

**Корнилов Василий Сергеевич** – Южный федеральный университет; e-mail: koresh-jr@yandex.ru; 347928, г. Таганрог, пер. Некрасовский, 44; кафедра САПР; аспирант.

**Глушань Валентин Михайлович** – e-mail: gluval07@rambler.ru; кафедра САПР; профессор.

**Лозовой Алексей Юрьевич** – e-mail: lozovoy@sfedu.ru; кафедра ИЯ; доцент.

**Kornilov Vasilii Sergeevich** – Southern Federal University; e-mail: koresh-jr@yandex.ru; 44, Nekrasovskiy, Taganrog, 347928, Russia; the department of computer aided design; postgraduate student.

**Glushan Valentin Mihailovich** – e-mail: gluval07@rambler.ru; the department of computer aided design; professor.

**Lozovoy Aleksey Yurievich** – e-mail: lozovoy@sfedu.ru; the department of foreign languages; associate professor.

УДК 004.023

**Ю.А. Кравченко, И.О. Курситыс**

### **КОМБИНИРОВАННЫЙ ПОДХОД К РЕШЕНИЮ ЗАДАЧИ РАСПРЕДЕЛЕНИЯ РЕСУРСОВ\***

*Данная статья посвящена решению задачи распределения вычислительных ресурсов в компьютерных системах. Общей смысл задачи распределения ресурсов – распределить ограниченный ресурс между потребителями оптимальным образом. В работе сформулирована постановка задачи и приведена ее математическая модель. Задача распределения вычислительных ресурсов относится к классу NP-сложных, что обуславливает невозможность или ограниченность применения традиционных средств математического анализа для ее решения. Разработанный авторами подход основан на использовании имитационного моделирования и аппарата сетей Петри для распределения ресурсов между процессами. Так как распределяемый ресурс является вычислительным, главным критерием оптимизации принято время, затрачиваемое на исполнение всех процессов, использующих ресурсы, а для моделирования задачи используются временные сети Петри. Комбинированный подход к решению задачи заключается в совместном применении аппарата сетей Петри для моделирования задачи, а также генетических алгоритмов, успешно показавших себя в решении NP-сложных задач. Для оптимизации модели применены известные генетические алгоритмы и модифицированные генетические операторы кроссинговера и мутации. Авторами разработан программный продукт – подсистема распределения ресурсов. В статье отражены основные функции подсистемы, ее модульная структура. Построена и проанализирована информационная модель подсистемы на основе стандарта IDEF0, отражающая информационные процессы, протекающие в подсистеме. Проведены вычислительные эксперименты на тестовых примерах (бенчмарках), доказывающие эффективность и перспективность разработанного подхода.*

*Распределение ресурсов; CALS-технологии; имитационное моделирование; сети Петри; генетические алгоритмы.*

**Yu.A. Kravchenko, I.O. Kursitys**

### **COMBINED APPROACH FOR SOLVING RESOURCE ALLOCATION PROBLEM**

*The paper is devoted to solving the computative resource allocation problem. In general, the resource allocation problem involves allocating the limited resource among processes in optimal way. Authors formulated the problem statement and its mathematical model. The resource allocation*

---

\* Работа выполнена при финансовой поддержке РФФИ (проект № 17-07-00446).

tion problem is referred as NP-complete, thus, the usage of traditional mathematical means is impossible or limited in terms of solving this problem. The developed approach is based on simulation modeling usage and Petri Nets tools to allocate resources. Since allocated resources are considered as computative, time spent for all processes execution and all Petri Net transition performance is taken as the main optimization criterion. Thus, authors apply temporal Petri Nets to model the resource allocation problem. The combined approach for solving computative resource allocation problem involves simultaneous application of Petri Net tools to model the problem, and genetic algorithms, which have proved their success in solving NP-complete problems. To optimize the Petri Net model authors use well-known genetic algorithms and modified genetic operators of crossover and mutation. Authors developed software representing the resource allocation subsystem. The paper describes main function of the subsystem and shows developed module structure of it. To represent main information processes flowing in subsystem, two information models on the basis of the IDEF0 standard are built and analyzed. Authors carried out computational experiments based on test benchmarks, which proved the effectiveness of the developed approach.

Resource allocation; CALS; simulation; Petri nets; genetic algorithm.

**Введение.** Задача распределения ресурсов является одной из фундаментальных задач, решение которых требуется во многих областях науки и техники. Общий смысл задачи – распределить ограниченный ресурс между процессами оптимальным образом.

От эффективности алгоритма распределения ресурсов во многом зависит эффективность всей вычислительной системы в целом. Например, неверный алгоритм управления ресурсами может повлечь за собой риск перевода процесса в состояние ожидания на неопределенное время. Такое явление, называемое «отстранением», отражает нерациональность распределения ресурсов. Технология планирования ресурсов обязана учитывать такую возможность и избегать ее. Когда процессы используют несколько ресурсов, возможны два других нежелательных последствия [1]:

1. Взаимной блокировкой называется неопределенная по времени блокировка нескольких процессов, каждый из которых ждет освобождения ресурсов, занятых другими. Избыточное число запросов на некоторые виды ресурсов может привести к ухудшению качества системы в целом.

2. «Живая» блокировка представляет собой другое явление – один процесс захватил какой-либо разделяемый по доступу ресурс, но из-за недостатка других ресурсов находится в ожидании, тем самым переводя в режим ожидания другие процессы, желающие получить доступ к тому же ресурсу.

В данной статье предлагается использовать комбинированный подход решения задачи вычислительных ресурсов с совместным использованием имитационного моделирования и генетических алгоритмов.

**1. Постановка задачи.** Сформулируем математическую модель задачи распределения ресурсов.

Пусть  $T=\{1,2,\dots,T_n\}$  – множество тактов планирования, ( $n$  – количество тактов),  $J$  – множество всех работ,  $I$  – множество различных ресурсов,  $r_{ij}$  – ресурсоемкость работы  $j$  по ресурсу  $i$ ,  $m_{ij}$ ,  $M_{ij}$  – минимальная и максимальная интенсивности потребления работой  $j$  ресурса  $i$ ,  $V_{it}$  – количество ресурса  $i$ , которое в такт  $t$  поступает в систему.

Варьируемые параметры математической модели:  $x=(x_1,\dots,x_{/J})$  – вектор времен начала выполнения работ,  $y=(y_1,\dots,y_{/J})$  – вектор времен окончания выполнения работ,  $z_{ijt}$  – интенсивность потребления работой  $j$  ресурса  $i$  в такт  $t$ ,  $i \in I$ ,  $j \in J$ ,  $t \in T$ .

**В рамках математической модели введены следующие ограничения:**

1) варьируемые параметры модели  $\langle x_j, y_j, z_{ijt}, i \in I, j \in J, t \in T \rangle$  определяют множество частично-целочисленных неотрицательных переменных:

$$\begin{aligned} x_j \in T, y_j \in T, j \in J, \\ z_{ijt} \geq 0, i \in I, j \in J, t \in T; \end{aligned} \quad (1)$$

2) взаимозависимость работ, определяющая каноничность сетевой модели, задается ограничениями:

$$x_j \geq \max_{l \in K(j)} y_l, j \in J; \quad (2)$$

3) ограничения на интенсивности потребления работами ресурсов и требования выполнения работ без перерывов:

$$\begin{aligned} m_{ij} \leq z_{ijt} \leq M_{ij}, x_j \leq t \leq y_j, i \in I, j \in J, t \in T, \\ z_{ijt} = 0, t \notin [x_j, y_j], i \in I, j \in J, t \in T; \end{aligned} \quad (3)$$

4) ограничения на длительности выполнения работ:

$$\begin{aligned} t_j^- \leq y_j - x_j, j \in J, \\ y_j - x_j \leq t_j^+, j \in J. \end{aligned} \quad (4)$$

Полное использование необходимых ресурсов означает полное выполнение работ:

$$\sum_{t \in T} z_{ijt} = r_{ij}, i \in I, j \in J. \quad (5)$$

Обозначим через

$$P_{it} = \sum_{t'=1}^t V_{it'} - \sum_{j \in J} \sum_{t'=1}^{t+n_j-1} z_{ijt'} - \sum_{t'=1}^{t-1} \max(0, P_{it'}), \quad (6)$$

где  $\max(0, P_{it})$  – потери ресурса  $i$ , поступившего в систему в такт  $t$ , из-за истечения срока его годности.

Тогда

$$W_{it} = \sum_{t'=1}^t V_{it'} - \sum_{j \in J} \sum_{t'=1}^{t-1} z_{ijt'} - \sum_{t'=1}^{t-n_i} \max(0, P_{it'}) \quad (7)$$

количество ресурса  $i$ , которое может быть использовано в такт  $t$  для выполнения работ. С учетом введенных ограничений, ресурсные условия в общем случае имеют вид:

$$\sum_{j \in J} z_{ijt} \leq W_{it}, i \in I, j \in J. \quad (8)$$

Задача заключается в определении совокупности варьируемых параметров, удовлетворяющих системе ограничений (1)–(4), для которых минимального значения достигает обобщенный критерий оптимальности (8). Рассмотрим совокупность частных критериев оптимальности, задаваемых кусочно-линейными функциями:

$$\varphi_{ij}(\sum_{j \in J} z_{ijt}, W_{it}) = \begin{cases} \alpha_{ij} \cdot \frac{W_{it} - \sum_{j \in J} z_{ijt}}{W_{it}} \cdot 100, \text{ если } W_{it} \geq \sum_{j \in J} z_{ijt} \\ \beta_{it} \cdot \frac{\sum_{j \in J} z_{ijt} - W_{it}}{W_{it}} \cdot 100, \text{ в противном случае} \end{cases} \quad (9)$$

$$i \in I, t \in T$$

Здесь  $\alpha_{ij}, \beta_{it}$  – соответственно, коэффициенты, определяющие штрафные санкции за один процент избытка и недостатка ресурса  $i$  в такт  $t, i \in I, t \in T$ . В качестве обобщенного критерия оптимальности выберем функционал

$$\Phi(x, y, z) = \max_{i \in I} \max_{t \in T} \varphi_{it}(\sum_{j \in J} z_{ijt}, W_{it}) \rightarrow \min, \quad (10)$$

который определяет стремление минимизировать штрафные санкции за неритмичность расходования ресурсов по тактам планирования [2, 3].

Задача распределения вычислительных ресурсов является NP-сложной, что ограничивает применение средств математического анализа для ее решения. Необходимо обращаться к группе методов, которые можно использовать в ситуациях, выходящих за рамки системы предпосылок, на которых основаны простые модели. Одним из таких методов является имитационное моделирование.

В качестве инструмента решения задачи распределения вычислительных ресурсов выбрано имитационное моделирование на основе сетей Петри, который обладает рядом достоинств [4, 7]:

- 1) позволяет моделировать асинхронность и недетерминизм параллельных процессов;
- 2) позволяет учитывать время выполнения тех или иных событий при моделировании;
- 3) включает возможности других моделей, предложенных для параллельных систем, позволяя описывать как типовые ситуации в данных системах (распределение ресурсов, взаимные блокировки), так и общую динамику работы сложной асинхронной системы.

Сеть Петри определяется как кортеж  $\langle P, T, I, O \rangle$ , где  $P$  и  $T$  являются конечными множествами позиций и переходов соответственно, а  $I$  и  $O$  – множествами входных и выходных функций [4–6].

Объекты в сетях Петри подразделяются на динамические (маркеры) и статические (вершины). Процесс распределение маркеров по позициям называется *маркировкой*. Каждое изменение маркировки называется *событием*, которое связано с определенным переходом. События происходят мгновенно и одновременно при выполнении некоторых условий. Каждому условию в сети Петри соответствует определенная *позиция*. Совершению события соответствует срабатывание перехода, при котором маркеры из входных позиций этого перехода перемещаются в выходные позиции. Последовательность событий образует моделируемый процесс [4–6].

**2. Имитационная модель задачи.** При использовании имитационного моделирования задача будет переформулирована следующим образом.

Пусть имеется  $n$  параллельных процессов и  $m$  ресурсов, которые необходимы для выполнения данных процессов. Каждый процесс имеет по  $p_n$  позиций и  $t_n$  переходов. Ресурсы представляют собой позиции  $m_i$ , фишки которых могут быть переданы либо в переход процесса  $n_i$ , либо в переход соседнего процесса  $n_{i+1}$  [9].

Отметим, что сеть Петри, используемая для моделирования, является временной. Это означает, что все переходы имеют определенный вес, который представляет собой время срабатывания перехода. Так как моделируемые ресурсы являются вычислительными, время принимаем за главный критерий при оптимизации распределения ресурсов [8, 9]. Задача сводится к нахождению такой очереди срабатывания переходов, при которой время, затраченное на срабатывание, будет минимальным.

На рис. 1 проиллюстрирована передача ресурсов одному из процессов и их освобождение по окончании выполнения того или иного процесса. Начальная маркировка помечает ресурсы, заданные позициями  $p_4$  и  $p_5$  доступными и указывает на готовность процессов.

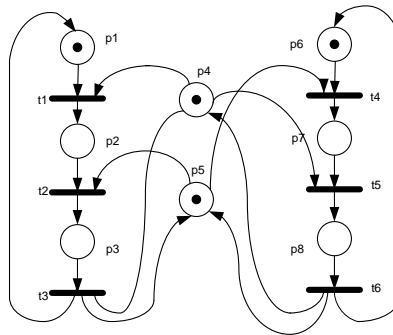


Рис. 1. Модель задачи распределения ресурсов для двух параллельных процессов

Одним из выполнений этой сети является последовательность переходов  $t_1t_2t_3t_4t_5t_6$ ; другим – последовательность  $t_4t_5t_6t_2t_3$ . В этом случае ни одно из выполнений данных последовательностей процессов не приводит к тупику. Рассмотрим последовательность, которая начинается переходами  $t_1t_4$ . Процесс, отображенный слева, обладает ресурсом  $p_4$  и хочет получить ресурс  $p_5$ . Процесс, отображенный справа, обладает ресурсом  $p_5$  и хочет получить ресурс  $p_4$ . Ни один из процессов не может продолжаться, так как система заблокирована.

Таким образом, дополнительным критерием для решения задачи распределения ресурсов будет являться отсутствие тупиков.

**3. Подсистема распределения ресурсов.** Разработанная подсистема распределения ресурсов выполняет следующие функции:

- 1) построение сети Петри, которая моделирует задачу распределения ресурсов;
- 2) присваивание весовых коэффициентов каждому из переходов;
- 3) решение задачи распределения ресурсов с помощью генетического алгоритма (ГА);
- 4) анализ полученного распределения ресурсов с помощью сети Петри;
- 5) построение графа достижимости на основе полученного решения задачи

Работа подсистемы начинается со считывания данных, введенных пользователем через интерфейс программы. Данную функцию выполняет модуль ввода и хранения данных. Кроме того, модуль отвечает за запись и хранение данных из других модулей, а также за своевременную передачу информации пользователю или частям программы.

После того, как данные введены, они передаются в модуль построения модели, где строится сеть Петри.

Модуль построения модели взаимодействует с модулем применения генетического алгоритма. Данные о сети Петри (переходы, их взаимосвязь и весовые коэффициенты) передаются в модуль применения ГА, который решает задачу распределения ресурсов. После того, как лучшее решение поставленной задачи найдено, данные об оптимальном решении задачи передаются в модуль анализа модели. Здесь строится граф достижимости. Далее происходит переход данных обратно в модуль построения модели, где строится заданная ранее сеть Петри с отраженным на ней решением.

Проведен анализ информационных процессов работы подсистемы по стандарту IDEF0 (рис. 2). Анализ информационных процессов позволяет убедиться в том, что подсистема распределения ресурсов в процессах удовлетворяет требованиям, сформулированным в техническом задании [4, 10]. Обратная связь отражена на диаграмме процесса работы подсистемы, где из блока  $A_2$  в блок  $A_1$  происходит передача данных о лучшем решении задачи распределения ресурсов, полученном из модуля применения ГА.

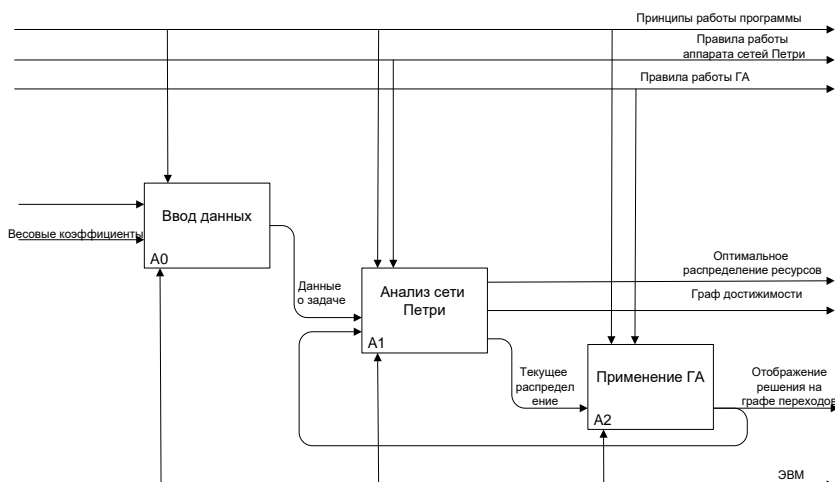


Рис. 2. Диаграмма процесса работы подсистемы

Указанное выше описание разработанной подсистемы отражено в укрупненной модульной структуре подсистемы (рис. 3).

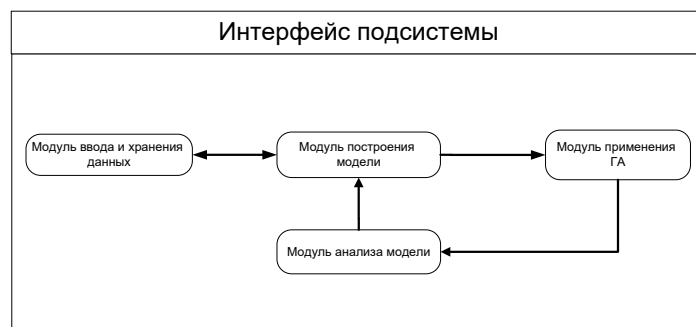


Рис. 3. Укрупненная модульная структура разработанной подсистемы

Модульная структура содержит следующие элементы [9]:

- ◆ интерфейс – через интерфейс осуществляется взаимодействие пользователя и подсистемы; интерфейс должен быть понятен и удобен для пользователя;
- ◆ модуль ввода и хранения данных – здесь происходит запись или извлечение данных для работы с программой;
- ◆ модуль построения модели – строит имитационную модель;
- ◆ модуль анализа модели – модуль, определяющий оптимальность текущего распределения ресурсов;
- ◆ модуль применения ГА – с помощью данного модуля осуществляется оптимизация сети Петри.

В качестве алгоритмов оптимизации графовой модели последовательности срабатывания переходов использованы три известных генетических алгоритма по схеме Холланда, Голдберга и Дэвиса (рис. 4) [13–15]. Для осуществления возможности решения задачи анализа сетей Петри, алгоритмы были доработаны. В них были включены блоки ввода сети Петри, формирование графа переходов сети, преобразование графа в полный граф, а также ввод весовых коэффициентов переходов.

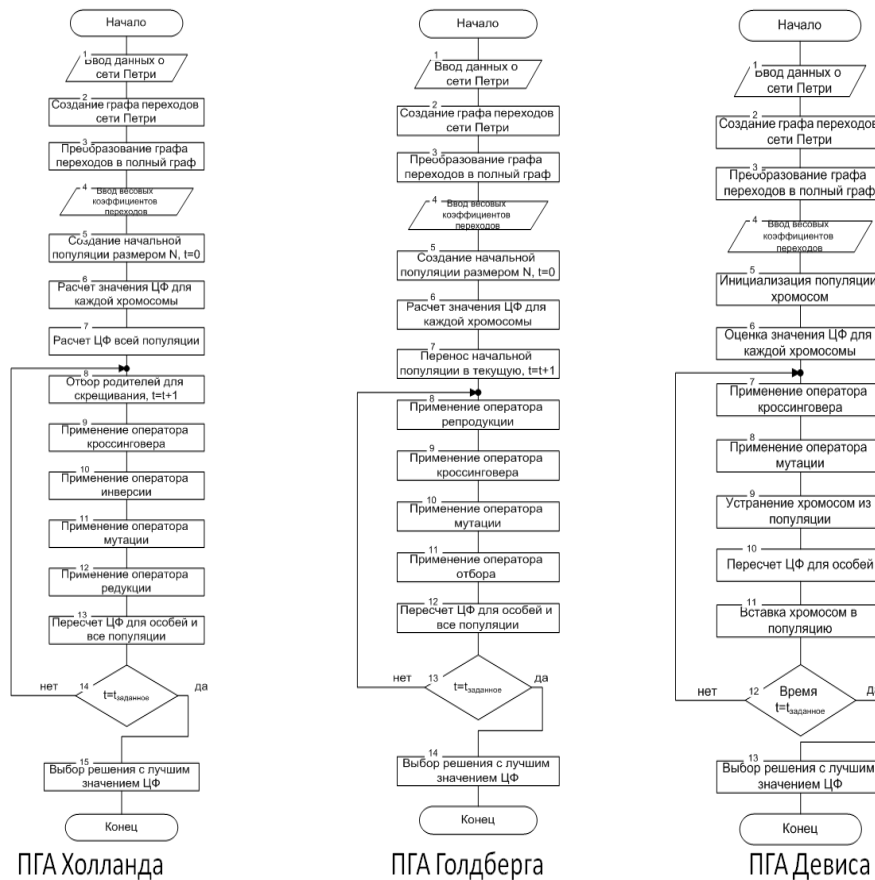


Рис. 4. Схемы генетических алгоритмов для решения поставленной задачи

Генетические алгоритмы обладают следующими преимуществами:

- 1) достаточно широкая область применения алгоритмов;
- 2) возможность комбинирования генетических алгоритмов с другими методами, в том числе и неэволюционными алгоритмами;
- 3) эффективность алгоритма в поиске решений в пространстве большой размерности;
- 4) отсутствие ограничений на вид целевой функции;
- 5) наглядность схемы построения генетических алгоритмов [11–20].

Для работы генетического алгоритма был применен ряд генетических операторов. Это двухточечный, модифицированный трехточечный и модифицированный жадный операторы кроссинговера; одноточечный оператор мутации, модифицированный оператор мутации на основе дихотомии, модифицированный оператор мутации с равновероятным распределением и оператор инверсии.

Разработанная подсистема была реализована в виде программы, интерфейс которой представлен на рис. 5. Задается сеть Петри, затем строится граф переходов сети. Граф переходов преобразовывается в полный граф. В матрице связей графа пользователь может задать весовые коэффициенты переходов. Затем выбираются настройки генетического алгоритма. Алгоритм запускается и решается задача оптимизации. На основании решения производится анализ сети Петри, т.е. строится граф достижимости и показывается оптимальное распределение ресурсов.

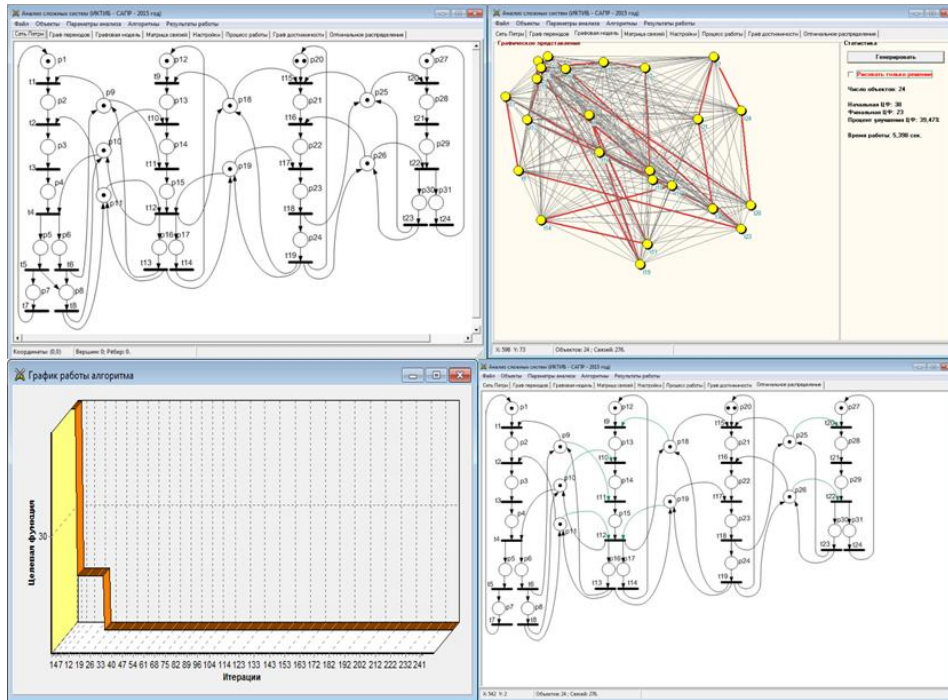


Рис. 5. Интерфейс подсистемы распределения ресурсов

**4. Экспериментальные исследования.** Экспериментальные исследования проводились с разным количеством переходов и позиций сети Петри – вершин графа переходов. Результаты показали, что временная сложность алгоритма полиномиальная  $O(n^2)$ . График временной сложности алгоритма показан на рис. 6.

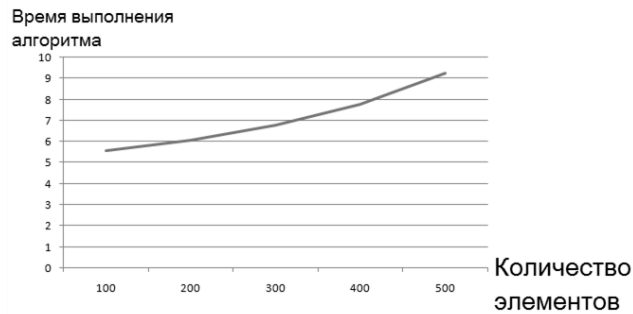


Рис. 6. График временной сложности алгоритма

На рис. 7 приведены сравнительные графики, отражающие результаты исследования эффективности работы генетического алгоритма и жадного алгоритма для решения поставленной задачи распределения ресурсов.

Разработанный генетический алгоритм в среднем на 37 % дает лучшие решения, чем жадный алгоритм, несмотря на то, что по времени работы жадный алгоритм немного превосходит генетический.



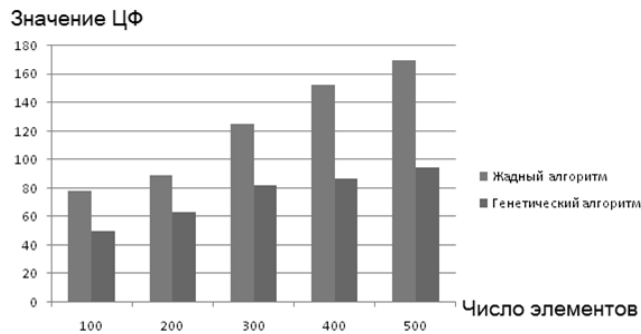


Рис. 7 Графики зависимости значения ЦФ от числа итераций при работе генетического и жадного алгоритмов

Подводя итоги, можно отметить, что разработанный генетический алгоритм показал преимущество в сравнении с жадным алгоритмом. При попадании в локальный оптимум увеличение числа итераций улучшает решение, но увеличивается время, затраченное алгоритмом на поиск решения. Разработанные генетические операторы позволяют повысить качество решения задачи. Возможность управления процессом генетического алгоритма путем изменения в настройках позволяет найти оптимальные параметры.

**Заключение.** Одним из самых эффективных методов оптимизации вычислительного процесса является внедрение параллельных схем вычислений в совокупности с применением вычислительных возможностей компьютерных сетей. В связи с этим, задача распределения вычислительных ресурсов между процессами всегда является актуальной. Данная задача является NP-сложной, поэтому применение традиционных средств математического анализа для ее решения затруднено. В работе разработан комбинированный подход для решения задачи распределения вычислительных ресурсов, который заключается в совместном применении аппарата сетей Петри и генетических алгоритмов. Сети Петри позволяют моделировать время выполнения тех или иных событий моделируемого процесса, а также учитывать его динамику. Генетические алгоритмы показали эффективность в нахождении квази-оптимальных решений в задачах большой размерности за приемлемое время.

В статье проведен обзор и анализ задачи, сформулирована ее математическая модель. Разработана подсистема распределения ресурсов и ее модульная структура, приведены основные функции. Построена информационная модель работы подсистемы на основе стандарта IDEF0, отражающая информационные процессы, протекающие в подсистеме.

На основе разработанной модульной структуры подсистемы распределения вычислительных ресурсов реализована программная среда. Проведены экспериментальные исследования разработанного подхода к решению задачи, которые показали его эффективность.

#### БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Белоусов С.М. Математическая модель многопоточной системы массового обслуживания, управляемой планировщиком ресурсов // Вестник Новосибирского государственного университета. Серия: информационные технологии. – 2006. – Т. 4, № 1. – С. 14-26.
2. Прилуцкий М.Х., Нефедов Д.С., Попов Д.В. Распределение ресурсов в дискретно управляемых системах // Электронный журнал “Исследовано в России”. – 2002. – № 5. – С. 322-337.

3. Прилуцкий М.Х., Вяхирев Д.В. Многостадийные задачи альтернативного распределения ресурсов // Вестник Нижегородского государственного университета. Математическое моделирование и оптимальное управление. – 2002. – Вып. 25 (1). – С. 36-43.
4. Yury Kravchenko, Ilona Kursitys, Dmitry Zaporozhets. Resource allocation system based on simulation modeling in computer-aided design system. In Conference proceedings // 2016 6th International Conference Cloud System and Data Engineering (Confluence). – 14-15 January 2016, Amity University Uttar Pradesh, Noida, India. – P. 395-400.
5. Кравченко Ю.А. CALS- и CASE-технологии: учебно-методическое пособие. – Таганрог, 2010. – 142 с.
6. Кравченко Ю.А. Cals-имитационные модели в управлении данными САПР // Известия ТРТУ. – 2006. – № 8 (63). – С. 143-146.
7. Эддоус М., Стэнсфилд Р. Методы принятия решений: пер. с англ. / под ред. член-корр. РАН И.И. Елисевой. – М.: Аудит, ЮНИТИ, 1997. – 590 с.
8. Курейчик В.М., Курейчик В.В. Эволюционные, синергетические и гомеостатические стратегии в искусственном интеллекте: состояние и перспективы // Новости искусственного интеллекта. – 2000. – № 3. – С. 39-67.
9. Кравченко Ю.А., Курсытыс И.О. Разработка модели задачи распределения ресурсов в процессах САПР и структуры подсистемы ее решения // Труды Конгресса по интеллектуальным системам и информационным технологиям «IS&IT'15». Научное издание в 4-х т. Т. 3. – М.: Физматлит, 2015.
10. Бова В.В., Курейчик В.В. Интегрированная подсистема гибридного и комбинированного поиска в задачах проектирования и управления // Известия ЮФУ. Технические науки. – 2010. – № 12 (113). – С. 37-43.
11. Запорожец Д.Ю., Курейчик В.В. Гибридный алгоритм решения задач транспортного типа // Известия ЮФУ. Технические науки. – 2013. – № 7 (144). – С. 80-85.
12. Kureichik, V.V., Kravchenko, Y.A., Bova, V.V. Decision support systems for knowledge management // Advances in Intelligent Systems and Computing. – 2015. – Vol. 349. – P. 123-130.
13. Гладков Л.А., Курейчик В.М., Курейчик В.В. Генетические алгоритмы. – М.: Физматлит, 2006. – 320 с.
14. Zaporozhets D.U., Zaruba D.V. and Kureichik V.V. Representation of solutions in genetic VLSI placement algorithms // IEEE East-West Design & Test Symposium – (EWDTS'2014) Kiev, Ukraine, 2014. – P. 1-4.
15. Подлазова А.В. Генетические алгоритмы на примерах задачи раскроя // Проблемы управления. – 2008. – № 2. – С. 57-62.
16. Курейчик В.В., Родзин С.И. О правилах представления решений в эволюционных алгоритмах // Известия ЮФУ. Технические науки. – 2010. – № 7 (108). – С. 13-21.
17. Запорожец Д.Ю., Кудаев А.Ю., Лежебоков А.А. Многоуровневый алгоритм решения задачи параметрической оптимизации на основе биоинспирированных эвристик // Известия Кабардино-Балкарского научного центра РАН. – 2013. – № 4 (54). – С. 21-28.
18. Курейчик В.М. Особенности построения систем поддержки принятия решений // Известия ЮФУ. Технические науки. – 2012. – № 7 (132). – С. 92-98.
19. Курейчик В.М., Кажаров А.А. Использование шаблонных решений в муравьиных алгоритмах // Известия ЮФУ. Технические науки. – 2013. – № 7 (144). – С. 11-17.
20. Gladkov L.A., Kravchenko Y.A., Kureichik V.V. Evolutionary Algorithm for Extremal Subsets Comprehension in Graphs // World Applied Sciences Journal. – 2013. – No. 27 (9). – P. 1212-1217.

## REFERENCES

1. Belousov S.M. Matematicheskaya model' mnogopotchnoy sistemy massovogo obsluzhivaniya, upravlyaemoy planirovshchikom resursov [A mathematical model of a multi-threaded Queuing system, scheduler-managed resources], *Vestnik Novosibirskogo gosudarstvennogo universiteta. Seriya: informatsionnye tekhnologii* [Vestnik of Novosibirsk State University. Series: Information Technologies], 2006, Vol. 4, No. 1, pp. 14-26.
2. Prilutskiy M.Kh., Nefedov D.S., Popov D.V. Raspredelenie resursov v diskretno upravlyaemykh sistemakh [Resource allocation in discrete controlled systems], *Elektronnyy zhurnal "Issledovano v Rossii"* [Electronic journal "Investigated in Russia"], 2002, No. 5, pp. 322-337.

3. Prilutskiy M.Kh., Vyakhirev D.V. mnogostadiynnye zadachi al'ternativnogo raspredeleniya resursov [Multistage tasks alternative allocation of resources], *Vestnik Nizhegorodskogo gosudarstvennogo universiteta. Matematicheskoe modelirovanie i optimal'noe upravlenie* [Vestnik of Nizhny Novgorod state University. Mathematical modeling and optimal control], 2002, Issue 25 (1), pp. 36-43.
4. Yury Kravchenko, Ilona Kursity, Dmitry Zaporozhets. Resource allocation system based on simulation modeling in computer-aided design system. In Conference proceedings, *2016 6th International Conference Cloud System and Data Engineering (Confluence)*. – 14-15 January 2016, Amity University Uttar Pradesh, Noida, India, pp. 395-400.
5. Kravchenko Yu.A. CALS- i CASE-tekhnologii: uchebno-metodicheskoe posobie [CALS - and CASE-technology: a teaching manual]. Taganrog, 2010, 142 p.
6. Kravchenko Yu.A. Cals-imitatsionnye modeli v upravlenii dannymi SAPR [Cals-simulation models in the management of CAD data], *Izvestiya TRTU* [Izvestiya TSURE], 2006, No. 8 (63), pp. 143-146.
7. Eddous M., Stensfild R. Metody prinyatiya resheniy [Decision-making methods]: translation from English, under the editorship of corresponding member of RAS I.I. Eliseevoy. Moscow: Audit, YuNITI, 1997, 590 p.
8. Kureychik V.M., Kureychik V.V. Evolyutsionnye, sinergeticheskie i gomeostaticheskie strategii v iskusstvennom intellekte: sostoyanie i perspektivy [Evolutionary, synergetic and homeostatic strategies in artificial intelligence: state and prospects], *Novosti iskusstvennogo intellekta* [Artificial Intelligence News], 2000, No. 3, pp. 39-67.
9. Kravchenko Yu.A., Kursity I.O. Razrabotka modeli zadachi raspredeleniya resursov v protsessakh SAPR i struktury podsistemy ee resheniya [Development of a model resource allocation problems in the processes of CAD and the structure of the subsystem of its solution], *Trudy Kongressa po intellektual'nym sistemam i informatsionnym tekhnologiyam «IS&IT'15»* [Proceedings of Congress on intelligent systems and information technologies "IS&IT'15"]. Scientific publication in 4 vol. Vol. 3. Moscow: Fizmatlit, 2015.
10. Bova V.V., Kureychik V.V. Integrirovannaya podsistema gibridnogo i kombinirovannogo poiska v zadachakh proektirovaniya i upravleniya [Integrated subsystem of the hybrid and combined search in problems of design and management], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2010, No. 12 (113), pp. 37-43.
11. Zaporozhets D.Yu., Kureychik V.V. Gibridnyy algoritm resheniya zadach transportnogo tipa [A hybrid algorithm for solving problems of transport type], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2013, No. 7 (144), pp. 80-85.
12. Kureichik, V.V., Kravchenko, Y.A., Bova, V.V. Decision support systems for knowledge management, *Advances in Intelligent Systems and Computing*, 2015, Vol. 349, pp. 123-130.
13. Gladkov L.A., Kureychik V.M., Kureychik V.V. Geneticheskie algoritmy [Genetic algorithms]. Moscow: Fizmatlit, 2006, 320 p.
14. Zaporozhets D.U., Zaruba D.V. and Kureichik V.V. Representation of solutions in genetic VLSI placement algorithms, *IEEE East-West Design & Test Symposium – (EWDTS'2014) Kiev, Ukraine, 2014*, pp. 1-4.
15. Podlazova A.V. Geneticheskie algoritmy na primerakh zadachi raskroya [Genetic algorithms on the instances of the problem of cutting], *Problemy upravleniya* [Problems of Management], 2008, No. 2, pp. 57-62.
16. Kureychik V.V., Rodzin S.I. O pravilakh predstavleniya resheniy v evolyutsionnykh algoritmakh [About the rules for the submission of solutions in evolutionary algorithms], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2010, No. 7 (108), pp. 13-21.
17. Zaporozhets D.Yu., Kudaev A.Yu., Lezhebokov A.A. Mnogourovnevyy algoritm resheniya zadachi parametricheskoy optimizatsii na osnove bioinspirirovannykh evristik [A multilevel algorithm for solving the problem of parametric optimization based on bio-inspired heuristics], *Izvestiya Kabardino-Balkarskogo nauchnogo tsentra RAN* [Izvestiya of Kabardino-Balkar scientific centre of the RAS], 2013, No. 4 (54), pp. 21-28.
18. Kureychik V.M. Osobennosti postroeniya sistem podderzhki prinyatiya resheniy [Features of construction of systems of support of acceptance of decisions], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2012, No. 7 (132), pp. 92-98.

19. Kureychik V.M., Kazharov A.A. Ispol'zovanie shablonnykh resheniy v murav'inykh algoritмах [The use of standard solutions in ant colony optimization algorithms], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2013, No. 7 (144), pp. 11-17.
20. Gladkov L.A., Kravchenko Y.A., Kureichik V.V. Evolutionary Algorithm for Extremal Subsets Comprehension in Graphs, *World Applied Sciences Journal*, 2013, No. 27 (9), pp. 1212-1217.

Статью рекомендовал к опубликованию д.т.н., профессор А.Н. Афанасьев.

**Кравченко Юрий Алексеевич** – Южный федеральный университет; e-mail: yakravchenko@sfedu.ru; 347928, г. Таганрог, пер. Некрасовский, 44; тел.: 88634371651; кафедра систем автоматизированного проектирования; доцент.

**Курситыс Илона Олеговна** – e-mail: i.kursitys@mail.ru; кафедра систем автоматизированного проектирования; аспирант.

**Kravchenko Yury Alekseevich** – Southern Federal University; e-mail: yakravchenko@sfedu.ru; 44, Nekrasovskiy lane, Taganrog, 347928, Russia; phone: +78634371651; the department of computer aided design; associate professor.

**Kursitys Iona Olegovna** – e-mail: i.kursitys@mail.ru; the department of computer aided design; postgraduate student.

УДК 002.53:004.89

**А.А. Новиков, Ю.С. Старкова, В.В. Марков, Д.Ю. Кравченко**

### **СЕМАНТИЧЕСКИЙ ПОИСК С ИСПОЛЬЗОВАНИЕМ ГЕНЕТИЧЕСКИХ ОПЕРАТОРОВ\***

*Традиционные механизмы поиска основываются на поиске по ключевым словам, в котором совсем не учитываются семантические связи между различными понятиями. Это приводит к потере релевантных документов из-за неточной формулировки запроса или использования в запросе близких по смыслу слов и понятий. Для решения проблем формулировки запросов пользователей и междисциплинарности понятий предлагается использовать семантический поиск. Предлагаемый в статье подход реализации семантического поиска применим к большим объёмам текстовых данных и основан на использовании генетического алгоритма. В отличие от стандартных подходов к поиску информации, описываемый подход позволяет учитывать семантику взаимосвязей между понятиями, а также корректно обрабатывать междисциплинарные понятия. Благодаря семантической индексации в документах определяются понятия, не присутствующие в исходном запросе пользователя, но являющиеся семантически близкими с понятиями из запроса. Семантическая индексация производится для каждого документа отдельно, что позволяет производить параллельную индексацию сразу по нескольким предметным областям. К моменту завершения формирования онтологического профиля рассматриваемого документа вычисляются все семантические расстояния между парами выделенных понятий. Понятия считаются близкими по смыслу, если их значение семантической близости выше некоторого порогового значения, которое задается в параметрах поиска. Построение онтологического профиля документа является многокритериальной задачей, так как зависит от множества характеристик, поэтому для ее решения можно использовать генетические алгоритмы. Разработанный генетический алгоритм предназначен для более точного распределения весовых коэффициентов и оценки семантической близости понятий.*

*Семантический поиск; информационный поиск; онтология; генетический алгоритм.*

---

\* Работа выполнена при финансовой поддержке РФФИ (проект № 16–07–00703).