

19. Alharbi J.R., Alhalabi W.S. Hybrid Approach for Sentiment Analysis of Twitter Posts Using a Dictionary-based Approach and Fuzzy Logic Methods: Study Case on Cloud Service Providers, *International Journal on Semantic Web and Information Systems (IJSWIS)*, 2020, Vol. 16, No. 1, pp. 116-145.
20. Phan H.T., Nguyen N.T., Cuong T.V., Hwang D. A Method for Detecting and Analyzing the Sentiment of Tweets Containing Fuzzy Sentiment Phrases, *2019 IEEE International Symposium on INnovations in Intelligent SysTems and Applications (INISTA)*, 2019, pp. 1-6.
21. Phan H.T., Tran V.C., Nguyen N.T., Hwang D. Improving the performance of sentiment analysis of tweets containing fuzzy sentiment using the feature ensemble model, *IEEE Access*, 2020, Vol. 8, pp. 14630-14641.
22. Pang B., Lee L. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts, *Proceedings of the 42nd annual meeting on Association for Computational Linguistics*, 2004, pp. 271.
23. Cavalcanti D. C., Ricardo B.C .Prudêncio, Shreyasee S. Pradhan, Jatin Y. Shah, Ricardo S. Pietrobon. Good to be bad? Distinguishing between positive and negative citations in scientific impact, *2011 IEEE 23rd International Conference on Tools with Artificial Intelligence*, 2011, pp. 156-162.
24. Hutto C. J., Gilbert E. Vader: A parsimonious rule-based model for sentiment analysis of social media text, *Eighth international AAAI conference on weblogs and social media*. – 2014.

Статью рекомендовала к опубликованию д.т.н., профессор Л.С. Лисицына.

Герасименко Евгения Михайловна – Южный федеральный университет; e-mail: egerasimenko@sfedu.ru; г. Таганрог, Россия; тел.: 88634371651; кафедра систем автоматизированного проектирования; к.т.н.; доцент.

Стеценко Валентина Витальевна – e-mail: vstecenko@sfedu.ru; кафедра систем автоматизированного проектирования; аспирант.

Gerasimenko Evgeniya Michailovna – Southern Federal University; e-mail: egerasimenko@sfedu.ru; Taganrog, Russia; phone +78634371651; the department of computer aided design; cand. of eng.sc.; associate professor.

Stetsenko Valentina Vitalievna – e-mail: vstecenko@sfedu.ru; the department of computer aided design, postgraduate.

УДК 519.712.2

DOI 10.18522/2311-3103-2022-5-116-128

А.Э. Саак, Л.А. Гладков, Н.В. Гладкова

ИНТЕГРИРОВАННАЯ МОДЕЛЬ РЕШЕНИЯ ЗАДАЧИ ДИСПЕТЧЕРИЗАЦИИ ЗАЯВОК*

Рассматривается задача диспетчеризации. Рассмотрена парадигма организации распределенных вычислений на основе Grid-компьютинга. Приведена классификация систем диспетчеризации задач. Описаны различные подходы к решению задачи диспетчеризации. Приведена модель задачи обслуживания заявок на основе принципов теории систем массового обслуживания. Сформулирована постановка задачи на основе Grid-диспетчеризации. Предложено понятие ресурсного прямоугольника. Определена среда диспетчеризации ресурсных прямоугольников. Предложена модель позволяющая формализовать заявку пользователя на обслуживание понятием ресурсного (незвклидова) прямоугольника. Вместо принципа оптимизации на основе машинного поиска лучшего распределения массива ресурсных прямоугольников, предложен эвристический принцип, что позволило снизить объем необходимых вычислений. Предложенный эвристический алгоритм диспетчеризации позволяет учитывать свойства массива и выполнять оценку качества решений. Построены модели среды спроса в фор-

* Исследование выполнено при поддержке РФФИ в рамках научного проекта № 20-01-00148.

ме единичных кубических граней. Модель кубических граней обобщена на эксперимент кубических слоев. Приведено описание используемой модели спроса. Построена модель среды предложений ресурсов в форме канонической пирамиды и введено понятие канонического эксперимента спроса-предложения на модельные однородные ресурсные элементы. Введено усечение эксперимента спроса-предложения. Предложена гибридная модель на основе сочетания эволюционных принципов поиска и методов нечеткого управления. Для решения задач диспетчеризации предложено использовать эволюционные алгоритмы. Разработана модифицированная методика кодирования решений и новые модификации генетических операторов для решения задач диспетчеризации. Представлена структурная схема алгоритма решения рассматриваемой задачи с учетом использования нечеткого логического контроллера. Выполнено компьютерное моделирование и приведены результаты вычислительных экспериментов. Выявлены особенности предложенного метода, сформулированы его достоинства и недостатки.

Диспетчирование очередей заявок; системы массового обслуживания; эволюционный поиск; нечеткое управление; Grid-компьютинг.

A.E. Saak, L.A. Gladkov, N.V. Gladkova

INTEGRATED MODEL FOR SOLVING THE PROBLEM OF REQUEST DISPATCHING

The paper considers the problem of scheduling. The paradigm of organization of distributed computing based on Grid-computing is considered. The classification of task scheduling systems is given. Various approaches to solving the scheduling problem are described. A model of the task of servicing applications based on the principles of the theory of queuing systems is presented. The task statement is formulated on the basis of Grid-scheduling. The concept of a resource rectangle is proposed. The environment for scheduling resource rectangles is defined. A model is proposed that allows formalizing the user's request for service by the concept of a resource (non-Euclidean) rectangle. Instead of the principle of optimization based on the machine search for the best distribution of the array of resource rectangles, a heuristic principle was proposed, which made it possible to reduce the amount of necessary calculations. The proposed heuristic scheduling algorithm makes it possible to take into account the properties of the array and evaluate the quality of solutions. Models of the demand environment in the form of single cubic faces are constructed. The model of cubic faces is generalized to the experiment of cubic layers. The description of the demand model used is given. A model of the resource supply environment in the form of a canonical pyramid is constructed and the concept of a canonical demand-supply experiment for model homogeneous resource elements is introduced. A truncation of the supply-demand experiment has been introduced. A hybrid model based on a combination of evolutionary search principles and fuzzy control methods is proposed. To solve scheduling problems, it is proposed to use evolutionary algorithms. A modified solution coding technique and new modifications of genetic operators for solving scheduling problems have been developed. A block diagram of the algorithm for solving the problem under consideration is presented, taking into account the use of a fuzzy logic controller. Computer simulation has been performed and the results of computational experiments have been presented. The features of the proposed method are revealed, its advantages and disadvantages are formulated.

Queue scheduling; queuing systems; evolutionary search; fuzzy control; Grid computing.

Введение. Возрастающие потребности пользователей в вычислительных ресурсах привело сначала к разработке и развитию многопроцессорных систем, кластерных систем, метакомпьютинга, а затем, в конце 90-х годов, к появлению Grid-компьютинга (*Grid computing*) [1]. Grid-компьютинг представляет собой одну из парадигм распределённых вычислений, таких, например, как «облачные вычисления» (*cloud computing*) и «вычислительные джунгли» (*jungle computing*) [2–5]. Эффективность функционирования таких систем во многом определяется правильным распределением вычислительных ресурсов, управлением задачами в условиях множественности. При этом на первый план выходит задача управления (диспет-

чирования) заявками пользователей, для обслуживания которых необходимо задействовать несколько процессоров одновременно. Данная задача может быть частично представлена в терминах теории расписаний, но проблему все же необходимо рассматривать гораздо шире. При выполнении машинного поиска наилучшего распределения оказывается, что трудоёмкость задачи, делает невозможным использование таких методов при управлении ресурсами Grid.

Существующая классификация систем диспетчирования задач (*job scheduling*) [6] и управления ресурсами Grid (*Grid resource management systems*) выделяет три базовые архитектуры: централизованная (*centralized*), иерархическая (*hierarchical*) и распределённая (*decentralized, distributed*), различающиеся способом принятия решения об управлении ресурсами и задачами. При использовании централизованной архитектуры управление осуществляется центральным диспетчером, обладающим всей полнотой информации о вычислительных ресурсах и задачах. В Grid-системах различают два вида диспетчирования по способу объединения вычислительных ресурсов для решения задачи: одно- (*single-site scheduling*) и мультисайтное диспетчирование (*multi-site scheduling*). При односайтном диспетчировании задача выполняется в пределах одного сайта, не выходя за рамки параллельной системы. При мультисайтном диспетчировании задача может выполняться одновременно на нескольких сайтах и, соответственно, вне рамок отдельных параллельных систем.

В настоящее время использование Grid-систем в системах диспетчирования задач и управления ресурсами с централизованной архитектурой, поддерживающей мультисайтное выполнение задач, недостаточно разработано. Поэтому актуальным является разработка моделей на основе эвристических алгоритмов распределения ресурсов, учитывающих свойства массива заявок пользователей, с оценкой качества расписания.

Эвристические алгоритмы диспетчирования планарными ресурсами, активно разрабатывавшиеся в конце прошлого и начале нынешнего века, недостаточно результативны. На сегодняшний момент пока не предложен формальный аппарат среды диспетчирования, который должен позволить выявить закономерности на множестве поступающих заявок, ввести их типизацию и предложить эвристические алгоритмы, адаптированные под соответствующие типы с оценкой качества. Разработка таких алгоритмов является актуальной задачей.

Системы обслуживания играют значительную роль в повседневной жизни. Опыт моделирования разных типов дискретных событийных систем свидетельствует о том, что приблизительно 80% математических моделей используют методы теории массового обслуживания [7–9].

Как показывает практика, подобные задачи могут эффективно решаться с использованием методов эволюционных вычислений. Предлагается использовать гибридный подход, основанный на сочетании различных подходов [10]. Гибридизация позволяет достигать «синергетического эффекта», усиливая достоинства встраиваемых методов [11–13].

Модели систем массового обслуживания. При использовании моделей теории массового обслуживания необходимо задать следующие параметры системы [14, 15]:

1) Параметры входного потока заявок поступающих в систему, например, время поступления и общее число поступивших заявок, количество одновременно приходящих заявок. Кроме того, необходимо определить правила поступления заявок: детерминированное расписание или с определенной вероятностью. Число заявок n в любой интервал времени может быть рассчитано на основе закона Пуассона.

2) Принципы постановки в очередь и выбор заявок. Для организации очереди заявок могут использоваться различные эвристические правила [16]. Например, очередь в порядке поступления заявок или правило «последний поступил – первый обслужился». Возможна также организация очереди по заданным параметрам или случайным образом (RANDOM). Возможно введение ограничений по длине или по времени пребывания в очереди.

3) Правила обслуживания заявок. Правила обслуживания могут различаться по времени обслуживания и числу одновременно обслуживаемых заявок. Они определяют условия прекращения обслуживания, выбор заявки для обслуживания и т.д. При этом мы можем оперировать абсолютными или относительными величинами приоритетов. Возможно также задание величины регулирующей глубину прерывания.

4) Параметры выходного потока требований. Структура выходного потока важна для многофазных систем. Распределение заявок по времени в выходном потоке зависит от плотности входного потока и характеристик работы устройств обслуживания.

5) Определение режимов работы. Выбор возможных вариантов режимов работы и условий для их изменения, например, в результате внешнего воздействия или изменения условий работы оборудования, а также возможность блокировки системы при наступлении заданных условий.

Для любых систем массового обслуживания справедлив, так называемый, закон Литтла: при любом распределении времени обслуживания любых двух заявок, любом числе устройств и любых параметрах обслуживания среднее число заявок N определяется интенсивностью их поступления λ и средним временем пребывания в системе T .

Постановка задачи. Выбор технологии для обработки исходной информации, зависит от особенностей решаемых задач, от числа количественных и качественных параметров, описывающих проблему, от уровня ее проработанности. Поэтому необходимо определить условия применимости каждой из рассматриваемых технологий, а также разработать методы и алгоритмы, позволяющие адаптировать их к решению конкретных задач проблемной области.

В парадигме Grid-диспетчирования можно выделить четыре основных этапа [17]:

- 1) выявление ресурсов (*resource discovery*);
- 2) выбор ресурсов (*resource selection*);
- 3) генерация расписания (*schedule generation*);
- 4) выполнение задачи (*job execution*).

Рассмотрим этап генерации расписания алгоритмом диспетчирования (*scheduling algorithm*). При этом алгоритм диспетчирования определяет способ, которым задачи назначаются на ресурсы, а под расписанием понимается назначение задач на ресурсы в определённые периоды времени.

Задача пользователя может быть представлена прямоугольником, со сторонами равными числу процессоров и времени решения. Составление расписания в таком случае может рассматриваться как задача упаковка прямоугольников в полосу (множество полос) с соблюдением условия параллельности сторон прямоугольников соответствующим сторонам полосы, неналожением их друг на друга, отсутствием пересечений сторон полос и минимизацией занятой части полос.

При представлении заявки пользователя для обслуживания Grid-системой введём понятие «ресурсный прямоугольник». Ресурсным прямоугольником назовём фигуру (рис. 1), у которой горизонтальное и вертикальное измерения принимаются равными числу единиц ресурса времени и процессоров, требуемому для обработки заявки, соответственно.

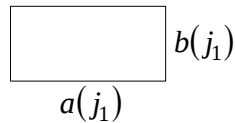


Рис. 1. Ресурсный прямоугольник

Несмотря на визуальное сходство ресурсного прямоугольника с геометрическим, сами они принципиально различаются. Звенья-измерения геометрического прямоугольника имеют размерность единиц длины, одного рода – однородны. В тоже время звенья-измерения ресурсного прямоугольника имеют размерность единиц ресурса одного рода (процессоры) и другого рода (время) – разнородны.

В процессе использования ресурсных прямоугольников могут применяться операции сложения, умножения, дифференцирования и динамического интегрирования, ориентированные на диспетчирование множественного компьютерного обслуживания [18].

Отметим, что наряду со спецификой согласованности, антисогласованности и рассогласованности в поведении измерений ресурсных прямоугольников у трёх квадратичных типизаций есть и важные общие черты. Прежде всего у каждого типа, как будет показано далее, выделяется центральный элемент: максимальный – у кругового; наиболее симметричный – у гиперболического и средний (по высотам, по среднересурсной величине, по протяжённости суммы оснований предшествующих граней и высоты среднего элемента) – у параболического квадратичного типа.

Классический принцип оптимизации состоит в определении решения задач 2SP, 2BP, SP, RP (RPAMP) локальными динамическими переходами из некоторого начального положения массива заявок. Принцип эвристики перераспределения массива заявок в первом ресурсном квадранте состоит в решении задачи локализации на основе программного задания алгоритма диспетчирования протяжённой полиэдрали ресурсных прямоугольников с одновременным контролем качества посредством эвристической меры. Предлагаемые полиномиальные алгоритмы диспетчирования адаптированы под соответствующий квадратичный тип массива заявок.

Таким образом, классической динамической оптимизации распределения планарных ресурсов противопоставляется эвристическая оптимизация на основе программного полиномиального алгоритма локализации первоначального протяжённого массива спроса в ресурсную оболочку с оценкой качества оболочки посредством эвристической меры.

В системе реализуется правило постановки заявок в очередь в порядке их поступления (*FIFO*), а также правило организации очереди с использованием заявок с абсолютным приоритетом. Время обслуживания всех типов заявок любым устройством имеет произвольное распределение. Схема взаимодействия отдельных узлов сети задана полносвязанным графом G . Маршрут прохождения заявок в сети массового обслуживания задается матрицей:

$$P = \|p_{im,i+1}\|,$$

где $p_{im,i+1}$ – Звенья-измерения вероятность того, что заявка класса m , после обслуживания i -м узле, перейдет в узел $i+1$.

Маршрут прохождения заявок каждого типа определяется с учетом необходимости посетить с некоторой вероятностью заданное число устройств, значения вероятностей посещения этих устройств обусловлены типом заявки.

Необходимо решить задачу построения графика работы обслуживающих устройств для заявок различных классов, позволяющего максимально повысить эффективность функционирования сети. При этом необходимо учитывать возможные временные затраты.

Модель спроса. Единичный спрос индивидуального пользователя однородными вычислительными ресурсами рассматривается в форме двоичности $0 \vee 1$ негативного исхода величины заявки $j_1 = 0$ и позитивного, реального, единичного исхода той же величины $j_1 = 1$ с единичными мощностными отсчётами $Y(0) = 1$, $Y(1) = 1$ в каждом значении составности (рис. 2).

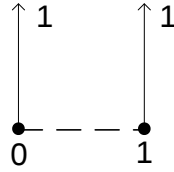


Рис. 2. Единичный спрос однородными вычислительными ресурсами

При комбинировании, умножении данных позитивностей по правилам умножения алгебры случайных событий негативные исходы могут быть опущены. Поэтому, наряду с двоичностью, принимается комбинационная модель единичного ресурсного требования в виде унодного элемента $\delta_{j_1}^{(1)}(j_1')$ и единичного звена-измерения 1-го рода на числовой Z^1 -оси (рис. 3).

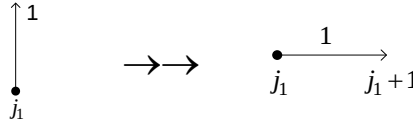


Рис. 3. Унодный элемент и единичное звено-измерение

Используем композицию « k » двоичностей $0 \vee 1$ действием k -кратной круговой первообразной в производящей форме $\sum_{j_1=0}^k C_k^{(j_1)} S_1^{j_1} W_1^0 = (1 + W_1)^k$ на начальный отсчёт W_1^0 гармонического распределения $W_1^{j_1}$. В области распределений переходом $W_1 \rightarrow 1$ получаем символический бином $(1+1)^k = \sum_{j_1=0}^k C_k^{(j_1)} \rightarrow \{C_k^{(j_1)}\}_{j_1=0}^k$ и сочетательную функцию, определяющую модельный параллельный спрос « k » пользователей на единичные ресурсы 1-го рода. Данной круговой модели параллельного спроса на единичные ресурсные элементы сопоставляем гиперболическую модель того же спроса в виде действия k -кратной гиперболической первообразной в производящей форме на предыдущее распределение гармонических образцов единичных унодов $\sum_{j_1=0}^k (-1)^{j_1} C_k^{(j_1)} S_1^{j_1} W_1^0 = (1 - W_1)^k$.

Модель предложений ресурсов. Пирамидальное распределение

$$\frac{1^k}{k!}, \frac{2^k}{k!}, \dots, \frac{j_1^k}{k!}, \dots, \frac{k^k}{k!}$$

индуцирует аддитивный базис эксперимента координатных пирамид (рис. 4).

Данная пирамида асимптотически исчерпывается указанным аддитивным базисом координатных пирамид. Множество X образует объемлющая каноническая пирамида в R^k с длиной смежных ребер « k » объема $k^k/k!$, аддитивный базис $\{X_{i'}\}_{i'=1}^k$ даёт распределение координатных пирамид, негативный исход X_0 выражается началом координат. Такова аддитивная форма эксперимента координатных пирамид в R^k .

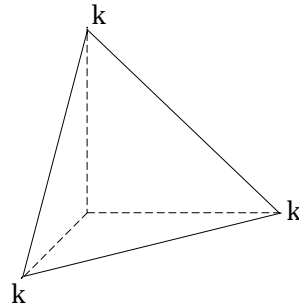


Рис. 4. Координатная пирамида

Модель нечеткого управления. Практика решения различных NP-полных задач оптимизации показывает, что достаточно эффективным инструментом решения подобных задач являются эволюционные алгоритмы. Использование эволюционных алгоритмов традиционно требует решения следующих задач [19, 20]:

- ◆ определение методики кодирования множества возможных решений;
- ◆ подбор управляющих параметров эволюционного алгоритма;
- ◆ выбор и модификация генетических операторов.

Для решения задач обслуживания заявок можно использовать хромосомы фиксированной длины N , состоящие из набора M векторов, каждый i -й вектор представляет собой последовательность обслуживания заявок на данном узле. Гены упорядочены по времени, так что каждый ген жестко привязан к определенному моменту времени. Пример хромосомы показан на рис. 5.

Узел 1	s_{1j}	$t_{1j, j+1}$	s_{1j+1}	...	s_{1N}
Узел 2	s_{2j}	$t_{2j, j+1}$	s_{2j+1}	...	s_{2N}
.....	...	...	...	...	...
Узел i	s_{ij}	$t_{ij, j+1}$	s_{ij+1}	...	s_{iN}
.....	...	...	...	...	...
Узел M	s_{Mj}	$t_{Mj, j+1}$	s_{Mj+1}	...	s_{MN}

Рис. 5. Пример хромосомы

Каждая строка – это последовательность обслуживания заявок различного типа. Например, строка номер 2 задает следующий порядок обслуживания заявок. Сначала обрабатывается заявка j -го класса, затем после переключения устройства, обрабатываются заявки $(j+1)$ -го класса и так далее. Продолжительность переключения между различными классами заявок задается матрицей переключений.

Основными генетическими операторами являются селекция, кроссинговер и мутация. Оператор селекции выполняет в генетическом алгоритме две важные задачи. Во-первых, выбор решений из текущей популяции, к которым будут при-

меняться операторы кроссинговера и мутации. После выполнения всех генетических операторов на текущей итерации алгоритма, образуется новая расширенная популяция. Для дальнейшей работы генетического алгоритма необходимо восстановить исходный размер популяции. Отбор решений из расширенной популяции, которые перейдут в следующее поколение является второй задачей селекции.

Также будем использовать модифицированный одноточечный оператор кроссинговера и модифицированный оператор мутации обмена.

Подбор и управления значениями управляющих параметров алгоритма осуществляется с использованием блока нечеткого управления. Управляющими параметрами в данном случае являются вероятности выполнения генетических операторов [21].

Структурная схема алгоритма решения рассматриваемой задачи с учетом использования нечеткого логического контроллера может быть представлена следующим образом (рис. 6).

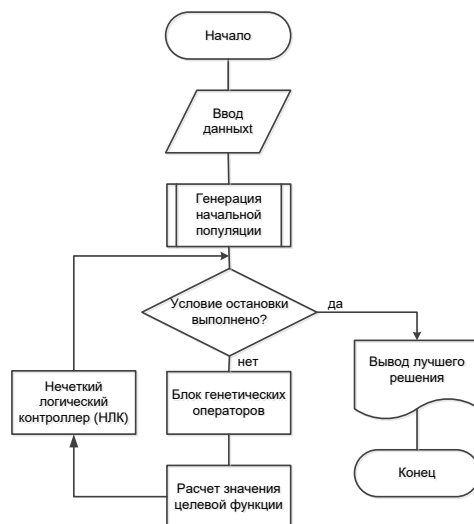


Рис. 6. Структура модели нечеткого управления

Знания, используемые в процессе функционирования модуля нечеткого управления, записываются в виде нечетких правил или нечетких множеств с функцией принадлежности, заданной выражением [22–24]:

$$\mu_{R^{(k)}}(x, y) = \mu_{A^k \rightarrow B^k}(x, y).$$

Конкретная форма функции принадлежности нечеткого множества зависит от применяемой Т-нормы, определения нечеткой импликации и от способа задания декартова произведения нечетких множеств.

Для оценки качества получаемых решений и процесса эволюции в целом используются параметры характеризующие динамику изменения среднего и лучшего значений ЦФ, а также разнообразие популяции.

На основании оценки динамики контрольных значений изменяются значения вероятностей выполнения операторов кроссинговера и мутации

$$Pc(t) = Pc(t - 1) + \Delta Pc(t), \quad Pm(t) = Pm(t - 1) + \Delta Pm(t).$$

Результаты компьютерного моделирования. При наличии достаточного количества достоинств, данное кодирование имеет и серьезные недостатки, в том числе:

Исследование проводилось по двенадцати точкам, выбранным экспертно. Каждая точка эксперимента, для которых строится производственное расписание, определяется вектором исходных данных, координатами которого являются: общее количество заданий N , общее количество линий M , количество спаренных линий $LD = \sum_{j_1=1}^M \sum_{j_2=(M-j_1-1)}^M ld_{j_1 j_2}$, горизонт планирования D в часах, количество оборудования RQ .

Целью проведения экспериментов было установление характера зависимостей временных затрат на поиск решения от размерности задачи (число работ N , число линий M). Под сходимостью понимается достижение такого значения целевой функции D , при котором за последующие δ циклов работы алгоритма изменение ΔD составит не более $\Delta\%$ от предыдущего значения

$$\frac{|D_{i-\delta} - D_i|}{D_{i-\delta}} * 100\% \leq \Delta,$$

где D_i и $D_{i-\delta}$ – значение целевой функции на i -м и $(i-\delta)$ -м шаге соответственно, а i – число итераций алгоритма.

Ниже представлена таблица исходных данных (таблица 1).

Таблица 1

Данные для оценки сходимости алгоритмов

№	N	M/LD	D	RQ
1	5	3/2	72	1
2	10	3/2	96	1
3	20	6/2	96	2
4	50	12/4	120	3
5	70	16/6	120	3
6	100	24/8	120	4
7	150	26/8	144	5
8	200	30/10	168	6
9	250	34/10	192	6
10	300	40/12	240	7
11	400	50/16	240	8
12	500	54/18	240	8

Также в ходе проведения экспериментов решалась задача получения количественной оценки улучшения качества решения. Улучшение качества оценивалось относительно значения целевой функции начального решения. В целях наглядности демонстрации в эксперимент был включен алгоритм случайного поиска решения. При проведении данного исследования для каждого алгоритма были взяты одни и те же исходные данные: количество заданий $N = 100$, количество линий, из них спаренных $M/LG = 24/8$, горизонт планирования $D = 120$; количество оснастки $RQ = 4$.

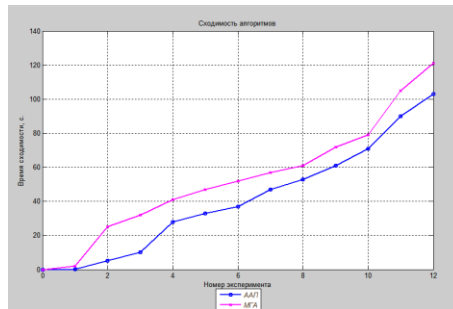


Рис. 7. Оценка сходимости результатов

Для предложенного модифицированного эволюционного алгоритма проводился анализ эффективности работы при 1000 итерациях, а работа алгоритма случайного поиска оценивалась на множестве 100000 случайно найденных решений. При сравнении результатов работы алгоритмов использовались следующие значения параметров (см. табл. 2).

Таблица 2

Параметры модифицированного эволюционного алгоритма

Наименование параметра	Значение
Размер популяции.	100
Вероятность скрещивания	0.9
Вероятность мутации	0.1
Схема миграции	«лучшие» – «худшие»
Период миграции, пок.	5
Доля мигрирующих особей, %	10
Количество точек скрещивания, %	0.0005
Количество точек мутации, %	0.001

Результаты сравнения эффективности работы различных алгоритмов приведены на рис. 8.

Заключение. На основании полученных результатов можно утверждать, что временная сложность растет линейно при увеличении размерности задачи. На рисунке 7 отражена зависимость времени сходимости решения от общего числа производственных линий.

Результаты экспериментов подтвердили общее предположение о целесообразности использования приближенного алгоритма для получения приемлемого (квазиоптимального) решения. Полученные зависимости подтверждают предположение о полиномиальной временной сложности предложенных алгоритмов.

Анализируя полученные зависимости, можно отметить тот факт, что увеличение числа циклов работы алгоритма случайного поиска не приводит значимому улучшению качества получаемых результатов. Что касается времени работы, то наиболее быстрым является модифицированный эволюционный алгоритм. Это происходит при наличии качественных правил или предметных эвристик, позволяющих разрешать нарушения ограничений. Модифицированный эволюционный алгоритм позволяет получать набор решений приемлемого качества за полиномиальное время.

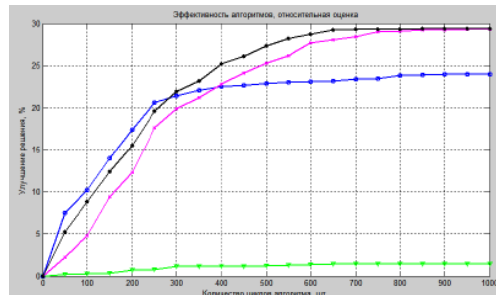


Рис. 8. Сравнение эффективности алгоритмов

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Foster I., Kesselman C. The grid: blueprint for a new computing infrastructure. – Morgan Kaufmann Publishers Inc., USA, 1998.
2. Cafaro M., Aloisio G. (eds.). Grids, clouds and virtualization, Computer Communications and Networks. – Springer London, 2011.
3. Yaqin Liu, Chubo Liu, Jing Liu, Yikun Hu, Kenli Li, Keqin Li Mobility-Aware and Code-Oriented Partitioning Computation Offloading in Multi-Access Edge Computing // Journal of Grid Computing. – 2022. – Vol. 20, Article number 11. – <https://doi.org/10.1007/s10723-022-09599-x>.
4. Jacob B., Brown M., Fukui K., Trivedi N. Introduction to Grid Computing. – ibm.com/redbooks 2005.
5. Kahanwal B., Singh T. The distributed computing paradigms: p2p, grid, cluster, cloud, and jungle // International Journal of Latest Research in Science and Technology. – 2012. – Vol. 1, Issue 2. – P. 183-187.
6. Magoulès F., Nguyen T., Yu L. Grid resource management: toward virtual and services compliant grid computing, Numerical analysis and scientific computing. – CRC Press, UK, 2009.
7. Жожикашвили В.В., Вишневский В.М. Сети массового обслуживания. Теория и применение к сетям ЭВМ. – М.: Радио и связь, 1998.
8. Pinedo M. Scheduling: Theory, Algorithms and Systems. – 3rd ed. – Springer Verlag, New York, 2008.
9. Conway R.M., Maxwell W.L., Miller L.W. Theory of Scheduling. – 2nd ed. – Dover Publications, Mineola NY, 2004.
10. Ярушкіна Н.Г. Основы теории нечетких и гибридных систем. – М.: Финансы и статистика, 2004.
11. Батыршин И.З., Недосекин А.О. и др. Нечеткие гибридные системы. Теория и практика. – М.: Физматлит, 2007.
12. Kar A.K. Bio Inspired Computing – A Review of Algorithms and Scope of Applications // Expert Systems with Applications. – 2016. – Vol. 59. – P. 20-32.
13. Nature-Inspired Computation and Swarm Intelligence: Algorithms, Theory and Applications. – Academic Press, 2020. – ISBN 978-0-12-819714-1. – <https://doi.org/10.1016/C2019-0-00628-0>.
14. Gross D., Shortle J.F., Thompson J.M., Harris C.M. Solutions Manual to Accompany Fundamentals of Queueing Theory. 4th Edition. – John Wiley & Sons Inc., 2008.
15. Bhat Narayan U. An Introduction to Queueing Theory: Modeling and Analysis in Applications (Statistics for Industry and Technology). – Birkhäuser, 2008.
16. Гладков Л.А., Гладкова Н.В., Громов С.А. Гибридная модель решения задач оперативного производственного планирования // Известия ЮФУ. Технические науки. – 2018. – № 4 (165). – С. 102-114.
17. Li M., Baker M. The grid: core technologies. – John Wiley & Sons Ltd, England, 2005.
18. Саак А.Э. Полиномиальные алгоритмы распределения ресурсов в Grid-системах на основе квадратичной типизации массивов заявок // Информационные технологии. – 2013. – № 7.
19. Гладков Л.А., Курейчик В.В., Курейчик В.М. Генетические алгоритмы. – М.: Физматлит, 2010.
20. Емельянов В.В., Курейчик В.В., Курейчик В.М. Теория и практика эволюционного моделирования. – М.: Физматлит, 2003.

21. Гладков Л.А., Гладкова Н.В., Громов С.А. Гибридный алгоритм решения задач оперативного планирования производственного процесса // Известия ЮФУ. Технические науки. – 2017. – № 9 (194). – С. 112-123.
22. Herrera F., Lozano M. Fuzzy Adaptive Genetic Algorithms: design, taxonomy, and future directions // J. Soft Computing. – 2003. – P. 545-562.
23. Michael A., Takagi H. Dynamic control of genetic algorithms using fuzzy logic techniques // Proceedings of the Fifth International Conference on Genetic Algorithms. – Morgan Kaufmann, 1993. – P. 76-83
24. Lee M.A., Takagi H. Integrating design stages of fuzzy systems using genetic algorithms // Proceedings of the 2nd IEEE International Conference on Fuzzy System. – 1993. – P. 612-617.

REFERENCES

1. Foster I., Kesselman C. The grid: blueprint for a new computing infrastructure. Morgan Kaufmann Publishers Inc., USA, 1998.
2. Cafaro M., Aloisio G. (eds.). Grids, clouds and virtualization, Computer Communications and Networks. Springer London, 2011.
3. Yaqin Liu, Chubo Liu, Jing Liu, Yikun Hu, Kenli Li, Keqin Li Mobility-Aware and Code-Oriented Partitioning Computation Offloading in Multi-Access Edge Computing, *Journal of Grid Computing*, 2022, Vol. 20, Article number 11. Available at: <https://doi.org/10.1007/s10723-022-09599-x>.
4. Jacob B., Brown M., Fukui K., Trivedi N. Introduction to Grid Computing. ibm.com/redbooks 2005.
5. Kahanwal B., Singh T. The distributed computing paradigms: p2p, grid, cluster, cloud, and jungle, *International Journal of Latest Research in Science and Technology*, 2012, Vol. 1, Issue 2, pp. 183-187.
6. Magoulès F., Nguyen T., Yu L. Grid resource management: toward virtual and services compliant grid computing, Numerical analysis and scientific computing. CRC Press, UK, 2009.
7. Zhzhikashvili V.V., Vishnevskiy V.M. Seti massovogo obsluzhivaniya. Teoriya i primeneniye k setyam EVM [Queuing networks. Theory and application to computer networks]. Moscow: Radio i svyaz', 1998.
8. Pinedo M. Scheduling: Theory, Algorithms and Systems. 3rd ed. Springer Verlag, New York, 2008.
9. Conway R.M., Maxwell W.L., Miller L.W. Theory of Scheduling. 2nd ed. Dover Publications, Mineola NY, 2004.
10. Yarushkina N.G. Osnovy teorii nechetkikh i gibridnykh system [Fundamentals of the theory of fuzzy and hybrid systems]. Moscow: Finansy i statistika, 2004.
11. Batyrshin I.Z., Nedosekin A.O. i dr. Nечеткие гибридные системы. Teoriya i praktika [Fuzzy hybrid systems. Theory and practice]. Moscow: Fizmatlit, 2007.
12. Kar A.K. Bio Inspired Computing – A Review of Algorithms and Scope of Applications, *Expert Systems with Applications*, 2016, Vol. 59, pp. 20-32.
13. Nature-Inspired Computation and Swarm Intelligence: Algorithms, Theory and Applications. Academic Press, 2020. ISBN 978-0-12-819714-1. Available at: <https://doi.org/10.1016/C2019-0-00628-0>.
14. Gross D., Shortle J.F., Thompson J.M., Harris C.M. Solutions Manual to Accompany Fundamentals of Queueing Theory. 4th Edition. John Wiley & Sons Inc., 2008.
15. Bhat Narayan U. An Introduction to Queueing Theory: Modeling and Analysis in Applications (Statistics for Industry and Technology). Birkhäuser, 2008.
16. Gladkov L.A., Gladkova N.V., Gromov S.A. Gibrinaya model' resheniya zadach operativnogo proizvodstvennogo planirovaniya [Hybrid model for solving operational production planning tasks], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2018, No. 4 (165), pp. 102-114.
17. Li M., Baker M. The grid: core technologies. John Wiley & Sons Ltd, England, 2005.
18. Saak A.E. Polinomial'nye algoritmy raspredeleniya resursov v Grid-sistemakh na osnove kvadrachnoy tipizatsii massivov zayavok [Polynomial algorithms of resource allocation in Grid systems based on quadratic typing of arrays of applications], *Informatsionnye tekhnologii* [Information Technologies], 2013, No. 7.

19. Gladkov L.A., Kureychik V.V., Kureychik V.M. Geneticheskie algoritmy [Kureychik V.M. Genetic algorithms]. Moscow: Fizmatlit, 2010.
20. Emel'yanov V.V., Kureychik V.V., Kureychik V.M. Teoriya i praktika evolyutsionnogo modelirovaniya [Theory and practice of evolutionary modeling]. Moscow: Fizmatlit, 2003.
21. Gladkov L.A., Gladkova N.V., Gromov S.A. Gibridnyy algoritm resheniya zadach operativnogo planirovaniya proizvodstvennogo protsessa [Hybrid algorithm for solving problems of operational planning of the production process], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2017, No. 9 (194), pp. 112-123.
22. Herrera F., Lozano M. Fuzzy Adaptive Genetic Algorithms: design, taxonomy, and future directions, *J. Soft Computing*, 2003, pp. 545-562.
23. Michael A., Takagi H. Dynamic control of genetic algorithms using fuzzy logic techniques, *Proceedings of the Fifth International Conference on Genetic Algorithms*. Morgan Kaufmann, 1993, pp. 76-83
24. Lee M.A., Takagi H. Integrating design stages of fuzzy systems using genetic algorithms, *Proceedings of the 2nd IEEE International Conference on Fuzzy System*, 1993, pp. 612-617.

Статью рекомендовала к опубликованию д.т.н., профессор Л.С. Лисицына.

Саак Андрей Эрнестович – Южный федеральный университет; e-mail: aesaak@sfedu.ru; г. Таганрог, Россия; кафедра государственного и муниципального управления; зав. кафедрой.

Гладков Леонид Анатольевич – e-mail: lagladkov@sfedu.ru; тел.: 88634371625; кафедра САПР; к.т.н.; доцент.

Гладкова Надежда Викторовна – e-mail: nvgladkova@sfedu.ru; тел.: 88634393260; кафедра САПР; старший преподаватель.

Saak Andrey Ernestovich – Southern Federal University; e-mail: aesaak@sfedu.ru; Taganrog, Russia; the department of state and municipal administration; head of department.

Gladkov Leonid Anatol'evich – e-mail: lagladkov@sfedu.ru; phone: +78634371625; the department of CAD; cand. of eng. sc.; associate professor.

Gladkova Nadezda Viktorovna – e-mail: nvgladkova@sfedu.ru; phone: +788634393260; the department of CAD; senior teacher.

УДК 004

DOI 10.18522/2311-3103-2022-5-128-141

А.Н. Беликов, С.А. Кучеров, В.С. Лапшин, Ю.Ю. Липко, А.С. Свиридов

АНАЛИЗ ИСПОЛЬЗОВАНИЯ МОДЕЛЕЙ КОММУНИКАЦИИ В ЖИЗНЕННОМ ЦИКЛЕ РАЗРАБОТКИ ИНФОРМАЦИОННЫХ СИСТЕМ

Проводится анализ существующих моделей коммуникации и их применимости к задачам организации эффективной коммуникации в процессе разработки информационных систем. Рассматривается использование моделей коммуникации в сочетании с различными, наиболее часто применяющимися моделями жизненного цикла информационных систем – водопадной, спиральной, SCRUM. Анализируются достоинства и недостатки таких сочетаний и их причины на примере модели Т. Ньюкомба. Основным элементом коммуникации является обеспечение понимания между двумя участниками относительно объекта, который возникает в проекте на очередном шаге его выполнения. Представлена систематизация знаний и информации с учетом субъектов и форм ее представления. Проведенный анализ показывает, что модели, реализуя заложенные в них идеи, в различной степени соответствуют разным моделям жизненного цикла, однако ни одна из моделей коммуникации в полной мере не позволяет решить проблемы коммуникации при разработке информационных систем. Выполненные исследования показывают, что ни одно из сочетаний моделей не позволяет эффективно организовать коммуникацию между участниками в процессе разработки. Необходимо кардинально пересматривать модели и механизмы таким обра-