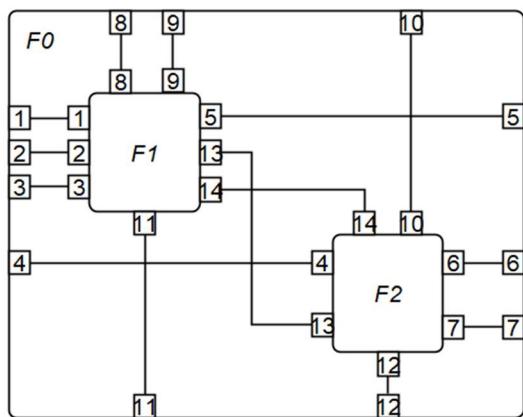
### Пример декомпозиции функциональной структуры системы

Пусть дана исходная функция  $F0$  со следующим набором портов:

$$F0: \{i: \{1, 2, 3, 4\}, d: \{5, 6, 7\}, c: \{8, 9, 10\}, m: \{11, 12\}\}.$$

Число портов каждого из четырёх типов (сигнатура)  $F0: \{4, 3, 3, 2\}$ .

Используя алгоритм (рисунок 5), производим декомпозицию первого уровня (рисунок 6):



$$F1: \{ \{1, 2, 3\}, \{5, 13, 14\}, \{8, 9\}, 11\}$$

$$F2: \{ \{4, 13\}, \{6, 7\}, \{14, 10\}, 12\}$$

Сигнатура:

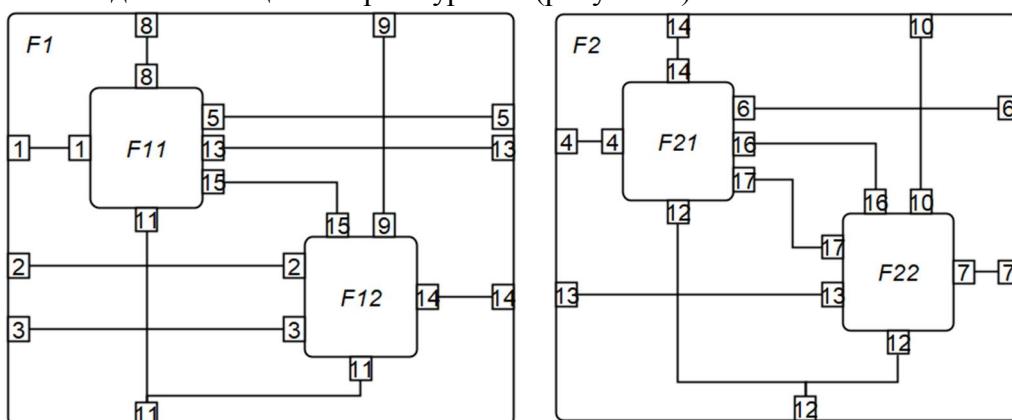
$$F1: \{3, 3, 2, 1\}$$

$$F2: \{2, 2, 2, 1\}$$

Рисунок 6 – Декомпозиция составной функции на составляющие  
Figure 6 – Decomposition of a composite function into its components

Как видно из результата декомпозиции, число функций выросло до двух, число портов увеличилось на 2 и сигнатура свидетельствует о том, что декомпозицию можно продолжить.

Выполняем декомпозицию второго уровня (рисунок 7):



$$F11: \{1, \{5, 13, 15\}, \{8\}, 11\}$$

$$F12: \{\{2, 3\}, 14, \{15, 9\}, 11\}$$

Сигнатура:

$$F11: \{1, 3, 1, 1\}$$

$$F12: \{2, 1, 2, 1\}$$

$$F21: \{4, \{6, 16, 17\}, 14, 12\}$$

$$F22: \{\{17, 13\}, 7, \{16, 10\}, 12\}$$

Сигнатура:

$$F21: \{1, 3, 1, 1\}$$

$$F22: \{2, 1, 2, 1\}$$

Рисунок 7 – Декомпозиция функций  $F1$  и  $F2$  на составляющие

Figure 7 – Decomposition of functions  $F1$  and  $F2$  into components

Как следует из сигнатуры, некоторые функции уже достигли уровня детализации, при котором число отношений между аргументом и значением равно 1.

Процесс декомпозиции продолжается до тех пор, пока для каждого отношения  $i$  и  $d$  не определится реализующий их метод и для каждого метода найдётся техническое решение – исполнительный механизм.

### Заключение

Научная новизна предложенного метода декомпозиции и интеграции функциональных структур состоит в использовании потенциала семантических связей, на которых строится функциональность системы. За счёт их применения накладываются и соблюдаются логические условия на отношения между элементами функций, независимости от их числа и сложности. Сама трансформация функциональных структур получает должный уровень управляемости и обеспечивается обратными связями. На практике разработку систем можно будет вести в обоих направлениях одновременно: от достигнутого результата и от конечной цели с учетом фазово-режимного многообразия и неоднородности (морфологии) систем. Дальнейшим шагом развития системной инженерии после покрытия декомпозиции и интеграции функциональных структур математическими методами может быть переход от функциональных моделей систем к моделям их поведения, который может быть поддержан возможностями искусственного интеллекта.

Общим итогом всех перечисленных инноваций может стать автоматизированный синтез систем с заданными свойствами.

### Библиографический список

1. ISO/IEC 15288-2011 Системная инженерия. Общие положения и терминология.
2. Методология функционального моделирования IDEF0. Москва. Государственный стандарт Российской Федерации, РД IDEF 0 - 2000. 75 с.
3. **Бочаров В.А.** Расширение нотации IDEF0 для функционального описания динамических систем. Вестник РГГУ. 2023. № 2. С. 8-29
4. **Омельченко В.В.** Основы Систематизации. Методология и философские аспекты. Принципы и законы познания реальной действительности / М.: Либроком. 2012.
5. **Бочаров В.А.** Функционально-ролевой метод проектирования систем // В книге: Управление в аэрокосмических системах (УАКС-2022) им. академика Е.А. Мицрина. Материалы 15-ой конференции по проблемам управления.
6. **Авдошин С.М., Набебин А.А.** Дискретная математика. Формально-логические системы и языки // Издательство ДМК. Москва, 2018 г.
7. **Жиляк Н.А.** Логико-комбинаторный подход к выбору оптимальных структур сложных технических систем // Труды БГТУ. 2007.
8. **Зюзков В.М.** Математическая логика и теория алгоритмов // Томский Государственный Университет Систем Управления И Радиоэлектроники (ТУСУР). 2015.
9. Основы Автоматизированного Проектирования : учеб. для вузов / **И.П. Норенков.** - 4-е изд., перераб. и доп. М.: Изд-во МГТУ им. Н.Э. Баумана, 2009. 430, [2] с.: ил. («Информатика в техническом университете»).

UDC 004.724

## MATHEMATICAL MODEL OF COMPLEX TECHNICAL SYSTEMS FUNCTIONAL STRUCTURES TRANSFORMATION

**V. A. Bocharov**, post-graduate student, BMSTU, Moscow, Russia;  
e-mail: bocharov.v@bmstu.ru

*This article is dedicated to the development of systems engineering tools and methods that improve the effectiveness of automated development tools. The most important part of development is the creation of linguistic and mathematical models that capture unused system properties, including the semantics of relation-*

ships between elements of function construction and the constraints imposed on it. The article describes the construction of a mathematical model based on these relationships and its application for identical and equivalent transformations of functional structures.

**Keywords:** system engineering, functionality, mathematical model, technical system, development, automation, transformation

**DOI:** 10.21667/1995-4565-2025-94-98-105

### References

1. ISO/IEC 15288-2011 *System Engineering. Terms and Definitions*.
2. *Metodologija funkciona'l'nogo modelirovaniya IDEF0*. Directive RD IDEF 0 - 2000. 75 p.
3. **Bocharov V.A.** *Rasshirenie notacii IDEF0 dlja funkciona'l'nogo opisanija dinamicheskikh sistem*.
4. **Omelchenko V.V.** *Basics of Systematization. Methodological and Philo-sophical aspects. Principles and Laws of Cognition*. Limbocom publisher 2012. (in Russian).
5. **Bocharov V.A.** *Funkcional'no-rolevoj metod proektirovaniya sistem*. V knige: Upravlenie v aero-kosmicheskikh sistemah (UAKS-2022) im. akademika E.A. Mikrina. Materialy 15 konferencii po problemam upravleniya. (in Russian).
6. **Avdoshin S.M., Nabebin A.A.** *Diskretnaja matematika. Formal'no-logichestkie sistemy i jazyki*. DMK Publishing. Moscow. 2018.
7. **Zhillyak N.A.** Logiko-kombinatornyj podhod k vyboru optimal'nyh struktur slozhnyh tehnicheskikh sistem. *BGTU publishing works*, 2007.
8. **Zuzkov V.M.** Matematicheskaja logika i teorija algoritmov. *Tomsk State University of Control Systems and Radio Electronics*. 2015.
9. **Norenkov I.P.** Osnovy Avtomatizirovannogo Proektirovaniya. Student's book. 4-th Edition. Moscow. *Bauman State Technical University*. 2009.