

## Обзор и анализ существующих модификаций генетических алгоритмов

*Национальный аэрокосмический университет им. Н. Е. Жуковского «ХАИ»*

Представлены обзор и анализ существующих модификаций генетических алгоритмов, показаны их основные достоинства и недостатки. Предложены классификация основных направлений модификации генетического алгоритма, а также методика тестирования генетических алгоритмов. Для тестирования использованы девять задач различной сложности и размерности. В качестве тестируемых алгоритмов выбраны двадцать две модификации простого генетического алгоритма, сходные по архитектуре, сложности реализации и потреблению ресурсов. В результате исследования определены перспективные направления модификаций генетических алгоритмов с целью повышения их эффективности.

**Ключевые слова:** генетические алгоритмы, модификация, тестирование, информационная сложность, трудоемкость метода, достоверность ответа.

### Введение

Одним из наиболее важных направлений научно-технического прогресса является развитие систем искусственного интеллекта, способных расширить круг решаемых человечеством задач. Совместно с развитием вычислительно-информационных технологий стало возможным автоматизировать процесс решения технических и экономических вопросов, ранее решаемых вручную. При этом, часть таких задач, практически не поддающихся решению классическими методами, решаются системами вычислительного интеллекта. Эти системы опираются на эвристические алгоритмы, используемые в следующих областях знаний:

- нечеткой логике и теории множеств;
- нечетких экспертных системах;
- системах приближенных вычислений;
- теории хаоса;
- фрактальном анализе;
- нелинейных динамических системах;
- нейронных сетях;
- эволюционных вычислениях.

Из них наиболее перспективными являются эволюционные вычисления, которые подразделяются на эволюционные стратегии, эволюционное программирование, генетические алгоритмы и генетическое программирование. В свою очередь, из эволюционных вычислений наибольшее распространение получили генетические алгоритмы (ГА), способные решать сложные оптимизационные задачи большой размерности.

ГА обладают следующими преимуществами: они не требуют никакой дополнительной информации о поверхности функции выбора; разрывы, существующие на ней, незначительно влияют на эффективность оптимизации; устойчивы к попаданию в локальные оптимумы; хорошо работают при решении задач многоцелевой оптимизации; могут быть использованы для широкого класса задач; ГА просты в реализации легко и удобно распараллеливаются.

Однако ГА обладают и существенными недостатками. Например, процесс создания исходного набора хромосом, критерии их отбора и используемые про-

цедуры являются эвристическими и далеко не гарантируют нахождения глобального экстремума, т. к. эволюция может начать вычисляться бесконечно долго на какой-либо непродуктивной ветви. И, наоборот, два неперспективных родителя, исключенные из эволюции генетическим алгоритмом, оказываются способными через несколько итераций произвести высокоэффективного потомка. Это становится особенно заметно при решении задач большой размерности со сложными внутренними связями. Не желательно и проблематично использовать генетические алгоритмы в случаях, когда необходимо найти точный глобальный оптимум, время расчета функции выбора велико, поверхность функции выбора имеет слабо изменяющийся рельеф.

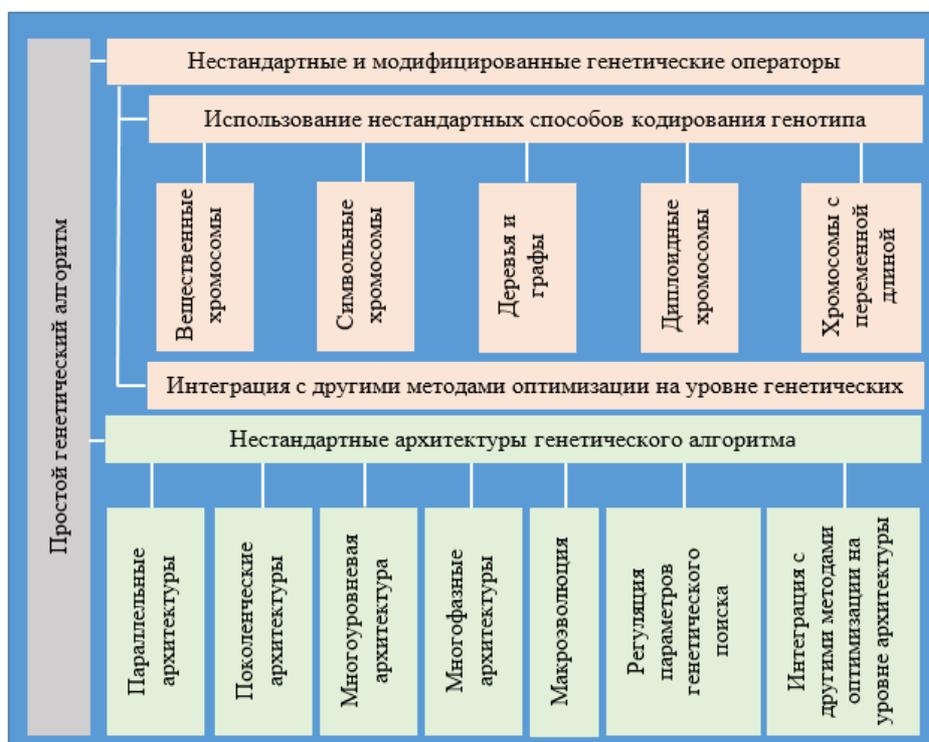
С целью совершенствования ГА и преодоления указанных недостатков создано и продолжает разрабатываться значительное число модификаций ГА.

### 1. Анализ основных направлений повышения эффективности генетических алгоритмов

Для устранения основных недостатков проводятся исследования по созданию математических моделей модификаций генетического алгоритма в следующих направлениях:

- нестандартные архитектуры генетического поиска;
- использование нестандартных или модифицированных генетических операторов;
- различные способы кодирования генотипа.

На рисунке показана классификация модификаций генетических алгоритмов.



Классификация модификаций генетических алгоритмов

Проанализируем несколько существующих модификаций ГА с использованием нестандартных способов кодирования. В работе [1] была рассмотрена возможность смещения отдельных подмножеств битов внутри строки по нелинейной оптимизации. В работе [2] были предложены различные модификации схем генетического переупорядочивания внутри строки, когда используется кроссинговер.

Исследователи ГА предлагают много других операторов отбора, кроссинговера и мутации. Самый распространённый - турнирный отбор [3, 4]. Турнирный отбор реализует  $l$  турниров, чтобы выбрать  $l$  особей. Каждый турнир построен на выборке  $k$  элементов из популяции и выбора лучшей особи среди них. Наиболее распространён турнирный отбор с  $k=2$ . Элитные методы отбора [5] гарантируют, что при отборе обязательно будут выживать лучшие члены популяции совокупности.

Двухточечный [4] и равномерный кроссинговеры являются альтернативой одноточечному оператору. В двухточечном кроссинговере выбирают две точки разрыва, и родительские хромосомы обмениваются сегментом, который находится между двумя этими точками. В равномерном кроссинговере каждый бит первого родителя наследуется первым потомком с заданной вероятностью; в противном случае этот бит передается второму потомку. В работе [6] предложен оператор нечёткого кроссинговера. ГА с таким оператором проигрывает простому ГА при оптимизации унимодальных функций и даёт выигрыш в скорости при оптимизации многомодальных функций.

Среди нестандартных способов кодирования генотипа распространены диплоидные хромосомы [7]. При таком способе кодирования хромосома состоит из двух строк X-хромосомы и Y-хромосомы. В работе [7] реализован принцип доминантности генов. Хромосома состоит из трех строк: двух строк X-хромосомы и Y-хромосомы, в которых содержится полная генетическая информация о данной особи и одной опорной, в которой присутствуют признаки доминантности генов либо X-хромосомы, либо Y-хромосомы. Соответственно в данной модификации генетического алгоритма используется свой набор генетических операторов.

Одно из направлений модификации генетического алгоритма заключается в разделении популяции на отдельные участки (ниши), в каждой из которых осуществляется поиск альтернативных вариантов решений. Большинство известных ГА принимают размер популяции постоянным (ГА "стационарного состояния"). Существуют модификации ГА, допускающие варьирование размера популяции как на различных итерациях алгоритма, так и на одной итерации после выполнения операций скрещивания и мутации ("поколенческие" ГА).

Также используется принцип «многодетной семьи» [8], который заключается в многократном применении генетических операторов к выбранной паре родителей с целью получения потомства с большей приспособленностью (лучшем значением функции выбора), чем у его родителей.

В настоящее время осуществляются исследования нестандартных архитектур генетического поиска, позволяющих в большинстве случаев решать проблему предварительной сходимости алгоритмов. Это методы миграции и искусственной селекции [9], метагенетической параметрической оптимизации, стохастически-итерационные генетические и поисковые, прерывистого равновесия, объединения генетического поиска и моделирования отжига, В работе [9] в отличие от традиционных ГА выполняется макроэволюция, то есть создается не одна популяция, а некоторое их множество. Генетический поиск здесь осуществляется путем объединения родителей из различных популяций.

Генетический алгоритм с элементами саморегуляции [10] основан на архитектуре простого ГА, но отличается динамически изменяющимися параметрами в ходе работы алгоритма и нестандартной архитектурой генетического поиска. Нестандартная архитектура заключается в добавлении блока оценки предварительной сходимости и блока адаптации. Добавление блока адаптации позволило адаптировать алгоритм по архитектуре генетического поиска на основе оценок, содержащихся во внутренней базе данных [11], и, таким образом, управлять генетическим поиском. Блок адаптации на основе блока оценки предварительной сходимости решает вопрос об изменении вероятностей генетических операторов. Это означает, что если алгоритм сходится на протяжении определенного числа итераций, то вероятности генетических операторов увеличиваются, иначе возвращаются к первоначальному значению. Этим методом реализуется выход из локальных экстремумов. Если же этот метод не дает улучшений, то блок адаптации заменяет часть популяции на случайно сгенерированные особи. Для компенсации быстрой сходимости вызываемой элитным отбором используется аутбриндинг. Данный алгоритм наравне со своими достоинствами имеет следующие недостатки: повышенный расход памяти и сложности в реализации блоков адаптации и оценки предварительной сходимости и необходимости хранения оценок во внутренней базе данных.

Двухуровневый генетический алгоритм [12] представляет собой двухуровневую систему: на 1-м уровне протекают процессы обычного ГА, в котором каждый индивид представляет собой решение некоторой задачи, на 2-м уровне в качестве индивидов выступают ГА 1-го уровня и каждая хромосома содержит информацию о генетических операторах, размере популяции, вероятности миграции и т. д. Поскольку индивиды 2-го уровня являются параметрами управления, то их структуры существенно различаются. Функция выбора ГА 2-го уровня представляет собой оценку эффективности соответствующего процесса ГА 1-го уровня. Лучшие ГА 1-го уровня имеют более высокую вероятность выживания и скрещивания на 2-м уровне, тогда как худшие процессы ГА 1-го уровня постепенно исчезают из популяции 2-го уровня. Данный алгоритм является эффективным, особенно в сочетании с кластеризацией, легко распараллеливается для работы на нескольких ЭВМ, но сложен в реализации и требователен к вычислительным и временным ресурсам.

В работе [13] предложен ГА с переменной структурой. Работа такого алгоритма разбивается на несколько этапов, в каждом из которых применяется свой набор генетических операторов. Например, в первом этапе используются турнирный отбор, унифицированный кроссинговер и стандартная мутация, во втором – селекция на основе рулетки, двуточечный кроссинговер и точечная мутация, а в третьем – пропорциональный отбор, стандартный кроссинговер, точечная мутация и оператор инверсии, причём в разных этапах – различные заранее заданные вероятности генетических операторов. Данный алгоритм сложнее простого ГА, сильно зависит от выбранных этапов и не обладает адаптивностью к рельефу функции выбора.

Особое место среди модификаций генетических алгоритмов занимают параллельные ГА [8, 9, 4]. Целью данных модификаций является распараллеливание эволюции для одновременной обработки на нескольких ЭВМ. Обычно на каждой ЭВМ моделируется некоторая субпопуляция, развивающаяся отдельно от остальных субпопуляций на других ЭВМ. Иногда в каждой субпопуляции используются различные генетические операторы с разными вероятностями, что гаран-

тирует различность популяций. Между ЭВМ посредством миграции субпопуляции обмениваются генетическим материалом. Таким образом, решается задача выхода из локального экстремума и проблема со скоростью сходимости. Но данный метод сложен в реализации и требует значительного числа вычислительных ресурсов.

В результате современные исследования ГА концентрируются вокруг следующих направлений:

- развитие более строгого математического аппарата ГА как метода оптимизации;
- анализ проблем, которые являются трудными при решении задач с использованием ГА [15];
- анализ зависимости эффективности алгоритма от различных генетических операторов и их параметров [4, 16];
- сравнение ГА с другими методами оптимизации и анализ возможных схем интеграции ГА с другими методами оптимизации;
- использование ГА в качестве базового для моделирования систем искусственной жизни [17];
- реализация ГА на параллельных архитектурах [18].

## **2. Методика тестирования генетических алгоритмов**

Критерий эффективности вычислительного метода – информационная сложность – для эволюционного поиска определяется свойствами функции выбора. В работах [11, 14] показано, что генетические алгоритмы эффективны для хорошо коррелированных поверхностей и больших популяций. Существуют различные варианты определения функции выбора. Она может быть определена как число, значение которого постоянно для данного индивида. В других задачах функция выбора определяется как способность индивида выполнять определенные задачи. В то же время функция выбора может быть определена как связь индивида с окружающей средой. В этом случае функция выбора изменяется при изменении среды. Одним из способов реализации этого подхода является введение конкуренции в популяцию.

В качестве первого критерия сравнения выбрана трудоёмкость метода решения поставленной задачи. ГА очень требовательны к ресурсам вычислительной техники, и решение современных важных задач отнимает много времени. В ГА основной расход времени идет на вычисление приспособленности каждой особи, временем, затрачиваемым на вычисление генетических операторов, можно пренебречь. Поскольку больше всего вычислительных ресурсов ЭВМ тратится именно на вычисление значений функции выбора, то этот критерий отражает некоторую относительную временную характеристику скорости решения задачи. За эталон скорости берётся скорость решения тестовой задачи неким базовым алгоритмом, относительно которого вычисляются оценки скорости остальных алгоритмов. Следовательно, в качестве критерия трудоёмкости метода решения поставленной задачи можно взять среднюю статистическую оценку числа вычислений функции выбора на протяжении всей эволюции.

Второй критерий сравнения ГА – достоверность ответа. ГА не гарантируют точного решения и поэтому применяются в тех случаях, когда требуется не точный, а приближенный ответ. В качестве критерия достоверности выступает процент нахождения тестируемым алгоритмом глобального экстремума с выбранной

точностью. По сути этот критерий представляет собой вероятность события, когда тестовый алгоритм в качестве решения поставленной перед ним задачи выдаст глобальный экстремум.

Чтобы получить некую численную оценку информационной сложности каждого алгоритма, необходимо ввести интегральную оценку трудоёмкости метода и достоверности ответа. Критерий достоверности имеет приоритетную важность, т. к. целью оптимизации является именно нахождение глобального экстремума. Критерий трудоёмкости является менее приоритетным. В качестве интегральной оценки информационной сложности берётся произведение критериев трудоёмкости и достоверности, отнесённых к их максимальным значениям. Такой вид оценки взят из следующих соображений. Пусть имеем два алгоритма: первый решает некую оптимизационную задачу в два раза быстрее, чем второй алгоритм, но при этом процент нахождения глобального экстремума вдвое меньше, чем у второго алгоритма. Интегральная оценка информационной сложности у обоих алгоритмов будет одинакова.

Если тестовых задач несколько, то рейтинг алгоритма по всем задачам вычисляют следующим образом:

- для каждой тестовой задачи составляют таблицу тестируемых алгоритмов, упорядоченную по полученной оценке эффективности;
- в соответствии с порядком тестируемого алгоритма в составленной таблице, каждому алгоритму назначают число баллов, равное разности количества тестируемых алгоритмов и номера в таблице;
- рейтинг тестируемого алгоритма будет равен сумме баллов всех тестовых задач.

Содержание методики [19] численной оценки информационной сложности следующее.

1. Определить множество тестируемых алгоритмов:  $\{Alg_1, Alg_2, \dots, Alg_N\}$ .
2. Определить множество тестовых задач:  $\{f_1, f_2, \dots, f_M\}$ .
3. Выбрать общие параметры тестируемых алгоритмов для каждой тестовой функции (для генетических алгоритмов – вероятности генетических операторов, размер популяции, способ кодирования, длина хромосомы). Выбрать требуемую точность решений  $\mathcal{E}$  и количество испытаний  $K$ .
4. Из множества тестируемых алгоритмов выбрать базовый алгоритм, относительно которого будут вычисляться оценки эффективности тестируемых алгоритмов.
5. Для каждой тестовой функции  $f_i, i = \overline{1, M}$ , из множества тестовых функций последовательно  $K$  раз испытывать каждый тестируемый алгоритм  $Alg_j, j = \overline{1, N}$  путём нахождения глобального экстремума. При этом вычислить следующие характеристики:  $C_{i,j}^{Гл.Экстр.}$  – количество испытаний, в которых алгоритм  $Alg_j$  нашёл глобальный экстремум тестовой функ-

ции.  $C_{i,j}^{Выч.ф.в.}$  – среднее число вычислений функции выбора в процессе поиска экстремума тестовой функции  $f_i$  алгоритмом  $Alg_j$ .

6. Для всей серии проведённых испытаний вычислить интегральный критерий эффективности оптимизации тестовой функции  $f_i$  алгоритмом

$$C_{i,j} = \frac{C_{i,j}^{Гл.Экстр.} \cdot C_{i,j}^{Выч.ф.в.}}{K} \cdot 100\%$$

7. Построить таблицы для каждой тестовой функции  $f_i$  упорядоченные по убыванию  $C_{i,j}$ ; в соответствии с порядковым номером  $l_{i,j}$  тестируемого алгоритма  $Alg_j$ , вычислить баллы:  $B_{i,j} = N - l_{i,j}$ .

8. Вычисляется рейтинг всех тестируемых алгоритмов  $Alg_j, j = \overline{1, N}$  как

$$\text{сумму баллов } R_j = \sum_{i=1}^M B_{i,j}.$$

9. Построить таблицу тестируемых алгоритмов и их рейтингов, упорядоченная по убыванию рейтингов. Алгоритмы с наиболее высокими рейтингами являются более эффективными по сравнению с алгоритмами с низким рейтингом.

В соответствии с выбранной методикой составлен алгоритм оценки вычислительной сложности генетических алгоритмов [19]. Он включает следующие операции.

1. Инициализация цикла по тестируемым алгоритмам: присвоить  $j = 1$ .
2. Инициализация цикла по тестовым функциям: присвоить  $i = 1$ ; присвоить  $C_{i,j}^{Гл.Экстр.} = 0$ ;  $C_{i,j}^{Выч.ф.в.} = 0$ .
3. Инициализация цикла сбора статистических оценок: присвоить  $k = 1$ .
4. Сбор статистических оценок: решить  $j$ -м тестируемым алгоритмом  $i$ -ю тестовую функцию, при этом считая число вычислений функции выбора  $C_{i,j}^{Ц.Ф.}$ , если полученное решение является глобальным оптимумом  $i$ -й тестовой функции с точностью  $\varepsilon$ , то присвоить

$$C_{i,j}^{Гл.Экстр.} = C_{i,j}^{Гл.Экстр.} + \frac{C_{i,j}^{Гл.Э.}}{K};$$

$$C_{i,j}^{Выч.ц.ф.} = C_{i,j}^{Выч.ф.в.} + \frac{C_{i,j}^{Ф.В.}}{K}; k = k + 1.$$

5. Окончание цикла сбора статистических оценок: если  $k \leq K$ , то перейти к шагу 4.
6. Вычисление интегрального критерия: присвоить

$$C_{i,j} = \frac{C_{i,j}^{Гл.Экстр.} \cdot C_{i,j}^{Выч.ф.в.}}{K} \cdot 100\%; j = j + 1.$$

7. Окончание цикла по тестовым функциям: если  $j \leq M$ , то перейти к шагу 3, иначе присвоить  $i = i + 1$ .
8. Окончание цикла по тестируемым алгоритмам: если  $i \leq N$ , то перейти к шагу 2.
9. Таблица рейтингов. По полученным рейтингам  $C_{i,j}$  построить упорядоченную таблицу рейтингов тестируемых алгоритмов.

В качестве тестовых функций обычно используются сложные многомерные и многоэкстремальные функции, практически не решаемые классическими методами. В данной работе были выбраны следующие функции для тестирования алгоритмов: De Jong 2, De Jong 3, De Jong 5; Расстригина – двумерная и десятимерная; Griewank – двумерная и десятимерная.

### 3. Результаты исследования

В список тестируемых алгоритмов входят наиболее распространенные модификации ГА. С помощью методики тестирования ГА, описанной выше и программного комплекса, созданного на основании методики, были получены сравнительные характеристики алгоритмов для каждой из тестовой задачи. По сравнительным характеристикам были построены оценки информационной сложности, достоверности результата и трудоемкости каждого алгоритма [19], представленные в таблице.

Тестируемый алгоритм	Информационная сложность	Трудоемкость метода	Достоверность ответа
1	2	3	4
Простой ГА	91	107	77,3
ГА с двухточечным кроссинговером	114	110	80,95
ГА с унифицированным кроссинговером	37	64	68,37
ГА с точечной мутацией	123	159	76,28
ГА с мутацией «инцест»	82	90	74,44
ГА с мутацией с равномерным распределением потомков	163	157	89,12
ГА с регуляцией вероятностей генетических операторов	137	176	77,16
ГА с регуляцией вероятностей генетических операторов и точечной мутацией	144	183	78,49
ГА с регуляцией вероятностей генетических операторов и мутацией «инцест»	117	156	76,18
ГА с регуляцией вероятностей генетических операторов и мутацией с равномерным распределением потомков	153	163	86,77
ГА с рулеточным отбором	56	60	76,09
ГА с двухточечным кроссинговером и точечной мутацией	102	115	80,1
ГА с двухточечным кроссинговером и мутацией «инцест»	75	79	78,23
ГА с двухточечным кроссинговером и мутацией с равномерным распределением потомков	117	98	90,57

1	2	3	4
ГА с двухточечным кроссинговером и регуляцией вероятностей генетических операторов	140	151	82,67
ГА с двухточечным кроссинговером, регуляцией вероятностей генетических операторов и мутацией с равномерным распределением потомков	172	157	90,74
ГА с двухточечным кроссинговером и рулеточным отбором	44	67	74,38
ГА с унифицированным кроссинговером и с точечной мутацией	126	152	78,79
ГА с унифицированным кроссинговером и мутацией «инцест»	28	61	67,49
ГА с унифицированным кроссинговером и мутацией с равномерным распределением потомков	56	75	78
ГА с унифицированным кроссинговером и регуляцией вероятностей генетических операторов	73	116	69,4
ГА с унифицированным кроссинговером, регуляцией вероятностей генетических операторов и мутацией с равномерным распределением потомков	88	119	74,74
ГА с унифицированным кроссинговером и рулеточным отбором	39	76	67,14

Как видно из таблицы, наибольший рейтинг имеют различные варианты ГА с регуляцией вероятностей генетических операторов и мутацией с равномерным распределением потомков. При этом самым эффективным оказалась комбинация ГА с двухточечным кроссинговером, регуляцией вероятностей генетических операторов и мутацией с равномерным распределением потомков.

#### 4. Выводы

Представлен обзор и анализ существующих модификаций генетических алгоритмов, показаны их основные достоинства и недостатки. Предложена классификация основных направлений модификации генетического алгоритма. Рассмотрены современные направления исследований ГА для повышения их эффективности.

Представлена методика тестирования генетических алгоритмов. В указанной методике возможно использование трёх критериев: трудоёмкости метода, достоверности и комплексного критерия информационной сложности модификации ГА по отношению к простому генетическому алгоритму. По методике тестирования генетических алгоритмов разработан программный комплекс для удобства проведения тестирования.

Для тестирования методики тестирования генетических алгоритмов были выбраны распространённые аналитические функции различной сложности и размерности, которые часто используются для тестирования генетических алгоритмов. В качестве набора тестируемых алгоритмов отобраны двадцать две модификации простого генетического алгоритма, сходные по архитектуре, сложности реализации и потреблению ресурсов ЭВМ.

Получено, что самое высокое значение вероятности нахождения глобально-го экстремума среди протестированных алгоритмов – у генетического алгоритма с двухточечным кроссинговером, регуляцией вероятностей генетических операторов, оператором мутации с равномерным распределением потомков.

### Список литературы

1. Bremermann, H. J. Search by Evolution in Biophysics and Cybernetic Systems [Text] / H. J. Bremermann, M. Rogson, S. Salaff, Washington DC: Spartan Books, 1965. – P. 157–167.
2. Bremermann, H. J. Global Properties of Evolution Processes in Natural Automata and Useful Simulations [Text] / H. J. Bremermann, M. Rogson, S. Salaff, Washington DC: Spartan Books, 1966. – P. 3–41.
3. Brindle, A. Genetic algorithms for function optimization [Text] : PhD thesis, University of Alberta, 1981.
4. Goldberg, D. E. Sizing populations for serial and parallel genetic algorithms [Text] : Proc. of the 3rd International conference on Genetic Algorithms and Their Applications / CA: San Mateo, 1989. – P. 70-79.
5. De Jong, K. A. Analysis of the behavior of class of genetic adaptive systems [Text] / University of Michigan Press, Ann Arbor, 1975. – P. 512.
6. Мезенцева, О. С. Разработка модифицированного генетического алгоритма с использованием нечеткого оператора отбора [Текст] / О. С. Мезенцева, Е. В. Маликов, Инфотелекоммуникационные технологии в науке, производстве и образовании // Первая междунар. науч.-техн. конф. – Ставрополь: СевКавГТУ, 2004.
7. Воронкин, Р. А. Математическое моделирование поиска глобальных экстремумов для решения задач адаптации на базе генетических алгоритмов [Текст] : дис. канд. техн. наук : 05.13.18 / Воронкин Роман Александрович – Ставрополь, 2004. – 237 с.
8. Применение распределённого генетического алгоритма при решении задачи об упаковке в контейнеры [Текст] / Ю. А. Бюргер, В. И. Гнатюк, В. И. Литвиненко, А. А. Ткачук // Перспективные информационные технологии и интеллектуальные системы, – 2003. – №1. – С. 11–15.
9. Курейчик, В. М. Перспективные архитектуры генетического поиска [Текст] / В. М. Курейчик // Перспективные информационные технологии и интеллектуальные системы – 2000. – № 1. – С. 58–60.
10. Полупанов, А. А. Адаптивная архитектура генетического поиска [Текст] / А. А. Полупанов // Перспективные информационные технологии и интеллектуальные системы – 2002. – № 3. – С. 49–55.
11. Полупанов, А. А., Повышение эффективности генетического поиска [Текст] / А. А. Полупанов // Тез. докл. VI Всерос. конф. студентов и аспирантов. – Таганрог: ТРТУ, 2002. – С.89.
12. Wang, G. On the Research and Implementation of Distributes Adaptive Genetic Algorithm [Text] / Master's thesis, Beijing University of Aeronautics and Astronautics, Beijing, March, 1995.
13. Еремеев, А. В. Разработка и анализ генетических и гибридных алгоритмов для решения задач дискретной оптимизации [Текст]: дис. канд. физ.-мат. наук : 05.13.17 / Еремеев Антон Валентинович. – Омск, 2000. – 119 с.
14. Емельянов, В. В. Теория и практика эволюционного моделирования [Текст] / В. В. Емельянов, В. В. Курейчик, В. М. Курейчик – М.: Физматлит, 2003. – 432 с.
15. Rudolph, G. Finite Markov Chain Results in Evolutionary Computation [Text] / G. Rudolph // Fundamenta Informaticae (Journal), 1998. – P. 67–89.

16. Michalewicz, Z. Genetic Algorithms + Data Structures = Evolutionary Programs [Text] / Z. Michalewicz // NY: Springer-Verlag, 1992. – P. 388.

17. Holland, J. H. Adaptation in Natural and Artificial Systems [Text] / J. H. Holland // University of Michigan Press, Ann Arbor, 1975. – 211 p.

18. Muhlenbein, H. Parallel genetic algorithm, population dynamics and combinatorial optimization [Text] / H. Muhlenbein // Proc. Third Inter. Conf. Genetic Alg. San Mateo: Morgan Kaufman, 1989. – P. 416–421.

19. Петров, Ю. Ю. Разработка и исследование математической модели генетического алгоритма для применения в технических системах [Текст]: дис. канд. тех. наук : 05.13.18 / Петров Юрий Юрьевич, – Ставрополь, 2008. – 284 с.

Поступила в редакцию 24.11.2015

## **Огляд і аналіз існуючих модифікацій генетичних алгоритмів**

Представлені огляд і аналіз існуючих модифікацій генетичних алгоритмів, показані їх основні переваги і недоліки. Запропоновано класифікація основних напрямків модифікації генетичного алгоритму, а також методика тестування генетичних алгоритмів. Для тестування використані дев'ять завдань різної складності і розмірності. В якості алгоритмів для тестування обрані двадцять дві модифікації простого генетичного алгоритму, подібні за архітектурі, складності реалізації та споживанню ресурсів. У результаті дослідження визначені перспективні напрямки модифікацій генетичних алгоритмів з метою підвищення їх ефективності.

**Ключові слова:** генетичні алгоритми, модифікація, тестування, інформаційна складність, трудомісткість методу, достовірність відповіді.

## **Using a Circular Queue in Discrete Event-Driven Method of Molecular Dynamics**

Provides an overview and analysis of existing versions of genetic algorithms, showing their main advantages and disadvantages. The classification of the main directions of the modification of the genetic algorithm, and the testing technique of genetic algorithms. For testing we used nine tasks of varying complexity and dimension. As test algorithms selected twenty-two modifications of simple genetic algorithm, similar in architecture, implementation complexity and resource consumption. The study identified promising areas for modifying the genetic algorithms in order to improve their effectiveness.

**Keywords:** genetic algorithms, modifying, testing, information complexity, the complexity of the method, the accuracy of the response.