

Ю.В. Доронина, В.О. РЯБОВАЯ, Д.И. ЧЕСНОКОВ
**ПРИМЕНЕНИЕ МОДЕЛЬНО-ОРИЕНТИРОВАННОГО
ПРОЕКТИРОВАНИЯ ДЛЯ РЕШЕНИЯ ЗАДАЧИ СТРУКТУРНОГО
СИНТЕЗА**

Доронина Ю.В., Рябовая В.О., Чесников Д.И. **Применение модельно-ориентированного проектирования для решения задачи структурного синтеза.**

Аннотация. Чтобы снизить сложность задач структурного синтеза, его делят на этапы, на каждом из которых исследователем при помощи систем поддержки принятия решений производится синтез и анализ моделей систем по заданным входным требованиям и ограничениям. Структурная оптимизация в данном контексте сводится к поиску экстремума некоторой целевой функции, значением которой управляют заданные конструктивные параметры, зависящие от вида задачи.

Для демонстрации работы имитационной модели рассматривается функциональный синтез структуры информационной системы управления предприятием, при этом функциональными элементами выступают автоматизируемые бизнес-процессы, а структурными — средства автоматизации. Разработан тестовый пример, в котором в качестве функциональных элементов описаны типовые процессы бюджета, маркетинга, закупок и продаж, производства, кадров.

Использование разработанной модели функционального синтеза продемонстрировано на примере задачи выбора программных продуктов при проектировании корпоративных информационных систем. На основании ряда экспериментов определено множество вариантов решения с наибольшим значением функции приспособленности. Установлено, что на значение функции влияет количество избыточных функций, которые содержат выбранные структурные элементы.

Ключевые слова: модельно-ориентированное проектирование, структурный синтез, информационная система, генетический алгоритм, направленная мутация, экстремум, эвристический подход.

1. Введение. Современное развитие информационных технологий дает большой выбор способов реализации проектируемых информационных систем [1]. Выбор в пользу той или иной технологии происходит исходя из требований и ограничений со стороны пользователей, причем требования имеют свойство изменяться в процессе разработки и проектирования. Применительно к теории построения систем поддержки принятия решений процесс проектирования информационной системы — это процесс принятия исследовательских, конструкторских и проектных решений, при помощи которых на выходе можно получить такой проект системы, который удовлетворяет всем требованиям заказчиков [2].

В соответствии с масштабами проекта, группой проектировщиков выбирается одна из технологий проектирования. Технология проектирования — это совокупность методологии и средств

проектирования ИС, а также методов и средств организации проектирования [3].

Проведенные ранее исследования показали [4], что задачи управления структурной динамикой информационной системы по своему содержанию относятся к классу задач структурно-функционального синтеза. К настоящему времени благодаря системной отрасли научных знаний, одной из компонент которой является квалиметрия моделей, представляющая из себя прикладную науку, ориентированную на разработку методологии и технологий оценивания качества моделей и комплексов [5], создан богатый методологический и методический аппарат, позволяющий успешно преодолевать указанные трудности. Однако при возникновении новых требований (или их корректировке) к работе системы (быстродействие, надежность, изменение частоты измерений) возможны проблемы структурного характера и качества оценки ее моделей (вариантов).

Для того чтобы снизить сложность задач структурного синтеза, его зачастую делят на этапы [6], на каждом из которых исследователем при помощи систем поддержки принятия решений производится синтез и анализ моделей систем по заданным входным требованиям и ограничениям. Структурная оптимизация в данном контексте сводится к поиску экстремума некоторой целевой функции, значением которой управляют заданные конструктивные параметры, которые зависят от вида задачи и соответствующей системы ограничений.

Модельно-ориентированный подход — это такой стиль разработки и проектирования информационных систем, при котором модели являются основными объектами разработки, на основании которых происходит генерация кода и других объектов [7]. Метамоделю представляет описание понятий, которые используются в модели, а также определяет метаданные, которые обрабатываются различными инструментами моделирования. Метамоделю являются актуальными в силу того, что необходимо стандартизировать средства описания на некотором абстрактном уровне для решения проблемы конфликта и взаимодействия проектировщиков на различных уровнях и этапах проектирования [8].

Модельно-ориентированный подход (МОП) обладает рядом особенностей [9], которые позволяют решить основные проблемы структурного синтеза:

— МОП позволяет абстрагироваться от стандартов, принятых на различных уровнях и этапах проектирования и синтеза, таким образом решая проблему взаимодействия проектировщиков;

— МОП предоставляет широкие возможности для участия проектировщика в процессе синтеза, визуализируя процессы,

происходящие в системе, и позволяя влиять на них в момент выполнения модели;

— МОП рассматривает синтезируемые системы с учетом только тех параметров, которые заданы на определенном этапе синтеза, то есть исследователь решает, какие параметры должны влиять на синтез;

— МОП позволяет повторно использовать модели для создания производных систем с улучшенными характеристиками.

2. Направление исследований. В статье предлагается применить модельно-ориентированное проектирование для решения задач структурного синтеза информационной системы (ИС) на основе генетического алгоритма на примере имитационной модели системы, разработанной в среде AnyLogic. Продемонстрировать работу модели с помощью функционального синтеза структуры информационной системы управлением предприятием.

3. Анализ задачи функционального синтеза. Задача проектирования ИС промышленных предприятий достаточно сложна, так как характер обрабатываемой такой системой информации крайне разнороден и сложно формализуем [10]. Крупные предприятия зачастую имеют множество подразделений и филиалов, а значит — распределенную структуру хранения информации и в общем случае целый ряд программных продуктов, которые обеспечивают его деятельность. Связи между видами деятельности зачастую имеют типовой характер, поэтому основной проблемой становится автоматизация учета всех бизнес-процессов, происходящих на предприятии.

Следовательно, существует необходимость выбора компонентов ИС таким образом, чтобы в совокупности они удовлетворяли всем потребностям, а значит, автоматизировали все бизнес-процессы, которые требуются по замыслу проектировщика. При этом функциональными элементами будут выступать автоматизируемые бизнес-процессы, а структурными — средства автоматизации.

Одним из способов формализации задачи функционального синтеза структур и их дальнейшей оптимизации можно определить нахождение значения некой целевой функции F и приведение ее к экстремуму [11]. Поскольку оптимизационные задачи имеют истоки, в том числе из экономики [12], то иногда целевую функцию называют функцией дохода, а балансовые уравнения — ресурсными ограничениями. В силу того, что любые практические задачи решаются в условиях ограниченности ресурсов, становится понятным, что они относятся к задачам нахождения условного экстремума.

Пусть $f_i \in \{F\}$ — некоторый функциональный элемент, который входит в исходное множество всех возможных функций $\{F\}$. Тогда

$g_i \in \{G\}$ — некоторый структурный элемент, входящий в исходное множество всех возможных элементов $\{G\}$. Структурный элемент принадлежит системе и соотносится с ней как часть целого. Необходимо найти такой вариант системы x (x — вектор), что

$$F(x, \tau) = \max_{V_j \in V} F(V_j, \tau_i), \quad (1)$$

где V_j — множество вариантов синтезируемой системы, а τ_i — вектор требований к синтезируемой системе.

При этом сам вариант x выбран из множества возможных решений S :

$$S = \bigcup_{\forall k} V_k(F_k^\Pi) \Big|_{k=1}^P \rightarrow \bigcup_i (\tau_i), \quad (2)$$

где P — количество возможных вариантов структуры, а F^Π — функция пересечения структурно-функциональных элементов системы.

$$F^\Pi = \{G\} \cap \{F\}. \quad (3)$$

Главная особенность решения представленной задачи (проблема структурно-функционального синтеза) состоит в следующем: определение оптимальных программ управления основными элементами и подсистемами ИС может быть выполнено лишь после того, как будет известен перечень функций и алгоритмов обработки информации и управления, который должен быть реализован в указанных элементах и подсистемах. В свою очередь, распределение функций и алгоритмов по элементам и подсистемам ИС зависит от структуры и параметров законов управления данными элементами и подсистемами.

Погрешность полученного решения представленной задачи оптимизации (из-за использования эвристических подходов [13]) определяется разностью между оптимальным значением x проектного параметра и приближением к нему x_* .

Требуется, чтобы эта погрешность по модулю была меньше заданного допустимого значения ε :

$$|x - x_*| < \varepsilon. \quad (4)$$

Процесс решения задачи оптимизации состоит в последовательном сужении интервала изменения проектного параметра,

называемого *интервалом неопределенности*, то есть оптимальное значение проектного параметра должно находиться в интервале неопределенности — отрезке $[x_n, x_{n+1}]$, причем $x_{n+1} - x_n < \varepsilon$.

Тогда для выполнения условия $|x - x_*| < \varepsilon$ в качестве приближения к оптимальному значению можно принять любое $x_* \in [x_n, x_{n+1}]$. Например, $x_* = x_n$ или $x_* = x_{n+1}$, или $x_* = (x_n + x_{n+1})/2$. В последнем случае достаточно выполнения неравенства $x_{n+1} - x_n < 2\varepsilon$.

Один из вариантов — это прямое соответствие (рисунок 1а). В данном случае каждый описанный блок (подсистема) выполняет четко определенную функцию. Еще одним вариантом будет случай, когда каждому блоку G_i соответствует несколько функций F_j (рисунок 1б). Последним возможным вариантом будет случай, когда один функциональный компонент F реализуется несколькими структурными блоками G (рисунок 1в):

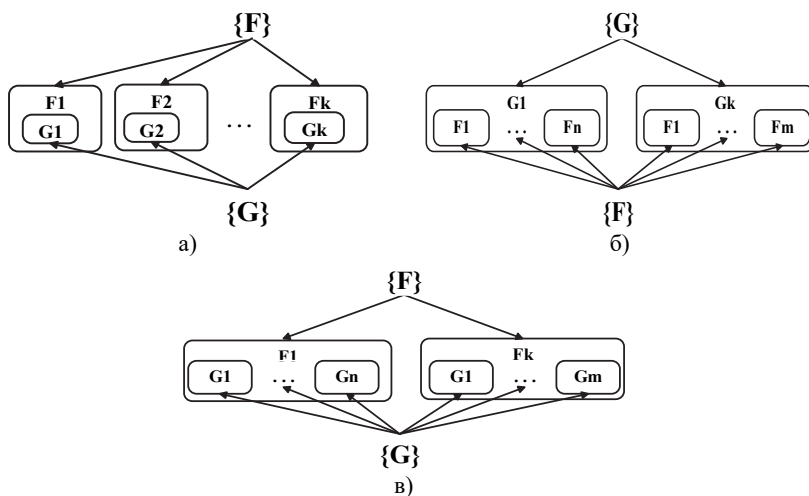


Рис. 1. Схемы соответствий: а) прямое соответствие блоков подсистем и выполняемых ими функций «один к одному»; б) соответствие структурных и функциональных элементов «один ко многим»; в) соответствие структурных и функциональных элементов «многие к одному»

В рамках разрабатываемой системы подлежат реализации первый и второй варианты (рисунок 1а и 1б), ситуация соответствия структурных и функциональных элементов «многие ко многим» не рассматривается.

Дальнейшее обобщение рассматриваемой задачи синтеза ИС проводится в направлении учета факторов неопределенности, вызванных воздействием внешней среды, а также в направлении, связанном с постановкой и решением многокритериальных задач выбора оптимальных структур [5]. Одной из основных трудностей решения сформулированной задачи синтеза структур ИС является ее большая размерность, которая определяется общим числом переменных и ограничений, используемых при формализации и решении данной задачи.

Поэтому в основу решения представленной задачи положены различные варианты реализации процедур декомпозиции (композиции), агрегирования (деагрегирования), координации, с помощью которых удается преодолеть «проклятие размерности» [11].

Дополнительные особенности данные процедуры приобретают еще и из-за того, что большинство переменных, используемых при решении задачи синтеза структур ИС, должны принимать целочисленные значения.

В качестве успешного примера решения задач данного класса можно привести результаты, полученные в работах [14-17]. Авторами данных работ был предложен агрегативно-декомпозиционный подход, предусматривающий (на основе альтернативно-графовой формализации) построение многоуровневого комплекса взаимосвязанных моделей различного типа: оптимизационных аналитических и/или имитационных.

При практической реализации рассматриваемого подхода широко использовались как универсальные, так и специализированные средства автоматизации моделирования [14-17]. Особенности формального описания рассматриваемого класса задач синтеза структур ИС позволяют при разной интерпретации переменных и ограничений, используемых при его описании, формулировать и решать не только задачи синтеза технической структуры ИС, но и задачи синтеза других типов структур (например, топологической, организационной и т. п.).

Многочисленные исследования задач синтеза структур ИС показали [14-17], что если при формировании облика ИС для некоторых ее элементов и подсистем возникают проблемы ликвидации больших (пиковых) информационных нагрузок, то в этом случае должны уже оптимизироваться сами правила, алгоритмы функционирования указанных элементов и подсистем.

4. Реализация модели. Для реализации модели выбрана среда имитационного моделирования AnyLogic [18]. Уникальность среды AnyLogic состоит в способности эффективно решать задачи моделирования любого масштаба и уровня абстракции, в том числе для

разнородных систем в их взаимосвязи. AnyLogic применяется в диапазоне от микромоделей «операционного» уровня, где важны конкретные размеры, расстояния, скорости, времена, до макромоделей «стратегического» уровня, на котором рассматривается глобальная динамика обратных связей, тенденции на длительных временных отрезках, и оцениваются стратегические решения. Кроме того, эта среда относится к отечественным разработкам.

На основании системотехнического анализа разрабатываемой системы реализация модели в среде AnyLogic делится на две составляющие:

- имитационная модель, отображающая основные процессы, происходящие в системе, отображающая всю необходимую информацию и статистические данные;
- библиотека структурного синтеза — модуль, содержащий в себе математическое описание модели и непосредственную реализацию операторов генетического алгоритма.

Имитационная часть модели реализована с помощью средств графической среды имитационного моделирования AnyLogic. Для моделирования логики функционирования модели ИС используется дискретно-событийная библиотека, встроенная в AnyLogic. Основной идеей является отображение процессов системы, посредством обслуживания заявок, а также контроль различных количественных показателей, связанных с обработкой данных в оптимизационном алгоритме. Визуализация дискретно-событийной модели ИС представлена на рисунке 2.

Разработанная логика описывает модель ИС в среде имитационного моделирования AnyLogic, реализующая этап структурно-функционального синтеза системы с использованием генетического алгоритма [19] и визуализирующая его от момента возникновения необходимости в синтезе некоторой функциональной структуры до момента нахождения такой структуры системы, которая удовлетворяла бы всем требованиям и ограничениям.

В качестве требований к системе на данном этапе можно задать лишь требования, исходя из которых осуществляется расчет функции приспособленности синтезируемой системы [20].

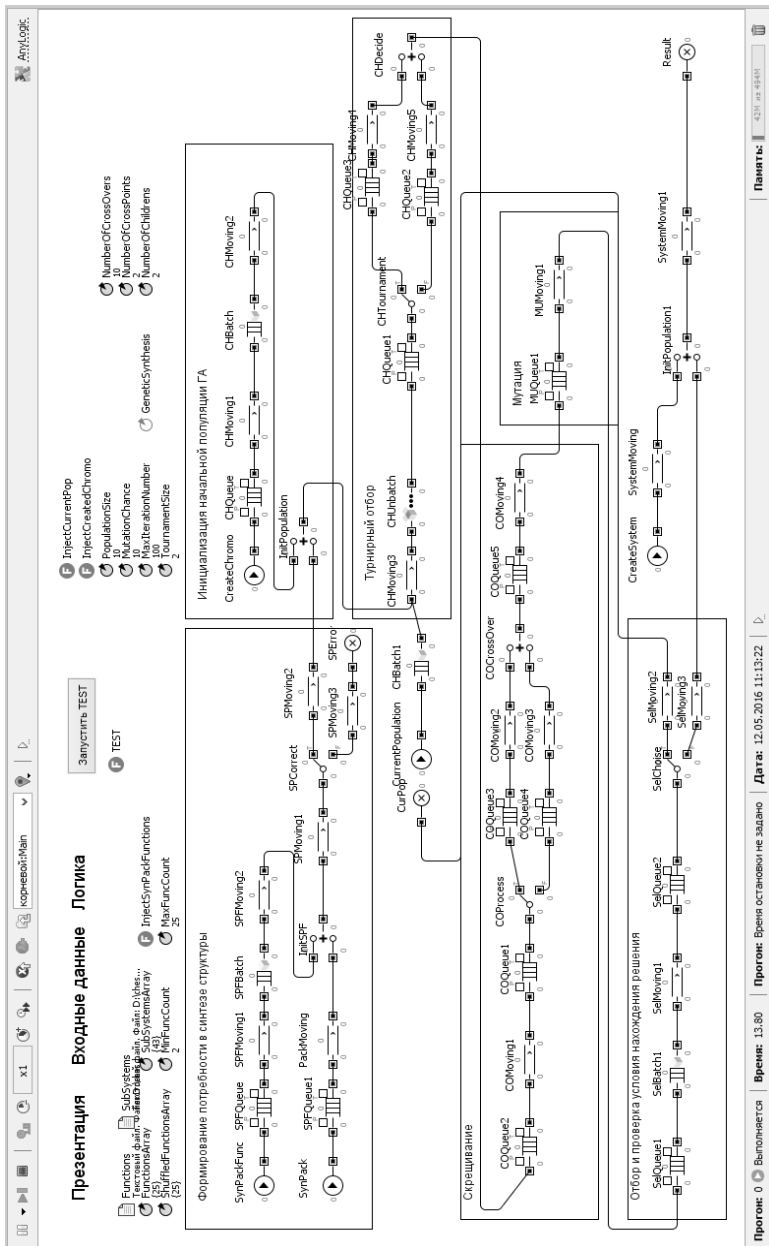


Рис.2. Дискретно-событийная модель

Разработанная модель позволяет визуализировать процессы, происходящие в системе с помощью графической библиотеки средствами AnyLogic. Вид графического отображения модели представлен на рисунке 3.

Для отслеживания текущих результатов работы модели выводится информация о текущем составе множества функциональных требований к синтезируемой системе, значения функции приспособленности для сгенерированных на текущей итерации структур, элементный состав решения, имеющего наиболее высокий показатель значения целевой функции на данный момент. Также для данного решения определяется, какие функциональные требования оно уже способно выполнять и какие существуют избыточные элементы у синтезируемой системы.

Для исследования характеристик модели ИС и их влияния на результат добавляется закладка с графиками зависимости и значениями для таких параметров, как среднее количество итераций оптимизационного алгоритма, коэффициент адекватности требований, среднее значение целевой функции. Таким образом, реализована имитационная модель ИС в среде AnyLogic, осуществляющая структурный синтез с использованием генетического алгоритма в качестве оптимизационного. План экспериментов представлен на рисунке 4.

В представленной имитационной модели исследуются следующие характеристики системы: среднее число итераций оптимизационного алгоритма до получения «хорошего» решения ($KolMean$); коэффициент адекватности требований, который исследуется для режима случайной генераций и равен отношению успешно прошедших проверку требований к общему числу сгенерированных векторов требований ($Adekv$); среднее значение целевой функции ($FMean$); количество нереализуемых требований ($FUnreal$).

Изменяемые исследователем параметры системы: режим автогенерации требований; изменение количества точек кроссовера ($CrossPoints$) генетического алгоритма; изменение вероятности мутации ($MutationChance$); изменение минимального порога числа синтезируемых функций ($MinFuncCount$) для режима автогенерации требований; изменение порогового значения целевой функции ($Threshold$) для выполнения условия останова; изменение числа популяции ($PopulationSize$).

Реальные объекты обычно обладают большой сложностью, из-за чего выполнение всех практических опытов невозможно [21].

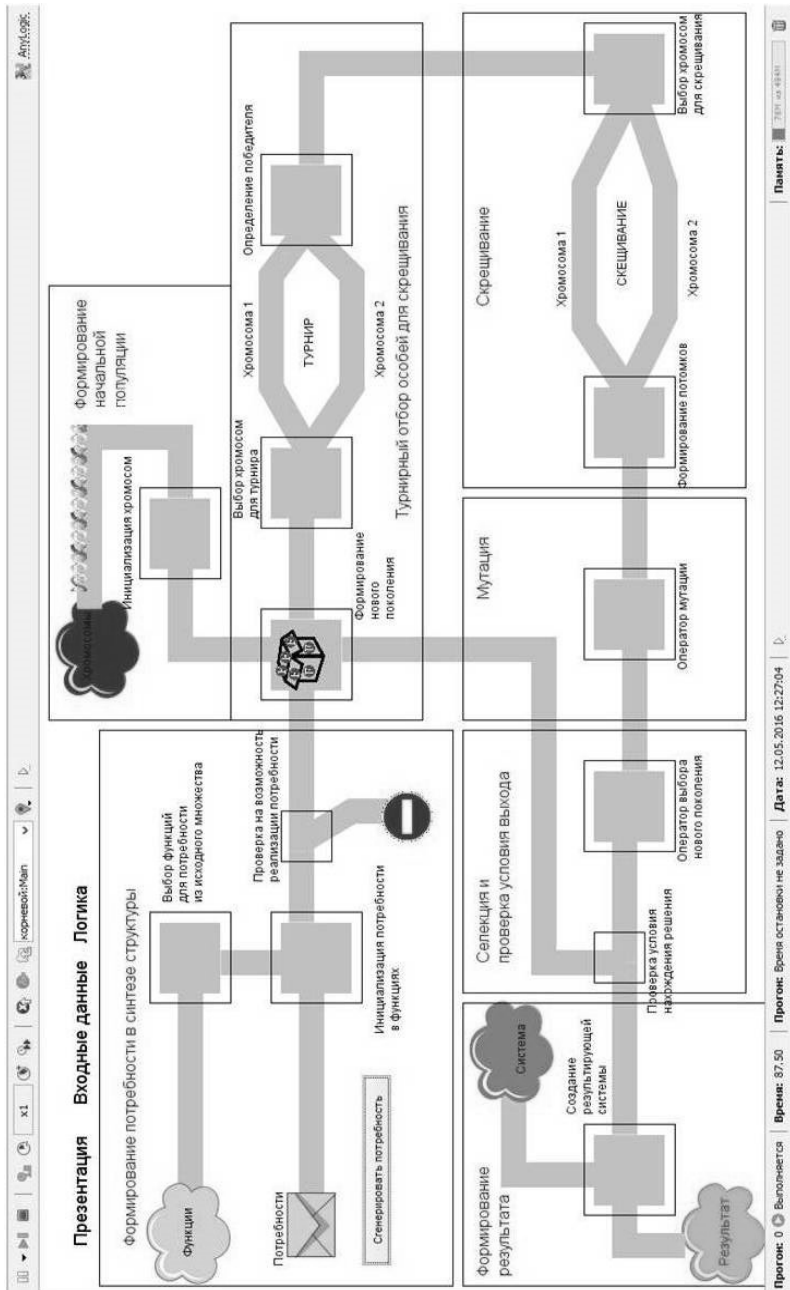


Рис.3. Графическое отображение модели



Рис. 4. План проведения экспериментов

В связи с этим было проведено несколько опытов, четыре из которых представлены в данной работе. Число возможных опытов определяется выражением $N = p^k$, где N — число опытов, p — число уровней, k — число факторов. Опыты проводились на тестовом множестве функциональных $F = \{f1, f2, \dots, f25\}$ и структурных $G = \{g1, g2, \dots, g43\}$ элементов. В таблице 1 приведено описание параметров модели.

Таблица 1. Значения параметров модели по умолчанию

Имя параметра – Назначение	Значение по умолчанию
MinFuncCount — минимальное количество функций, которые должна выполнять синтезируемая система	2
MaxFuncCount — максимальное допустимое количество функций, которые должна выполнять синтезируемая система	999
MaxIterationNumber — максимальное число итераций	100
PopulationSize — размер популяции ГА	100
MutationChance — шанс срабатывания оператора мутации	3
NumberOfCrossOvers — число скрещиваний на очередной итерации	100
NumberOfCrossPoints — число точек кроссинговера	2
NumberOfChildrens — число потомков от одного скрещивания	2

Опыт №1 исследования параметров модели. Пусть имеются следующие значения параметров модели: CrossPoints = 1; MutationChance = 1; PopulationSize = 10; MinFuncCount = 10. Производилось изменение значения параметра Threshold фиксируя KolMean. Результаты представлены в таблице 2.

Таблица 2. Исследование параметров модели

Threshold	PopulationsSize	KolMean
1.9	10	29.9
2.0	10	32.6
2.1	10	33.3
2.2	10	42.3
2.3	10	54.3
2.4	10	72.35
1.9	20	4.4
2.0	20	16.3
2.1	20	22.6
2.2	20	32
2.3	20	41.9
2.4	20	50.3

На рисунке 5 приведен полученный в AnyLogic график зависимости числа итераций алгоритма (ось y), необходимого для получения решения, от порогового значения целевой функции (ось x), которое является достаточным условием останова.

Из анализа значений таблицы 1 и рисунка 5 следует, что начиная со значения 2.1, идет резкое увеличение среднего числа итераций, это, в свою очередь, говорит о том, что пространство возможных решений, со значением целевой функции больше заданного, резко сокращается и оптимизационному алгоритму требуется больше итераций, а значит, и больше времени на нахождение приемлемого варианта решения.

Опыт №2 исследования параметров модели. Пусть имеются следующие значения параметров: CrossPoints = 1; Threshold = 2.4; PopulationSize = 10; MinFuncCount = 10. Производилось изменение значения параметра MutationChance при фиксированном KolMean. Результаты представлены в таблице 3.

Таблица 3. Исследование параметров модели

MutationChance	PopulationsSize	KolMean
1	10	192.8
2	10	143.5
3	10	110.62
4	10	85.4
5	10	70.2
6	10	63.9

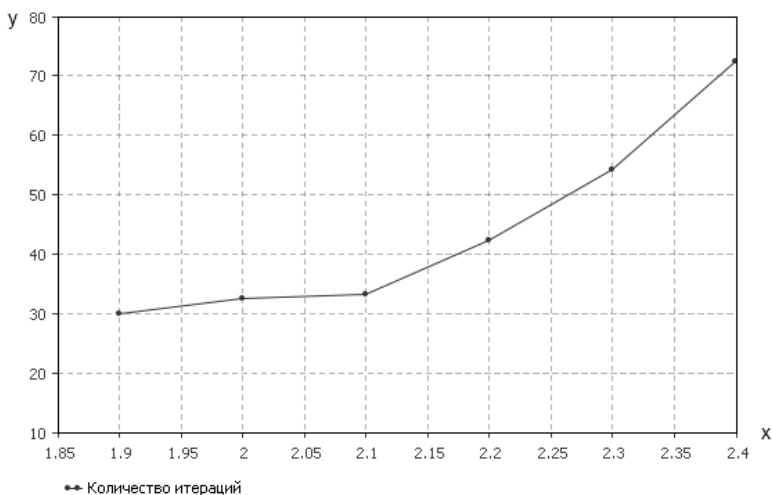


Рис. 5. График зависимости количества итераций алгоритма от порогового значения целевой функции

На рисунке 6 приведен полученный в AnyLogic график зависимости среднего числа итераций (ось y) от вероятности мутации генетического алгоритма (ось x). По графику и результатам таблицы 3 видно, что среднее число итераций генетического алгоритма, которое необходимо для получения приемлемого решения, имеет обратную зависимость относительно вероятности мутации при скрещивании.

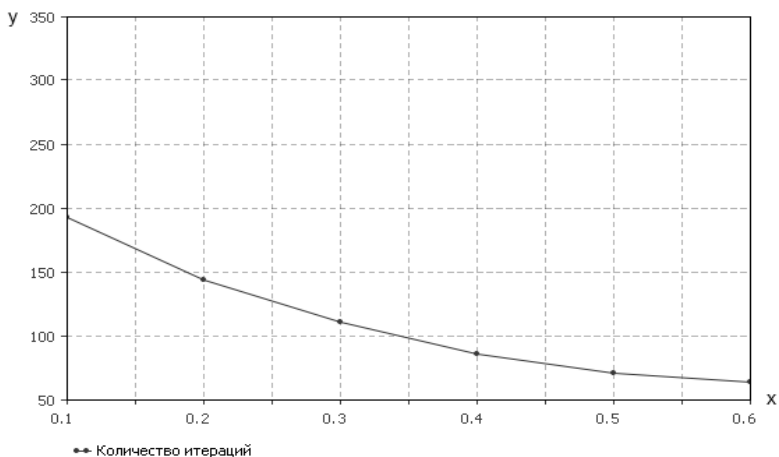


Рис. 6. График зависимости количества итераций алгоритма от вероятности мутации генетического алгоритма

На рисунке 7 (*Опыт №3*) показано, как изменяется среднее значение количества итераций генетического алгоритма (ось y) от количества запусков алгоритма (ось x) при следующих параметрах: CrossPoints = 2; Threshold = 2.4; PopulationSize = 10; MinFuncCount = 10; MutationChance = 1.

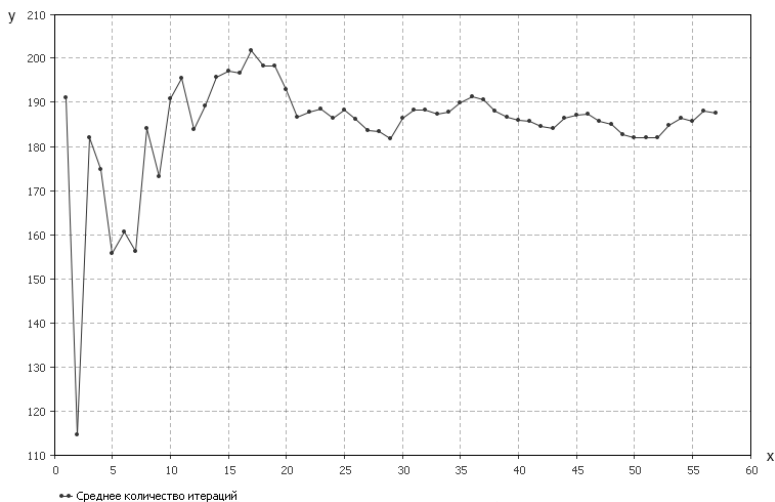


Рис. 7. Зависимость среднего числа итераций генетического алгоритма от количества запусков

На рисунке 7 видно, что график приближается к некоторой величине, и с каждым новым запуском мера разброса уменьшается.

Далее (*Опыт №4*), при тех же условиях: Threshold = 2.4; PopulationSize = 10; MinFuncCount = 10; MutationChance = 1, после 50 запуска меняется значение CrossPoints = 2; на рисунке 8 наблюдается увеличение среднего числа итераций после 50 итерации, что связано с зависимостью данного параметра от количества точек кроссовера генетического алгоритма. Можно сделать вывод о том, что эффективность двухточечного кроссовера выше, чем одноточечного.

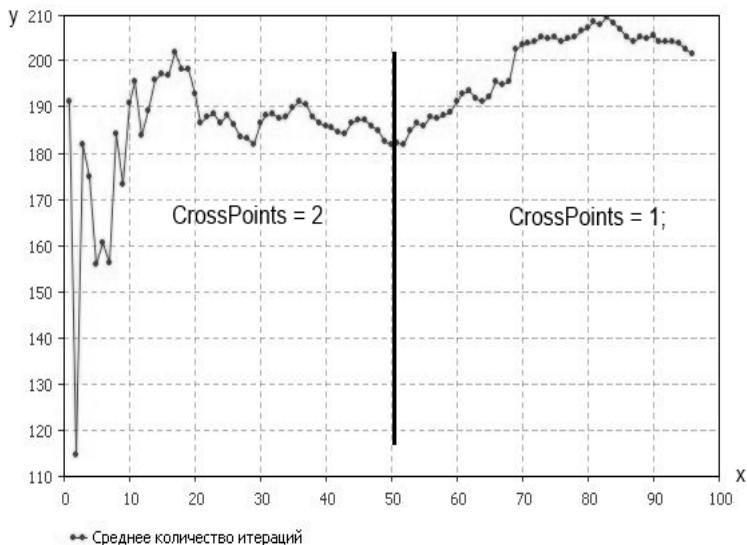


Рис. 8. Зависимость среднего числа итераций от количества точек скрещивания

При исследовании зависимости коэффициента адекватности требованиям в режиме их автоматической генерации, прежде всего, необходимо изменить параметры количества нереализуемых требований $FUnreal$ во входных данных и минимального количества синтезируемых функций $MinFuncCount$. Остальные параметры будут фиксированными: $CrossPoints = 2$; $Threshold = 2.4$; $PopulationSize = 10$; $MinFuncCount = 10$; $MutationChance = 1$.

В результате эксперимента получены значения коэффициента адекватности для $FUnreal = 2$, $MinFuncCount = 4$ (таблица 4).

Таблица 4. Значения коэффициента адекватности требований

FUnreal	MinFuncCount	Adekv
1	1	0.875
1	2	0.644
1	3	0.522
1	4	0.470
1	5	0.398
2	1	0.565
2	2	0.384
2	3	0.267
2	4	0.188

Как видно из таблицы 4, даже несколько нереализуемых требований во входных данных приводят к крайне низким значениям коэффициентов адекватности.

Таким образом, из проведенных экспериментов можно сделать следующие выводы:

- количество итераций генетического алгоритма, а значит, и скорость реализации алгоритма по синтезу структуры системы непосредственно соотносится с пороговым значением целевой функции;

- количество итераций генетического алгоритма имеет обратную зависимость относительно вероятности мутации при скрещивании;

- исходя из увеличения среднего числа итераций генетического алгоритма, при смене оператора двухточечного кроссовера на одноточечный эффективность двухточечного оператора выше;

- в режиме автоматической генерации требований даже несколько нереализуемых требований во входных данных приводят к крайне низким значениям коэффициентов адекватности.

5. Практическая реализация модели. Для демонстрации работы модели рассматривается функциональный синтез структуры (компонентной архитектуры) информационной системы (ИС) управления предприятием.

Разработан тестовый пример, где в качестве функциональных элементов (таблица 5) описаны типовые процессы бюджета, маркетинга, закупок и продаж, производства, кадров и т.д.

В качестве структурных элементов (таблица 6) используются реально существующие программные продукты, автоматизирующие то или иное направление деятельности предприятия.

Таблица 5. Сокращенный список функциональных элементов

Код функционального элемента	Наименование
f01	Бюджетирование и планирование. Составление плана закупок
f12	CRM и маркетинг. Анализ цен конкурентов
f17	Продажи. Учет реализации товаров и услуг
f21	Закупки. Учет поступления товаров и услуг
f27	Производство. Учет выпущенной продукции
f32	Кадры. Составление графиков работы
f39	Общепит. Учет выпускаемых блюд

Таблица 6. Сокращенный список структурных элементов

Код	Наименование	Список функций
s1	Центр финансовой отчетности	f2, f8, f9, f6, f7, f46, f19, f24, f30, f35, f46, f05
s6	CoffeeCupPlanning	f2, f5, f36, f37
s7	Market manager	f10, f11, f12, f13, f46
s9	EN-planning	f2, f4, f8, f13, f31, f32, f33, f34, f35
s10	Restaurant 2	f14, f47, f48, f49, f15, f18, f24
s11	BC:Бухгалтерия	f16, f17, f18, f19, f20, f21, f22, f23, f24, f31, f32, f33, f34, f35

После того как были определены множества структурных и функциональных элементов, необходимо определить вектор функциональных требований. Пусть для проектируемой ИС необходимо выбрать такой набор программных продуктов, который бы выполнял автоматизацию процессов управления предприятием для его эффективной работы (таблица 7).

Таблица 7. Функциональные требования к проектируемой ИС

Код	Наименование
f01	Составление плана закупок
f02	Составление бюджета предприятия
f03	Составление плана продаж
f08	Составление план-фактного анализа бюджетирования
f09	Возможность работы по разным сценариям
f10	Учет сделок с клиентами
f16	Учет заказов клиента
f22	Учет поступления товаров и услуг
f24	Отчет о закупках
f36	Учет рабочего времени
f37	Учет производственных задач

В таблице 8 приведены результаты моделирования при следующих исходных параметрах модели: пороговое значение целевой функции — 2; размер популяции — 20; вероятность мутации — 3; число скрещиваний на 1 витке — 10; число потомков от одного скрещивания — 4; количество точек скрещивания — 2.

Таблица 8. Результаты работы модели

Номер	Результат	Значение целевой функции
1	s6,s13,s19	2.06
2	s9,s11,s17,s23	2.04
3	s2,s8,s11,s19	2.07
4	s5,s20,s21	2.16
5	s5,s6,s17	2.12
6	s5,s11,s12	2.22
7	s5,s6,s21	2.26
8	s11,s13,s21	2.10
9	s6,s12,s13	2.12
10	s13,s20,s21	2.08

В результате работы модели было получено 10 вариантов системы. Значение целевой функции (нахождения условного экстремума) варьируется в небольшом диапазоне. Это связано, прежде всего, с тем, что входное множество структурных элементов в основном содержит элементы с большим количеством выполняемых функций.

Поэтому для выбранных 11 функциональных требований варианты решений могут содержать немало избыточных функций.

Полученные результаты сортируются по значению функции приспособленности и выбираются 3 лучших:

1) Первый вариант системы со значением целевой функции 2.26 содержит следующие программные компоненты:

- s5. Мастер финансов;
- s6. CoffeeCupPlanning;
- s21. BC:Бухгалтерия.

2) Второй вариант системы со значением целевой функции 2.22 содержит следующие программные компоненты:

- s5. Мастер финансов;
- s6. Athena;
- s21. OpenBravo.

3) Третий вариант системы со значением целевой функции 2.16 содержит следующие программные компоненты:

- s5. Мастер финансов;
- s20. Управление производством;
- s21. BC:Бухгалтерия.

Найденные решения интерпретируются следующим образом:

1. Решения с меньшим значением целевой функции состоят из таких структурных элементов, которые выполняют заданные требования, но при этом являются в большей мере избыточными, то есть выполняют также и «лишние» функции.

2. Решения же с большим значением целевой функции состоят из таких структурных элементов, которые выполняют заданные требования и при этом практически не содержат в себе «лишних» функциональных элементов.

Таким образом, в результате работы модели проектировщик получает различные варианты выбора программных продуктов для построения информационной системы предприятия. Поэтому можно говорить о модели как о системе поддержки принятия решений с решающей ролью исследователя.

Применительно к показанному примеру дальнейший выбор может осуществляться исходя из стратегии развития компании и ее информационной системы.

6. Заключение. В процессе применения модельно-ориентированного проектирования для решения задач структурного синтеза на основе генетического алгоритма разработана имитационная модель структурного синтеза, а также произведен анализ и настройка ее параметров. Определено влияние значений управляющих параметров на

характеристики разработанной модели. Исследованы такие характеристики модели, как: адекватность требований, среднее число итераций до получения решения, среднее значение целевой функции.

Установлены следующие зависимости:

– количество итераций генетического алгоритма, а значит, и скорость синтеза структуры системы имеет прямую зависимость относительно порогового значению целевой функции;

– количество итераций генетического алгоритма имеет обратную зависимость относительно вероятности мутации при скрещивании;

– исходя из увеличения среднего числа итераций генетического алгоритма, при смене оператора двухточечного кроссовера на одноточечный можно сделать вывод о большей эффективности двухточечного оператора;

– в режиме автоматической генерации требований даже несколько нереализуемых требований во входных данных приводят к крайне низким значениям коэффициентов адекватности.

Было продемонстрировано использование разработанной модели функционального синтеза на примере задачи выбора программных продуктов при проектировании корпоративных информационных систем. Исходное множество содержало 23 структурных и 49 функциональных элементов. По определенным функциональным требованиям сделано несколько запусков модели. Определено множество вариантов решения с наибольшим значением функции приспособленности. Установлено, что на значение функции влияет количество избыточных функций, которые содержат выбранные структурные элементы. Таким образом, модельно-ориентированный подход является эффективным при реализации задач структурного и структурно-функционального синтеза систем.

Литература

1. *Imboden D., Pfenninger S.* Introduction to Systems Analysis: Mathematically Modeling Natural Systems // Berlin, New York, Springer. 2013. № 8. pp. 235–252.
2. *Туголуков Е.Н., Ткачев А.Г., Рухов А.В. и др.* Проектирование сложных систем // Тамбов: Изд-во: Тамб. гос. тех. ун-т, 2008. 32 с.
3. *Коцюба И.Ю.* Основы проектирования информационных систем // Санкт-Петербург: Университет ИТМО, 2015. 206 с.
4. *Охтлев М. Ю., Соколов Б.В., Юсупов Р.М.* Интеллектуальные технологии мониторинга состояния и управления структурной динамикой сложных технических объектов // Москва: Наука, 2005. 291 с.
5. *Потрясаев С.А., Соколов Б.В., Юсупов Р.М.* Содержательное и формальное описание проблемы структурно-функционального синтеза и управления развитием информационной системы наземно-космического мониторинга // Труды СПИИРАН. 2013. Вып. 28. С. 82–106
6. Задачи структурного синтеза и принятия решений // URL: http://www.life-prog.ru/2_70003_zadachi-strukturnogo-sinteza-i-prinyatiya-resheniy.html (дата обращения: 24.10.2016)

7. *Зеленский В.А.* Проектирование сложных систем // Самара: Самар. гос. аэрокосм. ун-т им. С.П.Королева, 2012. 96 с.
8. *Баранов И.А.* Модельно-ориентированный подход в разработке программных компонент для комплексов СМ1820М // Москва: ОАО «ИНЭУМ им. И.С. Брука», 2012. 11 с.
9. *Бродский Ю.И.* Проблема описания и синтеза распределенных имитационных моделей сложных многокомпонентных систем // Москва, 2015. 343 с.
10. Два подхода к проектированию информационных систем // URL: http://consulting.ru/econs_art_417882730 (дата обращения: 24.10.2016)
11. *Доронина Ю.В., Рябовая В.О.* Метод модернизации информационных систем экологического мониторинга на основе анализа их функциональной нагрузки // Труды СПИИРАН. 2016. Вып. 1(44). С. 133–153.
12. *Дегтярев Ю.И.* Методы оптимизации: учебное пособие // М.: Сов. Радио, 1986.
13. *Латыпов Н.Н., Елкин С.В., Гаприлов Д.А.* Инженерная эвристика / под ред. А.А. Вассермана // М.: Астрель, 2012. 320 с.
14. *Цвиркун А.Д.* Основы синтеза структуры сложных систем // М.: Наука, 1982. 200 с.
15. *Цвиркун А.Д., Акинфиев В.И.* Структура многоуровневых и крупномасштабных систем (синтез и планирование развития) // М.: Наука, 1993. 157 с.
16. *Цвиркун А.Д., Акинфиев В.И., Фильмонов В.А.* Имитационное моделирование в задачах синтеза структуры сложных систем: Оптимизационно-имитационный подход // М.: Наука, 1985. 174 с.
17. *Цурков В.И.* Динамические задачи большой размерности // М.: Наука, 1988. 288 с.
18. Области применения AnyLogic // URL: <http://www.anylogic.ru/application-areas>, свободный (дата обращения: 24.10.2016)
19. *Каширина И.Л.* Генетический алгоритм решения квадратичной задачи о назначениях специального вида // Вестник ВГУ . 2003. Вып. 1. С. 66–71
20. *Жуков Д.В., Матвиш В.А., Мочалов В.Ф., Труфанов А.Ф.* Системный анализ актуальных прикладных задач наземно-аэрокосмического мониторинга эколого-технологических объектов, исследуемых в проекте ELRI-184 // Труды СПИИРАН. 2013. Вып. 28. С. 107–121.
21. *Павлов А.Н.* Комплексное моделирование структурно-функциональной реконфигурации сложных объектов // Труды СПИИРАН. 2013. Вып. 28. С. 143–168.

Доронина Юлия Валентиновна — д-р техн. наук, доцент, профессор кафедры Информационных систем Института информационных технологий и управления в технических системах Севастопольского государственного университета. Область научных интересов: совершенствование и реинжиниринг информационных систем, системная инженерия, технологии СУБД, системы поддержки принятия решений. Число научных публикаций — 100. juvado@yandex.ru; СевГУ, ул. Университетская, д. 33, г. Севастополь, 299053, РФ; р.т. +7(8692)435-038.

Рябовая Валентина Олеговна — старший преподаватель кафедры Информационных систем Института информационных технологий и управления в технических системах Севастопольского государственного университета. Область научных интересов: системный анализ, теория вероятности и экспертных оценок, системы поддержки принятия решений, применение методов структурной оптимизации в системах мониторинга. Число научных публикаций — 35. valentina_rb@mail.ru; СевГУ, ул. Университетская, д. 33, г. Севастополь, 299053, РФ; р.т. +7(8692)435-038.

Чесноков Дмитрий Игоревич — магистр кафедры Информационных систем Института информационных технологий и управления в технических системах Севастопольского государственного университета. Область научных интересов: системный анализ, генетические алгоритмы, системы поддержки принятия решений, применение методов структурной оптимизации в модельно-ориентированных системах. dimchesn@gmail.com; СевГУ, ул. Университетская, д. 33, г. Севастополь, 299053, РФ; р.т. +7(8692)435-038.

Y.V. DORONINA, V.O. RYABOVAYA, D.I. CHESNOKOV
**USING MODEL-BASED DESIGN FOR SOLVING STRUCTURAL
 SYNTHESIS PROBLEMS**

Doronina Y.V., Ryabovaya V.O., Chesnokov D. I. Using Model-based Design for Solving Structural Synthesis Problems.

Abstract. To reduce the complexity of the task of structural synthesis, it is divided into stages, during each of which the researcher, conducts (with the help of decision support systems) the synthesis and analysis of model systems for the given input requirements and restrictions. Structural optimization, in this context, is reduced to finding the extremum of a certain objective function, whose value is controlled by specified design parameters depending on the type of task.

To demonstrate how the simulation model works, the functional synthesis of the structure of information enterprise management system is considered, where the functional elements are the automated business processes, and the structural elements — automation facilities. A test case is made, where typical processes of budgeting, marketing, purchasing and sales, production and human resources are described as the functional elements.

The use of the developed model of functional synthesis is exemplified by the task of choosing software for the design of corporate information systems. On the basis of a series of experiments we have determined the set of possible solutions with the greatest value of the fitness function. It is established that the function value is affected by the number of redundant functions that contain selected structural elements.

Keywords: model-based design, structural synthesis, information system, genetic algorithm, directed mutation, extremum, a heuristic approach

Doronina Julia Valentinovna — Ph.D., Dr. Sci., associate professor, professor of department of the Information systems of Institute of information technologies and management in the technical systems of the Sevastopol state university. Research interests: perfection and reengineering of the informations systems, systems engineering, technologies of databases, systems of support of making a decision. The number of publications — 100. juvado@yandex.ru; SEVGU, St. University, 33, Sevastopol, 299053, Russia; office phone +7(8692) 435-038.

Ryabovaya Valentina Olegovna — senior lecturer of department of the Information systems of Institute of information technologies and management in the technical systems of the Sevastopol state university. Research interests: system analysis, probability theory and expertise, decision support systems, application of structural optimization in monitoring systems. The number of publications — 35. valentina_rb@mail.ru; SEVGU, St. University, 33, Sevastopol, 299053, Russia; office phone +7(8692) 435-038.

Chesnokov Dmitry Igorevich — master of department of the Informative systems of Institute of information technologies and management in the technical systems of the Sevastopol state university. Research interests: system analysis, genetic algorithms, decision support systems, application of structural optimization in model-oriented systems. dimchesn@gmail.com; SEVGU, St. University, 33, Sevastopol, 299053, Russia; office phone +7(8692) 435-038.

References

1. Imboden D., Pfenninger S. Introduction to Systems Analysis: Mathematically Modeling Natural Systems. *Berlin, New York, Springer*. 2013. no. 8. pp. 235–252.
2. Tugolukov E.N., Tkachev A.G., Ruhov A.V. et al. *Proektirovanie slozhnyh sistem* [Design of complex systems]. Tambov. 2008. 32 p. (In Russ.).

3. Kocjuba I.Ju. *Osnovy proektirovaniya informacionnyh sistem* [Basics of designing information systems]. S.-Peterburg. 2015. 206 p. (In Russ.).
4. Ohtilev M. Ju., Sokolov B.V., Jusupov R.M. *Intellektual'nye tehnologii monitoringa sostojanija i upravlenija strukturnoj dinamikoj slozhnyh tehnicheskikh objektov* [Intelligent technology for monitoring the status and management of the structural dynamics of complex technical objects]. Moscow: Nauka. 2005. 291 p. (In Russ.).
5. Potrjasaev S.A., Sokolov B.V., Jusupov R.M. [Substantive and formal description of the problem of structural-functional synthesis and Development Management Information System ground-space monitoring]. *Trudy SPIIRAN – SPIIRAS Proceedings*. 2013. vol. 28. pp. 82–106. (In Russ.).
6. Zadachi strukturnogo sinteza i prinjatija reshenij [The structural synthesis tasks and decision-making]. Available at: http://www.life-prog.ru/2_70003_zadachi-strukturnogo-sinteza-i-prinyatiya-resheniy.html (accessed: 24.10.2016) (In Russ.).
7. Zelenskij V.A. *Proektirovanie slozhnyh sistem* [Design of complex systems]. Samara: Samarskij nacional'nyj issledovatel'skij universitet imeni akademika S.P. Koroleva. 2012. 96 p. (In Russ.).
8. Baranov I.A. *Model'no-orientirovannyj podhod v razrabotke programmyh komponent dlja kompleksov SM1820M* [Model-based approach in the development of software components for SM1820M complexes]. Moscow: OAO «INJeUM im. I.S. Bruka». 2012. 11 p. (In Russ.).
9. Brodskij Ju.I. *Problema opisaniya i sinteza raspredelennyh imitacionnyh modelej slozhnyh mnogokomponentnyh sistem* [Problem description and synthesis of distributed simulation models of complex multicomponent systems]. Moscow. 2015. 343 p. (In Russ.).
10. *Dva podhoda k proektirovaniyu informacionnyh sistem* [Two approaches to the design of information systems]. Available at: http://consulting.ru/econs_art_417882730, svobodnyj (accessed: 24.10.2016) (In Russ.).
11. Doronina Ju.V., Rjabovaja V.O. [Modernization of information systems for environmental monitoring based on their functionality]. *Trudy SPIIRAN – SPIIRAS Proceedings*. 2016. № 1 (44). pp. 133–153. (In Russ.).
12. Degtjarev Ju.I. *Metody optimizacii: uchebnoe posobie* [Optimization Methods: Tutorial]. Moscow. 1986. (In Russ.).
13. Latypov N.N., Jolkin S.V., Gavrilov D.A. *Inzhenernaja jevrystika* [Engineering heuristics] / pod.red. A.A. Vassermana. Moscow: Astrel'. 2012. 320 p. (In Russ.).
14. Cvirkun A.D. *Osnovy sinteza struktury slozhnyh sistem* [Basics of synthesis of the structure of complex systems]. Moscow: Nauka. 1982. 200 p. (In Russ.).
15. Cvirkun A.D., Akinfiev V.I. *Struktura mnogourovnevnyh i krupnomasshtabnyh sistem (sintez i planirovanie razvitiya)* [The structure of multi-level and large-scale systems (synthesis and development planning)]. Moscow: Nauka. 1993. 157 p. (In Russ.).
16. Cvirkun A.D., Akinfiev V.I., Filimonov V.A. *Imitacionnoe modelirovanie v zadachah sinteza struktury slozhnyh sistem: Optimizacionno-imitacionnyj podhod* [Simulation modeling in the design of the structure of complex systems optimization-simulation approach.]. Moscow: Nauka. 1985. 174 p. (In Russ.).
17. Curkov V.I. *Dinamicheskie zadachi bol'shoj razmernosti* [Dynamic problems of large dimension]. Moscow: Nauka. 1988. 288 p. (In Russ.).
18. Oblasti primeneniya AnyLogic [Applications of AnyLogic]. Available at: <http://www.anylogic.ru/application-areas>, svobodnyj (accessed: 24.10.2016) (In Russ.).
19. Kashirina I.L. [Genetic algorithm for solving quadratic assignment problem of a special kind]. *Vestnik Voronezhskogo gosudarstvennogo universiteta – Bulletin of Voronezh State University*. 2003. no. 1. pp. 66–71. (In Russ.).
20. Zhukov D.V., Mat'jash V.A., Mochalov V.F., Trufanov A.F. [System analysis of topical applications ground-space monitoring ecological and technological objects, studied in the project ELRI-184]. *Trudy SPIIRAN – SPIIRAS Proceedings*. 2013. № 28. pp. 107–121. (In Russ.).
21. Pavlov A.N. [Complex modeling of structural and functional reconfiguration of complex objects]. *Trudy SPIIRAN – SPIIRAS Proceedings*. 2013. № 28. pp. 143–168. (In Russ.).