

УДК 621.391.7; 519.876.5

DOI: 10.18413/2518-1092-2019-4-4-0-8

Егоров И.А.¹
Бузов А.А.²
Забнин С.А.¹
Зайцев А.Н.¹

**ФОРМАЛЬНЫЕ ОСНОВЫ ПРОЦЕДУР ОПТИМИЗАЦИИ
СИСТЕМНО-ОБЪЕКТНЫХ ИМИТАЦИОННЫХ МОДЕЛЕЙ
ПРОЦЕССОВ И СИСТЕМ**

¹) Белгородский государственный национальный исследовательский университет, ул. Победы, д. 85, г. Белгород, 308015, Россия

²) АО «ОЭЗ «ВЛАДМИВА»», ул. Студенческая, д. 19, г. Белгород, 308018, Россия

e-mail: 888615@bsu.edu.ru, buzov@mail.ru, 509725@bsu.edu.ru, 433044@bsu.edu.ru

Аннотация

В работе рассматриваются некоторые методы оптимизации системно-объектных имитационных моделей процессов и систем. С целью повышения эффективности системно-объектного анализа организационно-деловых и производственно-технологических процессов за счет совершенствования теоретических и инструментальных средств оптимизации системно-объектных имитационных моделей авторами сформулированы некоторые принципы оптимизации, которые рассматриваются в статье. В работе показано, что оптимизация системно-объектной модели, в первую очередь, требуется с целью установления соответствия модели общесистемным принципам и закономерностям. При построении имитационных моделей авторы руководствуются, в том числе, обобщенной характеристикой системы – мерой системности. В данной статье предложена оптимизация системно-объектной имитационной модели на примере организационно-управленческой модели «руководитель-подчиненные» в среде «UFOModeler». Обосновывается, что сформулированные принципы оптимизации позволяют перейти к разработке методов оптимизации структурных, функциональных и объектных параметров сходных моделируемых систем. Также, следует отметить, что рассматриваемые в работе принципы оптимизации системно-объектных имитационных моделей не являются исчерпывающими.

Ключевые слова: системно-объектная модель; оптимизация модели; общесистемные принципы и закономерности; UFOModeler; принцип моноцентризма.

UDC 621.391.7; 519.876.5

Egorov I.A.¹
Buzov A.A.²
Zabnin S.A.¹
Zaitsev A.N.¹

**FORMAL BASES OF OPTIMIZATION PROCEDURES
OF SYSTEM-OBJECT IMITATION MODELS OF PROCESSES
AND SYSTEMS**

¹) Belgorod State National Research University, 85 Pobedy St., Belgorod, 308015, Russia

²) JSC «SEZ «VLADMIVA»», 19 Studentcheskaya St., Belgorod, 308018, Russia

e-mail: 888615@bsu.edu.ru, buzov@mail.ru, 509725@bsu.edu.ru, 433044@bsu.edu.ru

Annotation

The paper discusses some optimization methods for system-object simulation models of processes and systems. In order to increase the efficiency of the system-object analysis of organizational, business and industrial-technological processes by improving the theoretical and instrumental means of optimizing system-object simulation models, the authors formulated some optimization principles that are considered in the article. The paper shows that the optimization of the system-object model, first of all, is required in order to establish the conformity of the model with system-wide principles and patterns. When constructing simulation models, the authors are guided, inter alia, by the generalized characteristic of the system – a measure of systemicity. This article proposes the optimization of a system-object simulation model using the organizational-managerial model of "supervisor-subordinates" in the environment of "UFOModeler". It is proved that the formulated optimization principles allow us to proceed to the development of

optimization methods for structural, functional and object parameters of similar simulated systems. Also, it should be noted that the principles of optimization of system-object simulation models considered in the work are not exhaustive.

Keywords: system-object model; model optimization; system-wide principles and patterns; UFOModeler the principle of monocentrism.

АКТУАЛЬНОСТЬ

Ввиду бурного развития науки и техники, в современном мире, разработчики, проектировщики, инженеры все чаще прибегают к применению имитационного моделирования как метода исследования объектов и процессов окружающего мира. Причем, следует отметить, что исследуемые объекты с каждым годом усложняются с точки зрения их структуры, поведения и управления такими объектами, в то же время, для исследования и управления такими сложными системами требуются соответствующие средства и методики. Одной из таких методик является – имитационное моделирование, благодаря которому имеются возможности построения симуляторов – специализированные программные комплексы, замещающие объект или процесс реального мира с достаточной степенью точности. Системно-объектное имитационное моделирование – современный метод построения имитационных моделей, основанный на оригинальном графоаналитическом подходе «Узел-Функция-Объект». Причем, главной особенностью системно-объектных имитационных моделей является возможность построения модели, соответствующей общесистемным принципам и закономерностям [1]. Актуальность рассматриваемых задач обуславливается необходимостью приведения моделируемых систем в соответствие общесистемным принципам и закономерностям, что показано в работе [2].

МЕТОДЫ

Для формальной постановки задачи воспользуемся положениями исчисления систем как функциональных объектов [3].

В терминах упомянутого выше исчисления, модель системы представлена в виде:

$$M = L, S, \quad (1)$$

где:

- M – модель системы;
- L – множество потоковых объектов модели, элементы которого представляют собою объекты, которые не имеет методов и имеют лишь поля:

$$l=[r_1, r_2, \dots, r_k], \quad (2)$$

где:

- $l \in L$;
- k – количество полей потокового объекта l;
- r_1, r_2, \dots, r_k – поля потокового объекта, представляющие собой пару «идентификатор-значение»;
- S – множество узловых объектов модели.

Ранее авторами были сформулированы ряд методов оптимизации, которые подробно показаны в работе [4]. Рассмотрим некоторые формальные основы процедур оптимизации системно-объектных моделей систем, представляемых в терминах исчисления систем как функциональных объектов. Здесь в качестве критериев оптимальности, предлагается использовать, в первую очередь, некоторые общесистемные принципы и закономерности [1]. Рассмотрим подробнее принцип коммуникативности, согласно которому любая система должна быть связана множеством коммуникаций с окружающей средой (другими системами), иначе существование системы не имеет никакого смысла. Если рассматривать системно-объектную

модель вида (1), то в терминах исчисления систем, данный принцип можно формально представить в следующем виде:

$$\forall s \in S: \exists s. U = \emptyset \quad (3)$$

Так как, в терминах исчисления систем, связи узлового объекта с внешней средой представлены в форме его интерфейсных потоковых объектов $U=(L_?, L_!)$, тогда соответствие моделируемой системы принципу коммуникативности определяется отсутствием в модели узловых объектов, для которых множество интерфейсных потоковых объектов является пустым. Однако, может быть ситуация, когда, например, у узлового объекта имеются входные потоковые объекты и не имеется выходных потоковых объектов и наоборот, такая ситуация также противоречит принципу коммуникативности, поэтому данный принцип можно уточнить в следующей форме:

$$\forall s \in S: \exists s. L_? = \emptyset \parallel s. L_! = \emptyset \quad (4)$$

Таким образом, процесс оптимизации модели по принципу коммуникативности будет заключаться в связывании имеющихся узловых объектов с другими.

РЕЗУЛЬТАТЫ

Рассмотрим пример оптимизации принципу коммуникативности модели организационно-делового процесса. Воспользуемся инструментарием для моделирования систем UFOModeler [5]. Представим модель следующего вида (Рис. 1).

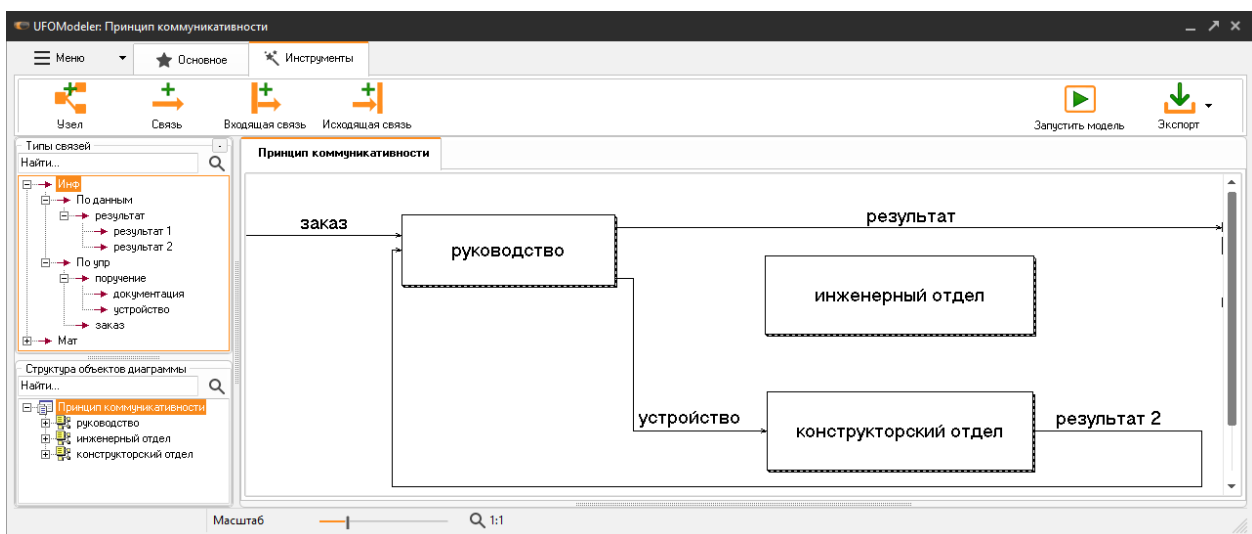


Рис. 1. Пример модели организационно-делового процесса
Fig. 1. An example of a business process model

Обозначим потоковые объекты модели, представленной модели:

- 11 – «документация» (на рисунке 1 обозначена в меню «Типы связей»);
- 12 – «результат 1» (на рисунке 1 обозначена в меню «Типы связей»);
- 13 – «устройство»;
- 14 – «результат 2»;
- 15 – «заказ»;
- 16 – «результат».

При этом каждый из потоковых объектов содержит поле потокового объекта: r1 – «количество».

Тогда множество потоковых объектов примет вид:

$$L = \{ l_1[r_1], l_2[r_1], l_3[r_1], l_4[r_1], l_5[r_1], l_6[r_1] \} \quad (5)$$

Приведем формальные имена узлов представленной выше модели:

- s_1 – «руководство»;
- s_2 – «инженерный отдел»;
- s_3 – «конструкторский отдел».

Тогда, согласно формальному описанию, множество узловых объектов будет выглядеть следующим образом:

$$S = \{ s_1[l_1 = \{l_4, l_5\}, l_1 = \{l_3, l_6\}; f(l_1)l_1; 0], s_2[l_2 = \emptyset, l_1 = \emptyset; f(l_2)l_1; 0], s_3[l_3 = \{l_3\}, l_1 = \{l_4\}; f(l_3)l_1; 0] \} \quad (6)$$

В соответствии со структурой множества узловых объектов модели (6), в представленной модели организационно-делового процесса узел «руководство» имеет входной интерфейс $l_1 = \{l_4, l_5\}$, через который контролирует результаты выполнения поручений (сборка устройства) конструкторским отделом и получает заказы от надсистемы, а также имеет выходной интерфейс $l_1 = \{l_3, l_6\}$, через который выдает поручения для исполнения узлам подсистемы и предоставляет результат работы над заказом надсистеме. Узел «конструкторский отдел» принимает поручения в обработку через входящий интерфейс $l_1 = \{l_3\}$. При выполнении поручения конструкторский отдел отчитывается о выполнении перед руководством через выходной интерфейс $l_1 = \{l_4\}$.

Стоит отметить, что потоковые объекты l_1, l_2 объединяющие узлы «руководство» и «инженерный отдел» отсутствуют, в таком случае, инженерный отдел будет простаивать из-за отсутствия поручений (составление документации) к исполнению. Таким образом, рассматриваемая модель не отвечает принципу коммуникативности. Проведем оптимизацию по принципу коммуникативности. Применим к рассматриваемой модели операторы присоединения:

$$s_1 \xrightarrow{l_1} s_2; s_2 \xrightarrow{l_2} s_1 \quad (7)$$

Выполним действие (7) на диаграмме модели и запустим модель в среде UFOModeler на выполнение в реальном времени. Результаты оптимизации модели представлены ниже (Рис. 2).

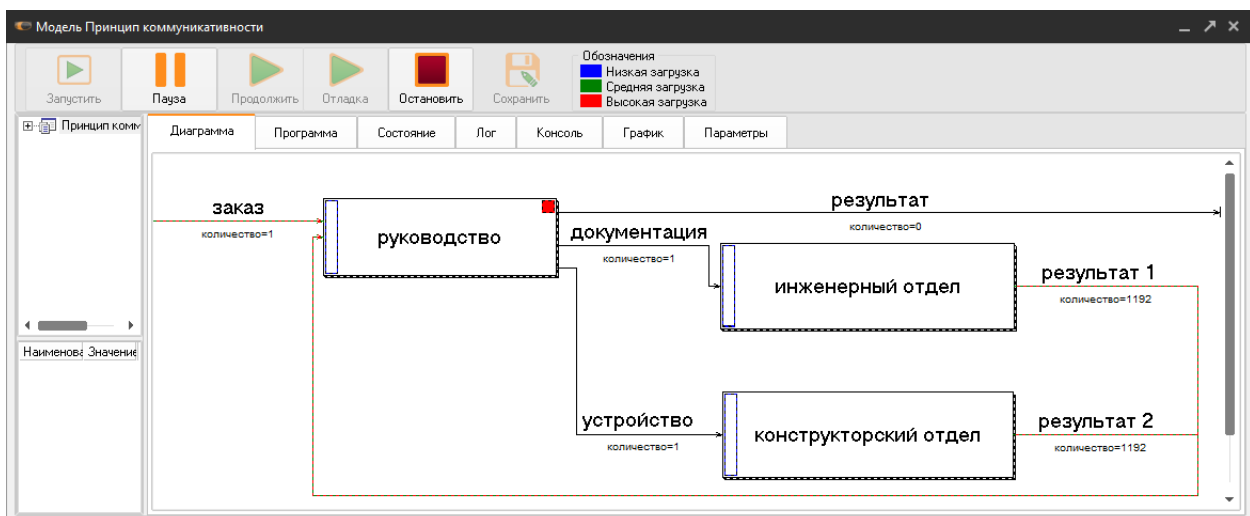


Рис. 2. Функционирование оптимизированной модели

Fig. 2. Optimized Model Performance

Динамика поступления новых поручений и их выполнения отделами с течением времени приведена ниже (рис. 3).

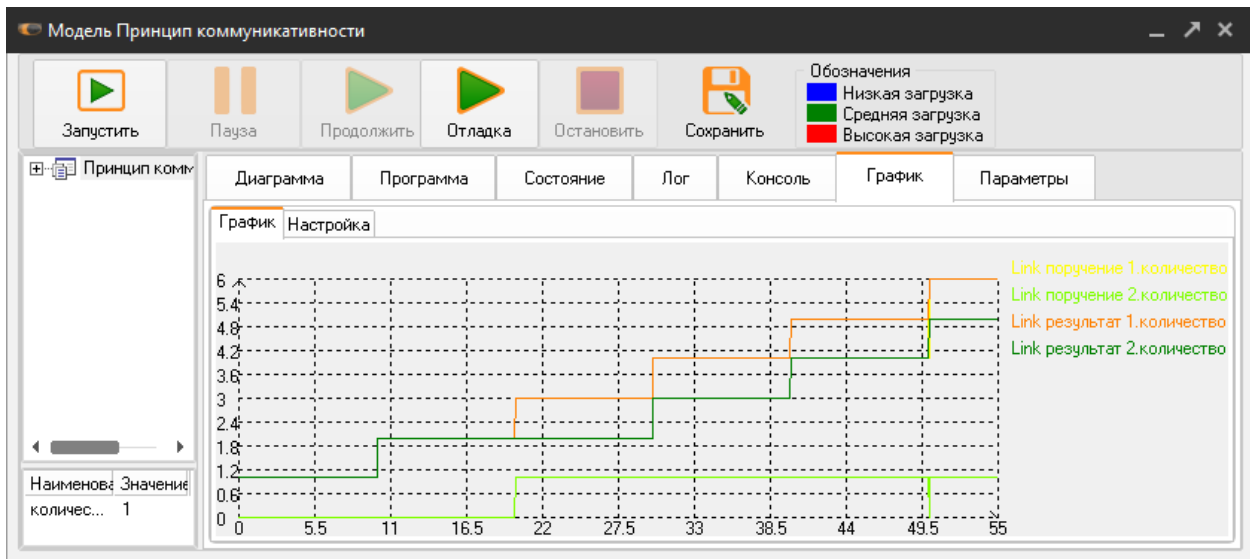


Рис. 3. Рост количества выполненных поручений в оптимизированной модели
Fig. 3. Growth in the number of completed orders in an optimized model

Из графика видно, что с течением времени растет количество выполненных поручений обоими отделами, при этом к исполнению принимается в каждый момент времени одно новое поручение.

Таким образом, если какой-либо узел системы не коммуницирует с внешним миром через собственные интерфейсы, то система является неоптимальной с точки зрения принципа коммуникативности. Стоит отметить, что такая организация, в которой узлы не связаны потоковыми объектами не сможет в полной мере функционировать, что приведет в итоге к снижению заинтересованности в её услугах и снижению количества заказов.

Далее рассмотрим процесс оптимизации модели по принципу обратной связи, согласно которому устойчивость в сложных динамических системах достигается за счет замыкания петель обратных связей. Формально, данный принцип можно представить в виде следующего выражения:

$$\exists s_i \in S, s_j \in S: l_m \in s_i \cdot L?, l_m \in s_j \cdot L!, l_n \in s_j \cdot L?, l_n \in s_i \cdot L! \quad (8)$$

Соответственно, оптимизация модели по принципу обратной связи заключается в присоединении соответствующими видами связей узловых объектов, используя операцию присоединения.

Приведем пример оптимизации организационно-правовой модели системы по принципу обратной связи. Составим модель в среде UFOModeler, как представлено ниже (рис. 4).

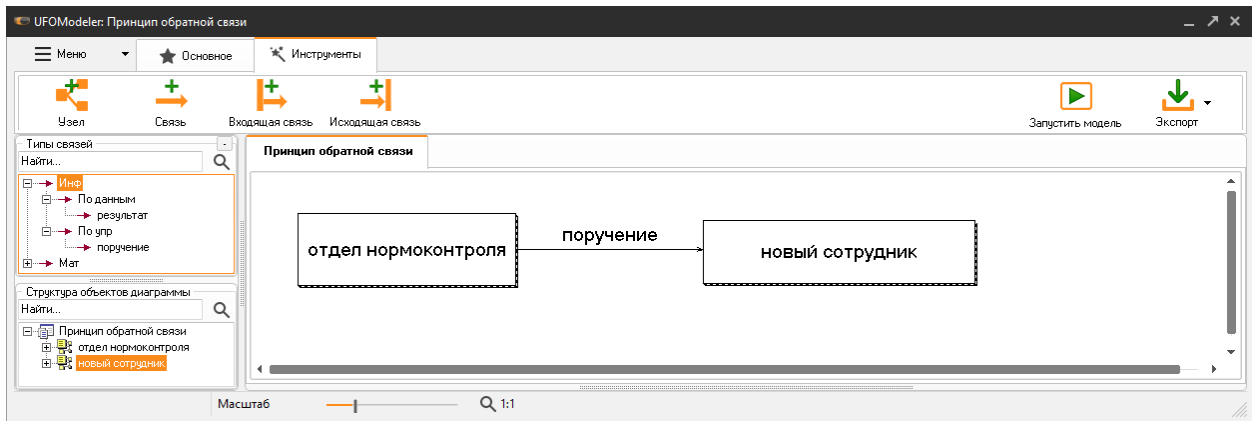


Рис. 4. Пример замкнутых связей системы
Fig. 4. An example of closed system connections

Обозначим потоковые объекты модели, представленной модели:

- l_1 – «поручение»;
- l_2 – «результат» связь между новым сотрудником и отделом отсутствует на диаграмме (на рисунке 4 наличие связи отражено в окне «Типы связей»).

Аналогично предыдущему примеру, представленные потоковые объекты обладают полем потокового объекта: r_1 – «количество». Множество потоковых объектов будет представлено следующим образом:

$$L = \{ l_1[r_1], l_2[r_1] \} \quad (9)$$

Имена узлов, представленных в модели:

- s_1 – «отдел нормоконтроля»;
- s_2 – «новый сотрудник».

Формальное описание множества узловых объектов представлено ниже:

$$S = \{ s_1[l_1 = \emptyset, l_1 = \{l_1\}; f(l_1)l_1; O], s_2[l_2 = \{l_1\}, l_2 = \emptyset; f(l_2)l_2; O] \} \quad (10)$$

В модели рассматривается организация испытательного срока нового сотрудника в отделе нормоконтроля.

Узел «отдел нормоконтроля» выдает поручение через выходной интерфейс $l_1 = \{l_1\}$ к исполнению. Узел «новый сотрудник» принимает поручение через входной интерфейс $l_2 = \{l_1\}$ и приступает к выполнению поручения.

Однако, из-за отсутствия связи l_2 между отделом нормоконтроля, в лице руководителя отдела, не может контролировать процесс выполнения новым сотрудником поручений. Руководитель отдела не может оценить среднюю скорость выполнения поручений сотрудником:

- сотрудник выполнил все поручения, но не сообщает руководству о готовности выполнения новых поручений;
- сотрудник не справляется с переданными ему поручениями, отдел передает на выполнение слишком большое количество поручений.

Таким образом, производительность труда резко снижается. Рассматриваемая модель не отвечает принципу обратной связи.

Проведем оптимизацию по принципу обратной связи. Осуществим присоединение связи l_2 и запишем действие в виде выражения:

$$s_2 \xrightarrow{l_2} s_1 \quad (11)$$

Добавим потоковый объект l_2 на диаграмму модели. Далее запустим модель на исполнение в реальном времени (Рис. 5).

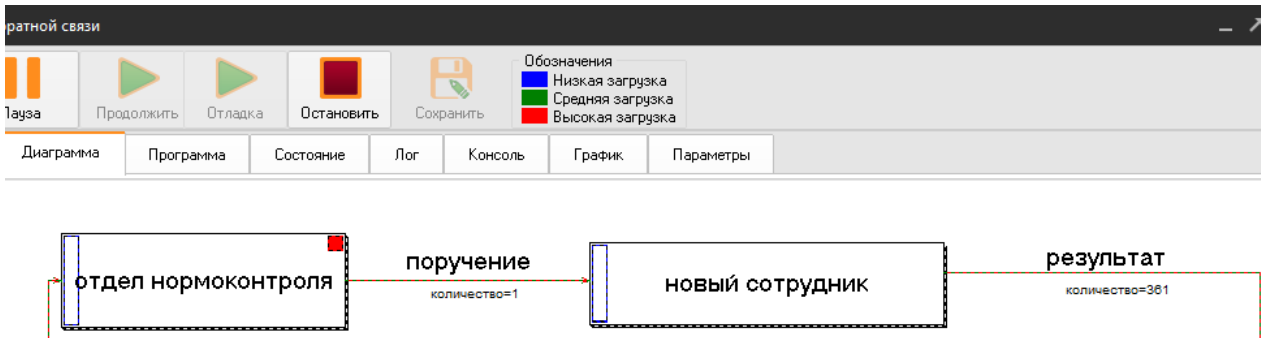


Рис. 5. Выполнение оптимизированной модели в реальном времени
Fig. 5. Real time optimized model execution

Динамика роста количества выполненных поручений, и количества принятых к исполнению поручений коррелирует с предыдущим примером. График динамики аналогичен рисунку 3.

Стоит отметить негативный аспект в функционировании структурных единиц предприятий при несвоевременном отчете сотрудников об исполнении поручений.

Таким образом, с целью повышения производительности труда, если существует связь «руководитель-подчиненный», то для увеличения эффективности производства необходима оптимизация по принципу обратной связи.

Далее рассмотрим, вытекающий из принципа коммуникативности, принцип моноцентризма, согласно которому устойчивая система обладает одним центром, а полицентричность приводит к нарушению процессов координации, что в перспективе обуславливает потерю целостности [6].

Формально принцип моноцентризма можно представить в следующем виде:

$$\forall s \in S: \exists s. L_s = \{l_1\} \parallel s. L_s = \{l_1, l_2 \dots l_m\} \quad (12)$$

Таким образом, оптимизация модели (1) по принципу моноцентризма заключается в устранении потоковых связей, которые позволяют нарушить процесс координации подчиненных объектов. Устранение связей осуществляется следующими методами:

- объединение управляющих объектов через выходные потоковые объекты;
- выстраивание иерархии среди управляющих объектов.

В первом случае объекты надсистемы формируют один общий управляющий потоковый объект. Данный метод применим при выполнении дополнительных условий для узлов s_i и s_j , у которых $L_{i1} = L_{j1}$ и $O_{i1} = O_{j1}$. То есть, управляющие объекты должны иметь одинаковые выходные интерфейсы и подобные потоковые объекты.

Во втором случае оптимизация по принципу моноцентризма может быть достигнута через выстраивание цепочки управляющих объектов в иерархию, таким образом, каждый последующий объект становится подчиненным вышестоящему вплоть до целевого. При этом каждый объект в выстроенной иерархии при формировании собственного управляющего сигнала учитывает, что ему было передано через входной интерфейс от вышестоящего узла. В результате каждый объект цепочки оказывает свое опосредованное управляющее воздействие на конечный объект.

Рассмотрим в качестве примера модель организационно-делового процесса, в которой приведен механизм взаимодействия структурных единиц научно-производственного предприятия при осуществлении планирования предстоящих задач. Модель составлена в среде UFOModeler (рис. 6).

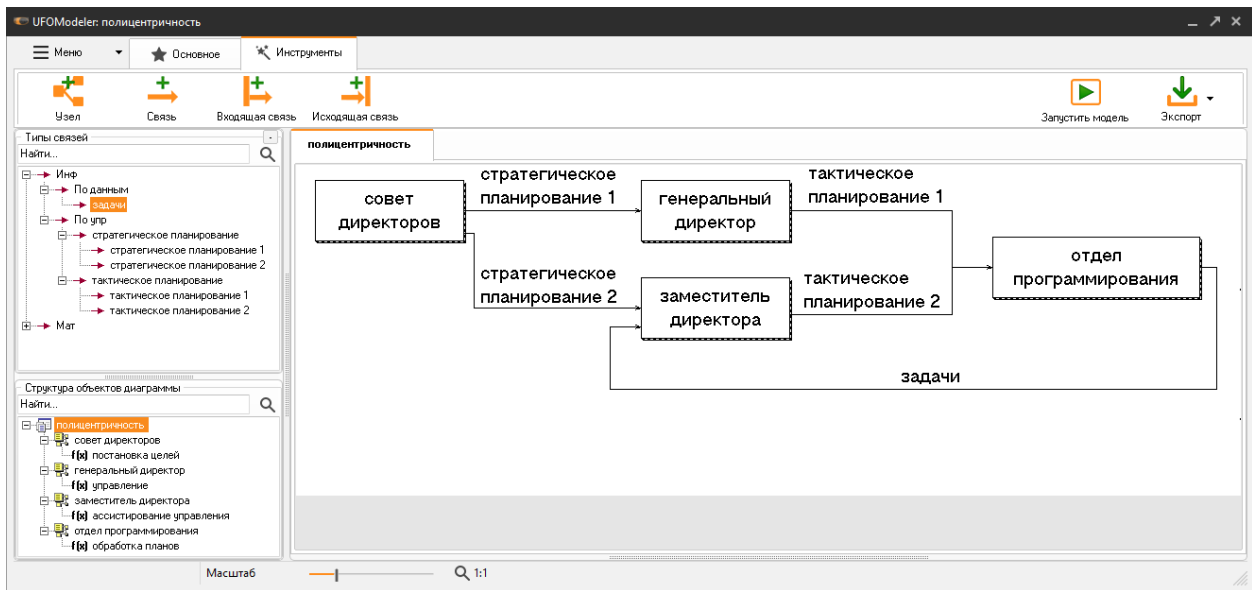


Рис. 6. Модель научно-производственного предприятия
Fig. 6. Model of a research and production enterprise

Данная модель не отвечает требованиям принципа моноцентризма, так как на диаграмме присутствуют связи «тактическое планирование 1» и «тактическое планирование 2», которые могут оказывать на объект узла «отдел программирования» взаимно-антагонистическое управляющее воздействие, которое, в конечном счете нарушает целостность системы. Результат выполнения имитации такой модели в реальном времени в отношении производительности узла «отдел программирования» сводится к графику функции, обозначенного на рисунке 3 салатным цветом (меньшая эффективность).

Приведем список объектов, участвующих в составлении формального описания оптимизационных действий над моделью. Поточковые объекты в представленной модели обозначим следующим образом:

- l_1 – «стратегическое планирование 1»;
- l_2 – «стратегическое планирование 2»;
- l_3 – «тактическое планирование 1»;
- l_4 – «тактическое планирование 2»;
- l_5 – «задачи».

При этом потоковые объекты l_1 - l_4 обладают полем потокового объекта r_1 – «план» строкового типа, а потоковый объект $l_5 - r_0$ – «выполнено» целочисленного типа. Соответственно множество потоковых объектов примет вид:

$$L = \{ l_1[r_1], l_2[r_1], l_3[r_1], l_4[r_1], l_5[r_0] \} \quad (13)$$

Приведем формальные имена узлов представленной выше модели:

- s_1 – «совет директоров»;
- s_2 – «генеральный директор»;
- s_3 – «заместитель директора»;
- s_4 – «отдел программирования».

Тогда, согласно формальному описанию, множество узловых объектов будет выглядеть следующим образом:

$$S = \{ s_1[l_7 = \emptyset, l_1 = \{l_1, l_2\}; f(l_7)l_1; O], s_2[l_7 = \{l_1\}, l_1 = \{l_3\}; f(l_7)l_1; o], s_3[l_7 = \{l_2, l_5\}, l_1 = \{l_4\}; f(l_7)l_1; O], s_4[l_7 = \{l_3, l_4\}, l_1 = \{l_5\}; f(l_7)l_1; O] \} \quad (14)$$

В модели узлы s_1, s_2, s_3 осуществляют управляющее воздействие через связи согласно выстроенной иерархической структуре предприятия. Управляющее воздействие отражается, во-первых, в целях, которые вынес узел s_1 , в виде стратегического плана перед старшим менеджментом предприятия (узлы s_2, s_3), во-вторых, в сформулированных тактических задачах, которые ставит старший менеджмент предприятия перед отделами (для примера взят отдел программирования, узел s_4). О выполнении задач узел s_4 отчитывается перед узлом s_3 , передавая данные о количестве выполненных задач.

Также считаем, что входные интерфейсы узлов s_2 и s_3 получили равные значения полей потоковых объектов, что является важным критерием при рассмотрении примера оптимизации методом «выстраивания иерархии среди управляющих объектов».

Разделим на два шага оптимизацию по методу «объединения управляющих объектов через выходные потоковые объекты». На первом шаге необходимо объединить узлы s_2 и s_3 в общую подсистему «старший менеджмент» (рис. 7).

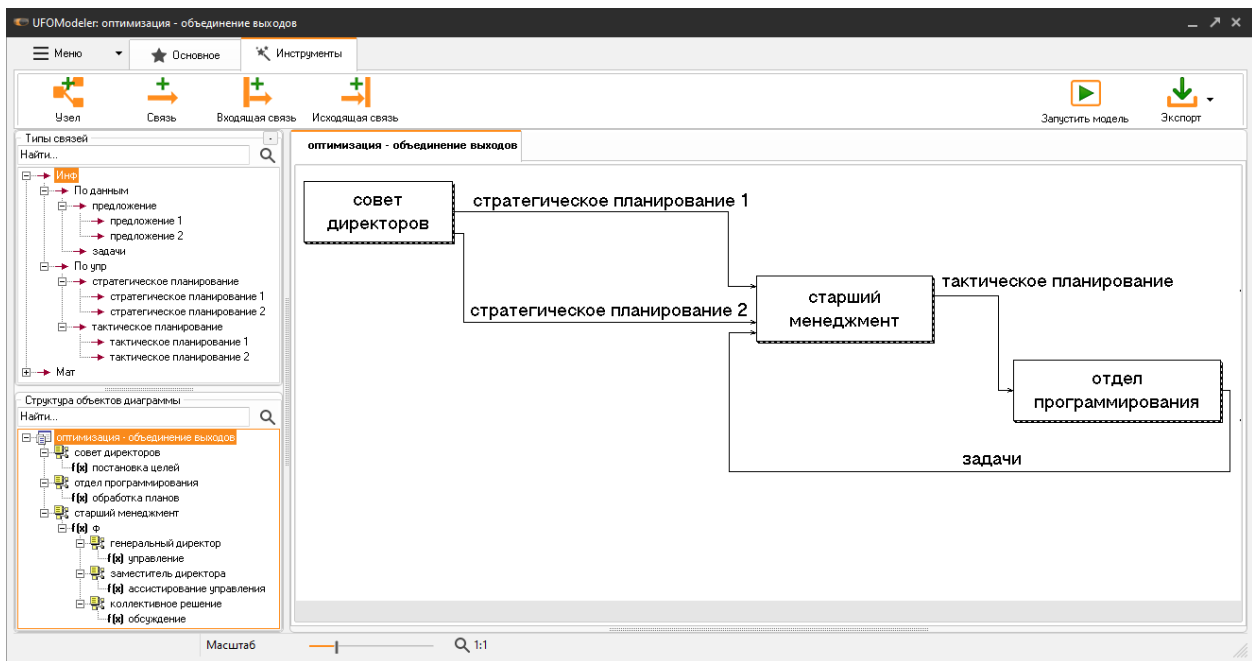


Рис. 7. Объединение менеджмента предприятия в общую подсистему

Fig. 7. Combining enterprise management into a common subsystem

Обозначим подсистему «старший менеджмент» через множество S^* . Присоединим входные потоковые объекты l_1 и l_2 , а также выходной потоковый объект «тактическое планирование» l_8 . Таким образом, потоковые объекты l_3 и l_4 более не имеют смысла и будут удалены из множества L . Формально, приведенное выше действие можно записать следующим образом:

$$L = L \setminus l_3, l_4, s_1 \xrightarrow{l_1} S^*, s_1 \xrightarrow{l_2} S^*, S^* \xrightarrow{l_8} s_4 \quad (15)$$

Тогда множество S^* будет содержать следующие элементы:

$$S^* = \{ s_2[l_7 = \{l_1\}, l_1 = \{l_3\}; f(l_7)l_1; o], s_3[l_7 = \{l_2, l_5\}, l_1 = \{l_4\}; f(l_7)l_1; O] \} \quad (16)$$

Удалим из множества S узлы, которые образуют подсистему S^* , и образуем новое множество узлов S' оптимизированной модели:

$$S = S \setminus S^*, S' = S \cup S^* \quad (17)$$

Далее опишем сформированную подсистему S^* . В ней представлены узлы «генеральный директор» s_2 , «заместитель директора» s_3 . Дополним подсистему узлом «коллективное решение» s_{23} , показывающий совместную обработку данных двух узлов.

Стоит отметить, что объединение узлов по выходу в среде UFOModeler достижимо через создание третьего узла и добавление двух новых потоковых объектов l_6 – «предложение 1» и l_7 – «предложение 2», отвечающих за передачу выдвинутого на обсуждение решения в узел s_{23} (аналог объединения потоковых объектов), и далее подсистема формирует выходной потоковый объект l_8 (рис. 8).

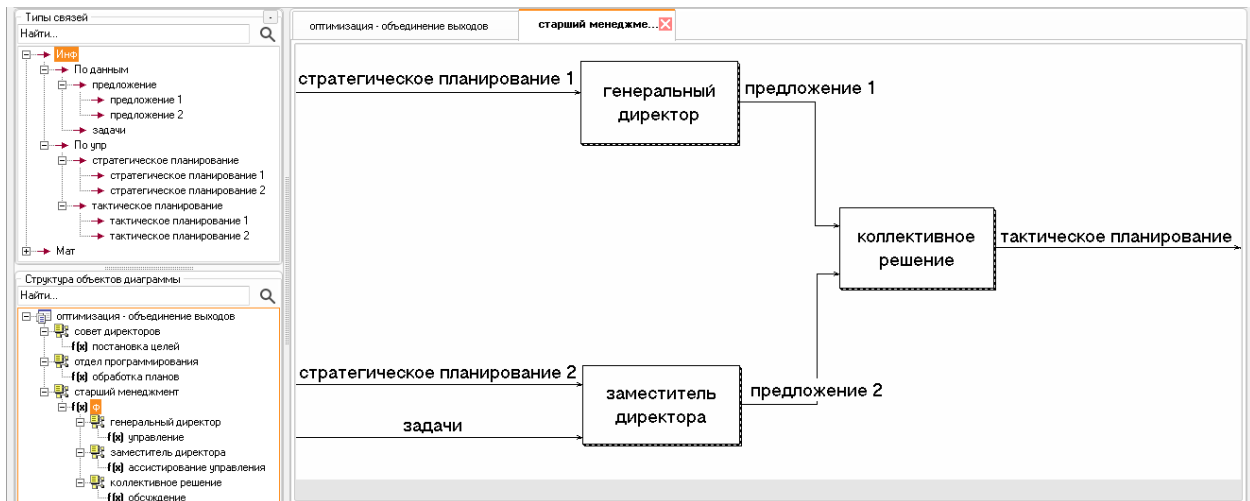


Рис. 8. Выстраивание узловых объектов подсистемы «старший менеджмент»

Fig. 8. Alignment of nodal objects of the subsystem "senior management"

Представим формальное описание выполнения второго шага рассматриваемого метода:

$$s_2 \xrightarrow{l_6} s_{23}, s_3 \xrightarrow{l_7} s_{23}, s_{23} \xrightarrow{l_8} s_4 \Rightarrow s_{23}[l_2=\{l_6, l_7\}, l_1=\{l_8\}; f(l_2)l_1; o], S^* = S' \cup \{s_{23}\} \quad (18)$$

Так как в формальном представлении узел s_{23} является имитацией объединения потоковых объектов при сопутствующем изменении функции объекта для узла s_3 .

$$s_3.f \Leftarrow [f(l_{73})l_{13} \rightarrow f(l_{22}, l_{73})l_{12}; l_{22}; l_{13} \rightarrow l_{12}; o_{13} \rightarrow o_{12}, o_{23}] \quad (19)$$

Запустим имитацию модели в реальном времени (рис. 9).

Из результатов работы модели видно, что через входные интерфейсы подсистема S^* получила управляющее воздействие от узла s_1 (поля r_1 потоковых объектов l_1 и l_2 содержат значение «создание современного ПО»).



Рис. 9. Оптимизация методом выстраивания иерархии среди управляющих объектов
Fig. 9. Optimization by building hierarchies among control objects

Далее представим работу как выглядит работа подсистемы S^* . Объекты узлов s_2 и s_3 , внесли предложения по реализации стратегического плана через потоковые объекты l_6 и l_7 на обсуждение (поле $l_6 \cdot r_1$ потокового объекта содержит значение «создание администраторской части»; поле $l_7 \cdot r_1$ потокового объекта содержит значение «создание пользовательской части»), пришли к общему мнению (поле r_1 потокового объекта l_8 содержит значение «создание администраторской части»), передали общее управляющее воздействие через потоковый объект l_7 , и, таким образом, входной интерфейс узла s_4 «отдел программирования» получил однозначное управляющее воздействие. В соответствии с поставленным тактическим планом отдел программирования выстроил оперативный план, в рамках которого необходимо реализовать 30 задач. Синий цвет в столбце прогресса в левой части узла «отдел программирования» означает количество выполненных задач. График функции выполнения задач с течением времени совпадает с функцией оранжевого цвета на рисунке 3.

Осуществим оптимизацию методом «выстраивания иерархии среди управляющих объектов». Изменим направление потокового объекта l_3 в исходной модели. Удалим связь между узлом s_2 и s_4 и организуем потоковый объект l_3 между узлами s_2 и s_3 (рис. 10).

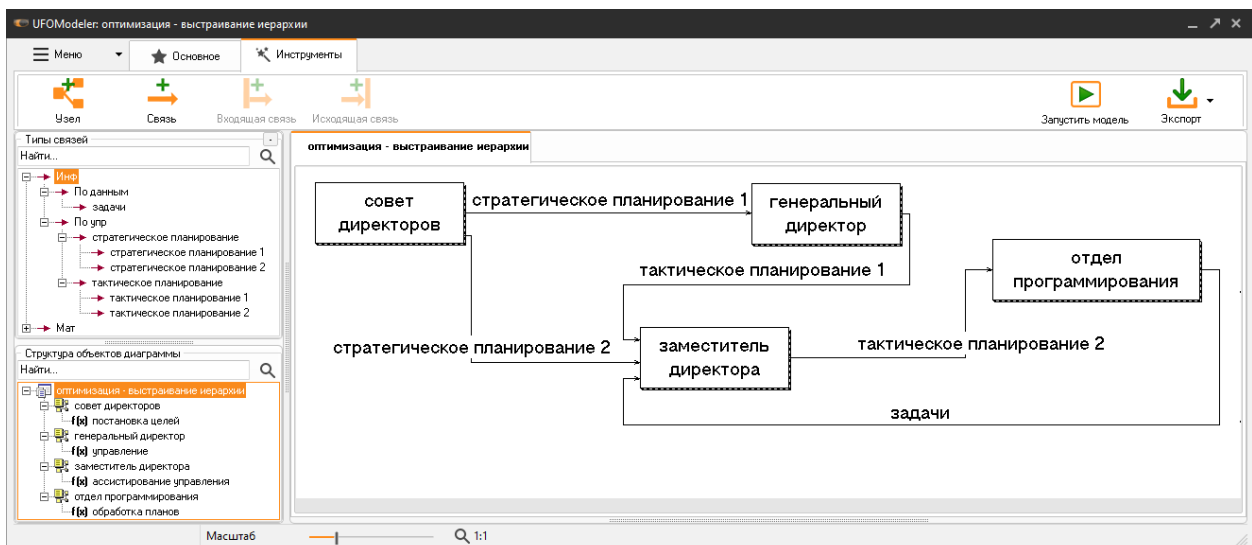


Рис. 10. Диаграмма взаимосвязей оптимизированной модели
Fig. 10. Optimized Model Relationship Diagram

Запишем формальное представление данного метода:

$$s_2 \cdot L_1 = L_1 \setminus \{l_3\}, s_4 \cdot L_2 = L_2 \setminus \{l_3\}, s_2 \xrightarrow{l_3} s_3 \quad (20)$$

Запустим модель на исполнение в реальном времени. Результат работы модели в целом сопоставим с результатом, представленным на рисунке 9. С тем различием, что управляющее воздействие, вынесенное узлом s_2 по отношению к узлу s_3 носит рекомендательный характер и может противоречить управляющему воздействию узла s_3 по отношению к узлу s_4 . Данный аспект отличает два описанных метода оптимизации. При этом, нельзя считать входящие в узел s_3 потоковые объекты l_2 и l_3 полицентрическими, так как оказываемое связями управляющее воздействие не противоречит друг другу (в основе управляющего воздействия потоковых объектов l_3 лежит управляющее воздействие узла s_1) и не нарушает координации управляемого объекта, следовательно, в данном случае принцип моноцентризма не нарушен.

ВЫВОДЫ

Таким образом, объект узла s_4 «отдел программирования», в оптимизированной по принципу моноцентризма модели, занимается решением задач, согласованных со всеми управляющими звеньями. При этом узел s_4 не испытывает дополнительного управляющего воздействия, что непременно скажется на скорости выполнения задач положительным образом.

Рассмотренные принципы оптимизации по соответствующим общесистемным принципам и закономерностям показывают, что наличие корректно выстроенных связей необходимо для создания эффективной, коммуницирующей системы. В то же время жизненный опыт любой организационной системы всегда доказывает, что соответствие предложенным принципами закономерностям делает организационную систему устойчивой к проявлениям внешней среды.

В перспективе на основе сформулированных принципов оптимизации системно-объектных моделей процессов и систем авторами будут разработаны методы и алгоритмы оптимизации моделей, что позволит автоматизировать данные процедуры.

Благодарности. Исследования выполнены при финансовой поддержке проектов Российского фонда фундаментальных исследований № 18-07-00355, № 19-07-00290 и № 19-07-00111.

Список литературы

1. С.И. Маторин, А.Г. Жихарев Общесистемные закономерности как содержательные элементы системной теории, основанной на системно-объектном подходе // Научные ведомости Белгородского государственного университета. Серия: экономика, информатика, том 45, № 2, стр. 372-385, 2018 г.
2. Маторин С.И., Белов С.П., Жихарев А.Г. Учет общесистемных закономерностей при моделировании сложных систем средствами системно-объектного подхода // Наука. Инновации. Технологии, № 3, 2018, с. 7-21.
3. Matorin, S.I., Zhikharev, A.G. Calculation of the function objects as the systems formal theory basis // Advances in Intelligent Systems and Computing, 679, pp. 182-191, 2018.
4. Жихарев А.Г., Бузов А.А., Егоров И.А., Кузнецов А.В., Жинкина Ю.В. Оптимизация системно-объектных имитационных моделей. Часть 1. // Научные ведомости БелГУ. Сер. Экономика. Информатика. – 2019. – № 2. – Том 46. С. 311-325.
5. Маторин С.И., Жихарев А.Г., Зайцева Н.О. Имитационное моделирование с использованием системно-объектного подхода. // Прикладная информатика. – 2015. – №6(60). – Выпуск 10.
6. Богданов А. А. Тектология: Всеобщая организационная наука. В 2-х книгах// Москва, «Экономика», 1989.

References

1. S.I. Matorin, A.G. Zhikharev System-wide regularities as the content elements of a system theory based on a system-object approach // Scientific reports of Belgorod State University. Series: Economics, Informatics, Volume 45, No. 2, pp. 372-385, 2018.
2. Matorin S.I., Belov S.P., Zhikharev A.G. Taking into account system-wide laws when modeling complex systems using the system-object approach // Science. Innovation Technology, No. 3, 2018, pp. 7-21
3. Matorin, S.I., Zhikharev, A.G. Calculation of the function objects as the systems formal theory basis // Advances in Intelligent Systems and Computing, 679, pp. 182-191, 2018.
4. Zhikharev A.G., Buzov A.A., Egorov I.A., Kuznetsov A.V., Zhinkina Yu.V. Optimization of system-object simulation models. Part 1. // Scientific statements of BelSU. Ser. Economy. Informatics. – 2019. – No. 2. – Volume 46.S. 311-325.
5. Matorin S.I., Zhikharev A.G., Zaitseva N.O. Simulation using a system-object approach. // Applied Informatics. – 2015. – No. 6 (60). – Issue 10.
6. Bogdanov A. A. Tectology: General organizational science. In 2 books // Moscow, "Economics", 1989.

Егоров Илья Александрович, аспирант кафедры информационных и робототехнических систем

Бузов Андрей Анатольевич, кандидат технических наук, технический директор

Забнин Сергей Александрович, аспирант кафедры информационных и робототехнических систем

Зайцев Анатолий Николаевич, аспирант кафедры прикладной информатики и информационных технологий

Egorov Ilya Alexandrovich, postgraduate student of the Department of Information and Robotic Systems

Buzov Andrey Anatolevich, Candidate of Technical Sciences, technical director

Zabnin Sergey Aleksandrovich, postgraduate student of the Department of Information and Robotic Systems

Zaitsev Anatoly Nikolaevich, postgraduate student of the Department of Applied Informatics and Information Technology