

**Ковалева Карина Сергеевна** – e-mail: kchicherina@sfedu.ru; кафедра математического обеспечения и применения ЭВМ; ассистент.

**Родзина Ольга Николаевна** – e-mail: orodzina@sfedu.ru; кафедра математического обеспечения и применения ЭВМ; старший преподаватель.

**Rodzin Sergey Ivanovich** – Southern Federal University; e-mail: srodzin@sfedu.ru; 44, Nekrasovskiy, Taganrog, 347928, Russia; phone: +78634371673; the department of software engineering; professor.

**Danilov Igor Gennad'evich** – e-mail: igdanilov@sfedu.ru; the department of software engineering; cand. of eng. sc.; associate professor.

**Kovaleva Karina Sergeevna** – e-mail: kchicherina@sfedu.ru; the department of software engineering; assistant.

**Rodzina Olga Nikolaevna** – e-mail: orodzina@sfedu.ru; the department of software engineering; senior teacher.

УДК 004.8

DOI 10.23683/2311-3103-2019-4-59-68

**Ю.О. Чернышев, Н.Н. Венцов**

**АЛГОРИТМ ОПРЕДЕЛЕНИЯ ПАРАМЕТРОВ БИОИНСПИРИРОВАННОГО ПОИСКА НА ОСНОВЕ ЗАИМСТВОВАНИЯ РЕЗУЛЬТАТОВ РЕШЕННЫХ РАНЕЕ ЗАДАЧ\***

*Рассмотрена проблема выбора наиболее эффективной аппаратной архитектуры при реализации генетических алгоритмов, решающих NP-полные задачи. Особенность подобного рода задач – недоказанность наличия алгоритмов, способных находить точное решение за полиномиальное время. Используемые для решения подобных задач эволюционные алгоритмы являются стохастическими по своей сути, что затрудняет эффективное планирование вычислительного процесса. Цель работы состоит в нахождении способов сокращения времени поиска за счет анализа результатов, полученных при решении аналогичных задач, рассмотренных ранее, и выбора эффективной архитектуры, реализующей данный эволюционный алгоритм. Актуальность работы обусловлена экспоненциальным ростом размерностей решаемых оптимизационных задач. Научная новизна заключается в использовании знаний о решенных ранее задачах для оптимизации параметров генетического алгоритма, решающего текущую задачу, и выбора эффективной аппаратной архитектуры, на которой будет реализован данный алгоритм. Точная граница, как функция от числа обрабатываемых стохастическим алгоритмом особей, наиболее приемлемой архитектуры может быть определена только приближенно. По этой причине, граница наиболее эффективной аппаратной архитектуры задается в виде нечеткого множества. Постановка задачи состоит в следующем: ускорить процесс решения текущей задачи, используя априорные настройки параметров генетического алгоритма, основываясь на анализе результатов решений, рассмотренных ранее задач, и выборе эффективной аппаратной архитектуры. Предложен подход к выбору эффективной аппаратной архитектуры, состоящий из трех этапов: анализа результатов решения рассмотренных ранее задач, определения количества особей, которые необходимо сгенерировать для получения приемлемого решения эволюционным алгоритмом, выбора аппаратной архитектуры, на которой будет реализован эволюционный алгоритм.*

*Оптимизация вычислений; нечеткие системы; выбор аппаратной архитектуры; перенос знаний; адаптация; контекст.*

---

\* Исследование выполнено при финансовой поддержке РФФИ в рамках научных проектов № 19-01-00357 и 18-01-00314.

Yu.O. Chernyshev, N.N. Ventsov

## ALGORITHM FOR DETERMINING THE PARAMETERS OF BIO-INSPIRED SEARCH BASED ON BORROWING THE RESULTS OF PREVIOUSLY SOLVED PROBLEMS

*The problem of selecting the most effective hardware architecture in the implementation of genetic algorithms that solve NP-complete problems is considered. The peculiarity of this kind of problems is the unproven existence of algorithms capable of finding an exact solution in polynomial time. Evolutionary algorithms used to solve such problems are stochastic in nature, which makes it difficult to plan the computational process effectively. The aim of the work is to find ways to reduce the search time by analyzing the results obtained in solving similar problems considered earlier and choosing an effective architecture that implements this evolutionary algorithm. The relevance of the work is due to the exponential growth of the dimensions of the optimization problems to be solved. The scientific novelty lies in the use of knowledge about previously solved problems to optimize the parameters of the genetic algorithm that solves the current problem, and the choice of an effective hardware architecture on which this algorithm will be implemented. The exact boundary, as a function of the number of individuals processed by the stochastic algorithm, of the most appropriate architecture cannot be determined precisely. For this reason, the boundary of the most efficient hardware architecture is defined as a fuzzy set. The problem statement is: to accelerate the process of solving the current task, using the a priori parameter settings of genetic algorithm based on the analysis of the results of decisions previously discussed tasks, and choosing efficient hardware architecture. An approach to the selection of effective hardware architecture is proposed, consisting of three stages: analysis of the results of solving the previously considered problems, determining the number of individuals that must be generated to obtain an acceptable solution by the evolutionary algorithm, selection of hardware architecture on which the evolutionary algorithm will be implemented.*

*Computing optimization; fuzzy systems; selection of hardware architecture; knowledge transfer; adaptation; context.*

**Введение.** Сложность современных задач принятия решений, во многих случаях, не позволяет получать приемлемые результаты за один запуск алгоритма. Данная проблема, как правило, обусловлена не только большой областью поиска, но и сложностью формализации моделируемых процессов и систем. Сжатие пространства поиска осуществляется за счет декомпозиции рассматриваемой задачи, а сложность формализации частично может быть нивелирована за счет использования нечетких составляющих при создании математической модели, т.е её фаззификации. Определить оптимальные параметры декомпозиции и фаззификации на этапе планирования вычислительного процесса, как правило, невозможно. Поэтому, применяют итерационное решение поставленной задачи, что позволяет не только адаптировать имеющуюся математическую модель, но и корректировать параметры алгоритмов поиска в рамках статической модели. Известно, что при анализе моделей, содержащих неявно определенные параметры, широко используются эвристические методы, основанные на итерационных процессах [1]. Для решения особо сложных задач разрабатываются эволюционные алгоритмы, оперирующие более чем миллиардом переменных [2]. Тематикой снижения сложности процедур многокритериальных биоэвристик для многомерных задач занимается значительное число исследовательских групп [3].

**Постановка задачи.** До настоящего времени научный интерес представляет разработка алгоритмов, решающих NP-полные задачи. Ключевая особенность подобного рода задач – недоказанность наличия алгоритмов, способных находить точное решение за полиномиальное время [4]. Поэтому, для решения многих NP-полных оптимизационных задач эффективно используются биоинспирированные алгоритмы (генетические алгоритмы, роевые методы и др.) [5–8]. Подобные

подходы оперируют с популяциями решений, возвращают множество субоптимальных результатов, но, в тоже время, крайне зависимы от качества начальных решений. При этом, главное, трудно формализуемое, требование к начальной популяции состоит в возможности получения приемлемых решений из особой стартовой популяции за приемлемое время. Ещё одной особенностью этих алгоритмов является случайность процесса поиска, что затрудняет планирование вычислительного процесса в рамках декомпозиции задачи [5, 9–11]. В настоящее время активно ведутся исследования, связанные с использованием имеющейся аппаратной архитектуры для эффективного решения как можно большего количества задач [12]. Возможность решения задачи как с помощью центрального процессора, так и графического, позволяет рассматривать большинство ЭВМ как разнородные вычислительные системы. Известно, что в случае гетерогенной вычислительной архитектуры актуальным является распределение решаемых задач по разнородным процессорам, с учетом предполагаемых параметров алгоритмов, решающих данные задачи. При многократном решении задач, построенных на схожих математических моделях, возникает актуальная проблема использования полученных ранее результатов при решении очередной подобной задачи [13–16]. Алгоритмы, осуществляющие заимствование знаний, при решении задач анализа и принятия решений, в настоящее время активно разрабатываются [13, 15], а их математический аппарат постоянно совершенствуется [14].

**Предлагаемый подход.** Известно, что существует определенный рубеж, после превышения которого дальнейшее увеличение параметров генетического алгоритма, таких как число особей или количество популяций, становится неэффективным. При показателях, находящихся ниже значения данного рубежа, результаты работы алгоритма значительно ухудшаются. Таким образом, корректный выбор значений числа особей в популяции и числа поколений генетического алгоритма оптимизации многоэкстремальных функций значительно повышает его эффективность [17]. В данном случае под эффективностью понимается способность генетического алгоритма получить приемлемое решение, обработав указанное количество особей. Генетический алгоритм, обрабатывающий заданное число особей на различных аппаратных архитектурах, будет выполнен за разное время. Поэтому, возникает проблема выбора эффективной аппаратной архитектуры, на которой эволюционный алгоритм, обрабатывающий заданное число особей, будет реализован за наименьшее время.

В процессе первой итерации, на основе информации о структуре пространства поиска, а также информации о решенных ранее задачах, предполагается определить приближенное количество особей, которые необходимо сгенерировать, для получения приемлемого решения, а также другие параметры поиска, такие как вероятность применения операторов мутации и селекции. Если для определения указанных параметров информации недостаточно, первая итерация осуществляется на основе случайно определенных параметров до достижения приемлемого решения. В соответствии с выбранными параметрами, определяется эффективная аппаратная архитектура, на которой будет реализован эволюционный алгоритм. Проблема выбора эффективной аппаратной архитектуры для реализации генетического алгоритма состоит в том, что априорно определить границу эффективности каждой аппаратной архитектуры достаточно сложно [10].

Нечеткие формулировки в условиях оптимизационной задачи, во многих случаях, подразумевают недостаток объективных данных о моделируемой системе, и, как правило, обуславливают последующие верификации получаемых решений. Важная положительная характеристика генетических алгоритмов

заключается в том, что они хорошо работают в задачах, для которых необходимо выстраивать сложный ландшафт проектирования – тех, где целевая функция прерывается, имеет неточности, изменяется с течением времени или имеет много локальных оптимумов [4].

Для более точного прогноза количества особей, которые необходимо сгенерировать, и последующего выбора эффективной аппаратной архитектуры, целесообразным представляется использование оценки результатов решения контекстно связанных задач [13]. На рис. 1 приведена структурная схема алгоритма однократного решения оптимизационной задачи.

Блок «Построение математической модели задачи» – описывает процесс определения размерности решаемой задачи, целевой функции, нечетких ограничений и переменных.

Блок «Определение контекстно связанных задач» – описывает процесс выбора из рассмотренных ранее задач, имеющих схожие параметры математических моделей, с рассматриваемой [13]. Оценивается схожесть целевых функций (например, порядки полиномов), области определения и т.д.

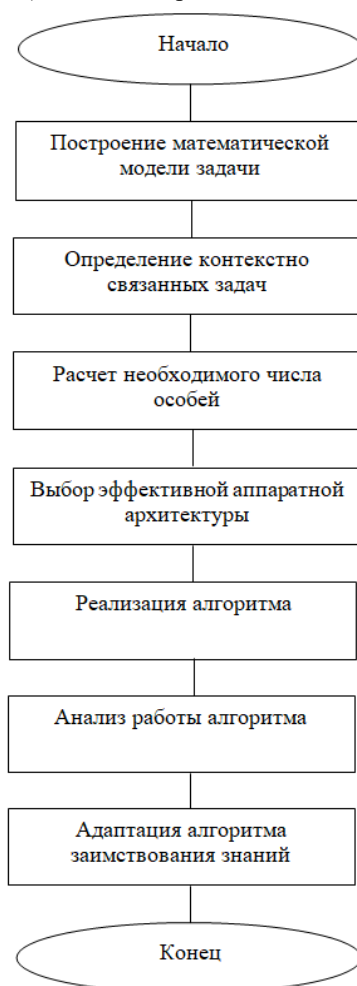


Рис. 1. Структурная схема алгоритма однократного решения оптимизационной задачи

Блок «Расчет необходимого числа особей» – приближенно определяет минимальное необходимое число особей для решения оптимизационной задачи на основе сопоставлений характеристик решаемой и рассмотренных ранее задач (целевые функции, области определения и т.д.). Определяется количество особей, которые необходимо заимствовать, также число особей, которые необходимо сгенерировать. При заимствовании особей целесообразно применять метод рассеивания, при котором часть популяции заимствуется по критерию качества (исходной и целевой функций), а часть – по расстоянию от уже имеющихся особей [7, 8].

Блок «Выбор эффективной аппаратной архитектуры» на основе прогнозируемого числа особей, обрабатываемых генетическим алгоритмом, определяет наиболее предпочтительную аппаратную архитектуру. Точная граница, как функция от числа обрабатываемых стохастическим алгоритмом особей, наиболее приемлемой архитектуры, не может быть определена точно [10]. По этой причине границу наиболее эффективной аппаратной архитектуры можно задать в виде нечеткого множества [18]:

$$\tilde{B} = \{(x, \mu_1(x))\},$$

где  $\tilde{B}$  – нечеткое число, описывающее границы эффективности аппаратных архитектур;  $x$  – число особей, обрабатываемых генетическим алгоритмом;  $\mu_1(x)$  – функция принадлежности к границе эффективности аппаратных архитектур. Универсальное множество  $X = \{X_0, X_1, \dots, X_m\}$  описывает допустимое количество особей, используемое генетическим алгоритмом в рамках рассматриваемой задачи.

При такой постановке чем меньше значение  $\mu_1(x)$ , тем меньшая противоречивость возникает при выборе аппаратной архитектуры, на которой будет реализован генетический алгоритм, которому предположительно потребуется обработать  $x$  особей. Если же  $\mu_1(x)$  стремится к единице, то возрастает сложность выбора наиболее эффективной аппаратной архитектуры, при этом выигрыш, при правильном выборе аппаратной архитектуры, будет уменьшаться. Задав  $\alpha$ -срез [13] множества  $\tilde{B}$ , можно определить подмножество  $X_0$  множества  $X$ ,  $X_0 \subseteq X$  для которого выбор наиболее эффективной аппаратной архитектуры с одной стороны сложен, а с другой не приносит существенного выигрыша в производительности. В случае, если  $x \subseteq X_0$ , то эволюционный алгоритм можно выполнить на любой из двух граничащих аппаратных архитектур, ранжированных по критерию эффективности, без существенных временных потерь.

Блок «Реализация алгоритма» – описывает процесс реализации генетического алгоритма с заданными параметрами на выбранной аппаратной архитектуре. При генерации особей начальной популяции, в зависимости от полноты знаний о решаемой задаче, целесообразно использовать эвристики, позволяющие [5, 9]:

- ♦ получать наборы решений, расположенные на несущественном удалении от области субоптимальных решений;
- ♦ получать наборы решений, распределенные по всему пространству поиска.

После того, как особи стартовой популяции сгенерированы и заимствованы, необходимо проанализировать распределение особей и удалить дублирующие. При необходимости в стартовую популяцию добавляются новые особи на основе метода рассеивания при помощи локального и глобального расширения [7, 8].

Блок «Анализ работы алгоритма» – сопоставляет фактическое время работы алгоритма на выбранной аппаратной архитектуре с расчетным.

Блок «Адаптация алгоритма заимствования знаний» – корректирует алгоритм переноса имеющихся знаний на основе отклонений прогнозируемых параметров процесса решения задачи от фактических значений.

**Вычислительный эксперимент.** Произвести точный расчет числа особей, которые должны быть обработаны генетическим алгоритмом, для получения приемлемого решения не всегда возможно. Поэтому было проведено моделирование процесса влияния погрешности оценки необходимого числа особей с учетом заимствования части решений (блок «Расчет необходимого числа особей») на работу генетического алгоритма с временной сложностью близкой к  $O(n^2)$ . Погрешность  $\Delta n$  априорной оценки необходимого числа особей определялась по формуле:

$$\Delta n = (n_f - n_p) / n_f,$$

где  $n_p$  – прогнозируемое количество особей, которые необходимо обработать ГА для получения приемлемого решения,  $n_f$  – фактическое количество особей, обработанное ГА в процессе получения приемлемого решения.

Погрешность  $\Delta t$  априорной оценки необходимого времени для реализации алгоритма определялась по формуле:

$$\Delta t = (t_f - t_p) / t_f,$$

где  $t_p$  – прогнозируемое время, которое необходимо ГА для обработки  $n_p$  особей,  $t_f$  – фактическое время, потребовавшееся ГА для получения приемлемого решения, т.е. обработки  $n_f$  особей.

На время выполнения ГА на эффективной аппаратной архитектуре, наряду с  $n_f$  и  $n_p$ , оказывает влияние принадлежность  $n_f$  и  $n_p$  к общей границе эффективности аппаратных архитектур. Поэтому вычислительный эксперимент проводился с учетом следующих ограничений:

- $n_f$  и  $n_p$  принадлежат к одной, общей границе эффективности;
- $n_f > n_p$ ;
- $\mu_1(n_p) < \mu_1(n_f) < 1$  ;
- $\forall n_i : n_p < n_i < n_f \rightarrow \mu_1(n_p) < \mu_1(n_i) < \mu_1(n_f)$ .

В процессе вычислительного эксперимента выявлена зависимость погрешности  $\Delta t$  времени работы алгоритма от погрешности  $\Delta n$  оценки необходимого числа особей, приведённая на рис.2 в виде графика.

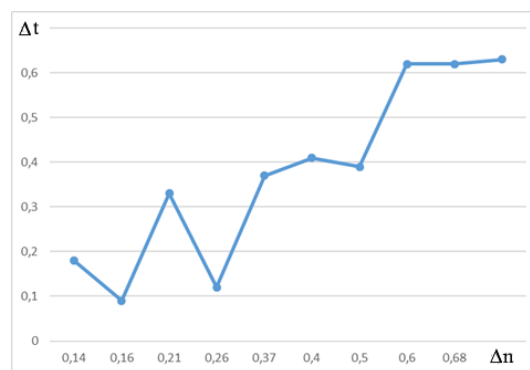


Рис. 2. График зависимости погрешности времени работы алгоритма от погрешности оценки необходимого числа особей

Скачкообразность графика обусловлена стохастичностью ГА, в том числе, наличием единичных результатов его работы, существенно отличающихся от усредненных значений. Сглаженный график зависимости погрешности  $\Delta t_s$  времени работы алгоритма от погрешности  $\Delta n$  оценки необходимого числа особей приведён на рис. 3.

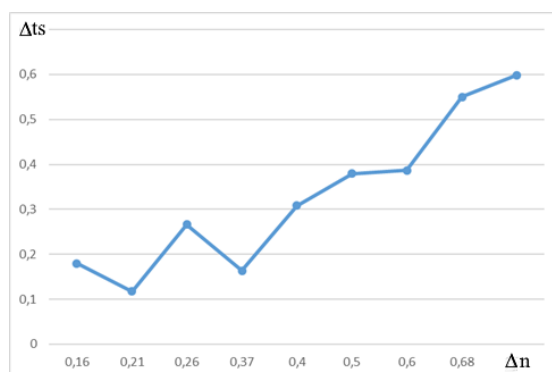


Рис. 3. Сглаженный график зависимости погрешности времени работы алгоритма от погрешности оценки необходимого числа особей

Представленные на рис. 3 результаты эксперимента позволяют предположить, зависимость погрешности времени работы алгоритма от погрешности оценки необходимого числа особей близка к  $O(n)-O(n^2)$ .

В дальнейшем, если полученное решение окажется неприемлемым, по причине дополнительно введенных ограничений, необходимо повторно решить задачу, скорректировав при необходимости математическую модель. Если изменения несущественны, можно воспользоваться знаниями, полученными при предшествующем решении задачи:

- ◆ количестве особей, которые потребуется сгенерировать;
- ◆ частью решений (начальных, промежуточных, итоговых);
- ◆ оценками времени работы генетического алгоритма на аппаратных архитектурах.

После реализации алгоритма может потребоваться корректировка степеней нечеткости параметров решаемой задачи. В особо сложных случаях следует выполнить процедуру декомпозиции. Декомпозиционные методы заключаются в последовательном, параллельном, параллельно-последовательном (последовательно-параллельном) разбиении модели пространства поиска. Суть общей задачи декомпозиции состоит в делении целого на части по определенным правилам [11, 20]. Одним из таких правил является то, что цель более высокого уровня иерархии может быть достигнута только через реализацию детализирующих (обеспечивающих) её подцелей. Таким образом, основная задача структуризации целей – последовательное развертывание исходной (иначе, «генеральной») цели на множество обеспечивающих ее подцелей и задач для формирования детального и полного информационного представления о процессе достижения исходной цели [20]. Поэтому, время достижения «генеральной» цели или промежуточных подцелей напрямую зависит от времени решения задач, составляющих цель или подцели. По этой причине, актуальной является задача выбора эффективной аппаратной архитектуры для решения задач, образующих цель и подцели.

**Заключение.** Предложен подход к выбору эффективной аппаратной архитектуры, состоящий из трех этапов: анализа результатов решения рассмотренных ранее задач, определения количества особей, которые необходимо сгенерировать для

получения приемлемого решения эволюционным алгоритмом, выбора эффективной аппаратной архитектуры, на которой будет реализован эволюционный алгоритм. На этапе анализа определяются задачи, схожие с решаемой, и параметры эволюционных алгоритмов, решающих данные задачи. Поиск в базе знаний, содержащей информацию о всех решенных ранее задачах, может быть неприемлемо длительным, поэтому, целесообразно использовать маркерные задачи, содержащие шаблонные характеристики целевых функций и областей поиска решений. В качестве критериев сходства решаемой задачи с рассмотренными ранее являются: размерности пространств поиска, прерывистости целевых функций, порядки полиномов в целевых функциях и ограничениях и другие. На основе проведенного анализа, определяется приближенная оценка количества особей, которое необходимо сгенерировать для получения приемлемого решения. На основании оценки необходимого количества особей определяется эффективная аппаратная архитектура, на которой будет реализован эволюционный алгоритм.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Курейчик В.М. Особенности построения систем поддержки принятия решений // Известия ЮФУ. Технические науки. – 2012. – № 7 (132). – С. 92-98.
2. Deb K., Myburgh C. Breaking the Billion-Variable Barrier in Real-World Optimization Using a Customized Evolutionary Algorithm // Proceedings of Genetic and Evolutionary Computation Conference. – 2016. – P. 653-660.
3. Родзин С.И., Скобцов Ю.А., Эль-Хатиб С.А. Биоэвристики: теория, алгоритмы и приложения: монография. – Чебоксары: ИД «Среда», 2019. – 224 с.
4. Курейчик В.М., Данильченко В.И. Генетический алгоритм планирования размещения СБИС // Известия ЮФУ. Технические науки. – 2019. – № 2 (204). – С. 26-34.
5. Гладков Л.А., Курейчик В.В., Курейчик В.М. Генетические алгоритмы. – М.: Физматлит, 2006, 2010. – 386 с.
6. Карпенко А.П. Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой. – М.: Изд-во МГТУ им. Н.Э. Баумана, 2014. – 446 с.
7. Пантелеев А.В., Метлицкая Д.В., Алешина Е.А. Методы глобальной оптимизации. Метаэвристические стратегии и алгоритмы. – М.: Вузовская книга, 2013. – 244 с.
8. Пантелеев А.В. Применение эволюционных методов глобальной оптимизации в задачах оптимального управления детерминированными системами. – М.: Изд-во МАИ, 2013. – 160 с.
9. Курейчик В.М. Гибридные генетические алгоритмы // Известия ЮФУ. Технические науки. – 2007. – № 2 (77). – С. 5-12.
10. Агбалов О.И., Венцов Н.Н. Оценка зависимостей времени работы генетического алгоритма, выполняемого на CPU и GPU // Кибернетика и программирование. – 2017. – № 6. – С. 1-8. – DOI: 10.25136/2306-4196.2017.6.24509. – URL: [http://e-notabene.ru/kp/article\\_24509.html](http://e-notabene.ru/kp/article_24509.html).
11. Чернышев В.О., Грабауров В.А. Системный подход к управлению субъектами хозяйствования / под ред. Ф.А. Романюка. – Минск: БНТУ, 2014. – 272 с.
12. Fan Zhang, Zheng Li, Bingnan Wang, Maosheng Xiang, Wen Hong. Hybrid general-purpose computation on GPU (GPGPU) and computer graphics synthetic aperture radar simulation for complex scenes // International Journal of Physical Sciences. – 16 February, 2012. – Vol. 7 (8). – P. 1224-1234.
13. Shell J., Coupland S. Fuzzy Transfer Learning: Methodology and Application // Preprint submitted to Information Sciences. – May 23, 2014. – 27 p.
14. Samitha Herath, Mehrtash Harandi, Basura Fernando, Richard Nock. Min-Max Statistical Alignment for Transfer Learning // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. – 2019. – P. 9288-9297.
15. Yiyang Li, Guanyu Tao, Weinan Zhang, Yong Yu, Jun Wang. Content Recommendation by Noise Contrastive Transfer Learning of Feature Representation // Proceedings of the 2017 ACM Conference on Information and Knowledge Management, ACM. – 2017. – P 1657-1665.
16. Pan S.J., Yang Q. IEEE Transactions on knowledge and data engineering // Institute of Electrical and Electronics Engineers, Inc., NY, 2010. – Vol. 22 (10). – P 1345-1359.



17. Сапрыкин А.Н., Акинина К.Д., Сапрыкина Е.Н. Нахождение оптимального числа полезных особей в популяции и конвергируемых поколений генетического алгоритма оптимизации простых многоэкстремальных функций // Actualscience. – 2016. – Т. 2. № 11. – С. 168-169.
18. Zadeh L.A. Fuzzy sets // Information and Control. – 1965. – Vol. 8. – P. 338.
19. Нечеткие множества в моделях управления и искусственного интеллекта / под ред. Д.А. Поспелова. – М.: Наука, 1986 – 312 с.
20. Курейчик В.М., Данильченко В.И. Классификация и анализ методов решения задачи размещения СБИС // Информатика, вычислительная техника и инженерное образование. – 2018. – № 1 (32). – С. 21-40.

#### REFERENCES

1. 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.
2. Deb K., Myburgh C. Breaking the Billion-Variable Barrier in Real-World Optimization Using a Customized Evolutionary Algorithm, *Proceedings of Genetic and Evolutionary Computation Conference*, 2016, pp. 653-660.
3. Rodzin S.I., Skobtsov Yu.A., El'-Khatib S.A. Bioevristiki: teoriya, algoritmy i prilozheniya: monografiya [Bymistake: theory, algorithms and applications: monograph]. Cheboksary: ID «Sreda», 2019, 224 p.
4. Kureychik V.M., Danil'chenko V.I. Geneticheskiy algoritm planirovaniya razmeshcheniya SBIS [Genetic algorithm of VLSI placement planning], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2019, No. 2 (204), pp. 26-34.
5. Gladkov L.A., Kureychik V.V., Kureychik V.M. Geneticheskie algoritmy [Genetic algorithm]. Moscow: Fizmatlit, 2006, 2010, 386 p.
6. Karpenko A.P. Sovremennyye algoritmy poiskovoy optimizatsii. Algoritmy, vdokhnovlennyye prirodoy [Modern search engine optimization algorithms. Nature-inspired algorithms]. Moscow: Izd-vo MGTU im. N.E. Bauman, 2014, 446 p.
7. Pantelev A.V., Metlitskaya D.V., Aleshina E.A. Metody global'noy optimizatsii. Metaevristicheskie strategii i algoritmy [Methods of global optimization. Metaheuristic strategies and algorithms]. Moscow: Vuzovskaya kniga, 2013, 244 p.
8. Pantelev A.V. Primeneniye evolyutsionnykh metodov global'noy optimizatsii v zadachakh optimal'nogo upravleniya determinirovannymi sistemami [Application of evolutionary methods of global optimization in problems of optimal control of deterministic systems]. Moscow: Izd-vo MAI, 2013, 160 p.
9. Kureychik V.M. Gibriddnye geneticheskie algoritmy [Hybrid genetic algorithms], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2007, No. 2 (77), pp. 5-12.
10. Agibalov O.I., Ventsov N.N. Otsenka zavisimostey vremeni raboty geneticheskogo algoritma, vypolnyaemogo na CPU i GPU [Estimation of time dependencies of genetic algorithm running on CPU and GPU], *Kibernetika i programmirovaniye* [Cybernetics and programming], 2017, No. 6, pp. 1-8. DOI: 10.25136/2306-4196.2017.6.24509. Available at: [http://e-notabene.ru/kp/article\\_24509.html](http://e-notabene.ru/kp/article_24509.html).
11. Chernyshev V.O., Grabaurov V.A. Sistemnyy podkhod k upravleniyu sub"ektami khozyaystvovaniya [A systematic approach to the management of economic entities], ed. by F.A. Romanyuka. Minsk: BNTU, 2014, 272 p.
12. Fan Zhang, Zheng Li, Bingnan Wang, Maosheng Xiang, Wen Hong. Hybrid general-purpose computation on GPU (GPGPU) and computer graphics synthetic aperture radar simulation for complex scenes, *International Journal of Physical Sciences*, 16 February, 2012, Vol. 7 (8), pp. 1224-1234.
13. Shell J., Coupland S. Fuzzy Transfer Learning: Methodology and Application, *Preprint submitted to Information Sciences*, May 23, 2014, 27 p.
14. Samitha Herath, Mehrtash Harandi, Basura Fernando, Richard Nock. Min-Max Statistical Alignment for Transfer Learning, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9288-9297.

15. *Yiyang Li, Guanyu Tao, Weinan Zhang, Yong Yu, Jun Wang*. Content Recommendation by Noise Contrastive Transfer Learning of Feature Representation, *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, ACM*, 2017, pp 1657-1665.
16. *Pan S.J., Yang Q.* IEEE Transactions on knowledge and data engineering, *Institute of Electrical and Electronics Engineers, Inc., NY, 2010*, Vol. 22 (10), pp 1345-1359.
17. *Saprykin A.N., Akinina K.D., Saprykina E.N.* Nakhozhdenie optimal'nogo chisla poleznykh osobey v populyatsii i konvergiruemykh pokoleniy geneticheskogo algoritma optimizatsii prostykh mnogoekstremal'nykh funktsiy [Finding the optimal number of useful individuals in the population and converged generations of the genetic algorithm for optimization of simple multi-extreme functions], *Actualscience [Actualscience]*, 2016, Vol. 2. No. 11, pp. 168-169.
18. *Zadeh L.A.* Fuzzy sets, *Information and Control*, 1965, Vol. 8, pp. 338.
19. *Nechetkie mnozhestva v modelyakh upravleniya i iskusstvennogo intellekta [Fuzzy sets in control and artificial intelligence models]*, ed. by D.A. Pospelova. Moscow: Nauka, 1986, 312 p.
20. *Kureychik V.M., Danil'chenko V.I.* Klassifikatsiya i analiz metodov resheniya zadachi razmeshcheniya SBIS [Classification and analysis of methods for solving the problem of VLSI placement], *Informatika, vychislitel'naya tekhnika i inzhenernoe obrazovanie [Computer science, computer engineering and engineering education]*, 2018, No. 1 (32), pp. 21-40.

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

**Чернышев Юрий Олегович** – Донской государственный технический университет, e-mail: myvnn@list.ru; г. Ростов-на-Дону, Площадь Гагарина, 1; тел.: 88632738510; кафедра автоматизации производственных процессов; д.т.н.; профессор.

**Венцов Николай Николаевич** – e-mail: vencov@list.ru; тел.: 88632738582; кафедра информационных технологий; к.т.н.

**Chernyshev Yury Olegovich** – Don State Technical University; e-mail: myvnn@list.ru; 1, Gagarin square, Rostov-on-Don, Russia; phone: +78632738510; the department of automation of productions; dr. of eng. sc.; professor.

**Ventsov Nikolay Nikolaevich** – e-mail: vencov@list.ru; phone: +78632738582; the department of information technologies; cand. of eng. sc.; associate professor.

УДК 004.056.5

DOI 10.23683/2311-3103-2019-4-68-80

**С.А. Ховансков, В.А. Литвиненко, В.С. Хованскова**

### **МЕТОДИКА ЗАЩИТЫ РАСПРЕДЕЛЕННЫХ ВЫЧИСЛЕНИЙ В МНОГОАГЕНТНОЙ СИСТЕМЕ\***

*Рассматривается организация и защита распределенных вычислений на основе многоагентной системы для решения задач многовариантного моделирования. При моделировании выбор одного из многих вариантов может потребовать перебора огромного множества параметров недоступного для быстройдействующей ЭВМ. Для сокращения времени решения таких задач используют распределенные вычисления. Существует множество различных подходов для организации распределенных вычислений в компьютерной сети - технология grid, metacomputing (BOINC, PVM и другие). Основным недостатком большинства существующих подходов является то, что они предназначены для создания централизованных систем распределенных вычислений. Распределенные вычисления организуются на основе многоагентной системы на вычислительных узлах любой компьютерной сети. При использовании в качестве вычислительной среды компьютерную сеть большого масштаба могут возникнуть угрозы безопасности распределенных вычислений со стороны злоумышленников. Одной из таких угроз является получение в процессе вычислений ложного результата злоумышленником. Ложный результат может привести в процессе моделирования к принятию не оптимального, либо неправильного решения. Разработан метод защиты распределенных вычислений на основе много-*

\* Работа выполнена при поддержке РФФИ (проект № 18-01-00041).