

УДК 004.8

**ГИБРИДНЫЙ ОПТИМИЗАЦИОННЫЙ
АЛГОРИТМ ГРИФОВ НА ОСНОВЕ
МЕХАНИЗМОВ РОЕВОГО ИНТЕЛЛЕКТА**Частикова Вера Аркадьевна
к.т.н., доцентОстапов Дмитрий Сергеевич
аспирант
*Кубанский государственный технологический
университет, Краснодар, Россия*

В статье предложен гибридный алгоритм поиска гриффов с минимальным числом входных параметров для оптимизации сложных функций, меняющихся с течением времени. Проведены исследования эффективности его работы и сравнительный анализ с некоторыми другими алгоритмами

Ключевые слова: ГИБРИДНЫЙ ОПТИМИЗАЦИОННЫЙ АЛГОРИТМ ГРИФОВ НА ОСНОВЕ МЕХАНИЗМОВ РОЕВОГО ИНТЕЛЛЕКТА, РОЕВОЙ ИНТЕЛЛЕКТ, АЛГОРИТМ РОЯ ЧАСТИЦ, АЛГОРИТМ ПЧЕЛИНОЙ КОЛОНИИ, АЛГОРИТМ ДВИЖЕНИЯ КОСЯКОВ РЫБ, ГЛОБАЛЬНАЯ ОПТИМИЗАЦИЯ

UDC 004.8

**HYBRID OPTIMIZING GRIFFON-VULTURE
ALGORITHM BASED ON SWARM
INTELLIGENCE MECHANISMS**Chastikova Vera Arkadyevna
Cand.Tech.Sci., associate professorOstapov Dmitriy Sergeevich
postgraduate student
*Kuban State Technological University, Krasnodar,
Russia*

Griffon-vultures with input parameters minimal value for compound functions optimization that change during the time searching hybrid algorithm offered in this article. Researches of its efficiency and comparing analysis with some other systems have been performed

Keywords: HYBRID OPTIMIZING GRIFFON-VULTURE ALGORITHM BASED ON SWARM INTELLIGENCE MECHANISMS, SWARM INTELLIGENCE, PARTICLE SWARM OPTIMIZATION, BEE COLONY OPTIMIZATION, FISH SCHOOL SEARCH ALGORITHM, GLOBAL OPTIMIZATION

Целью задач глобальной оптимизации является нахождение точки в пространстве, в которой заданная функция принимает оптимальное значение. Для поиска оптимальных решений могут использоваться как классические методы (метод Монте-Карло, последовательного перебора, градиентного спуска и т.д.), так и эвристические алгоритмы [6], к которым относятся и алгоритмы роевого интеллекта.

Все алгоритмы роевого интеллекта, направленные на поиск глобального экстремума функции имеют ряд параметров, значения которых необходимо подбирать самостоятельно для каждой рассматриваемой задачи[2,3,4]. Например, в алгоритме роя частиц, который основан на поведении некоторых частиц в пространстве, скорость каждой частицы рассчитывается по следующей формуле:

$$v_{i,t+1} = w \cdot v_{i,t} + \varphi_p \cdot r_p \cdot (p_i - x_{i,t}) + \varphi_g \cdot r_g \cdot (g_i - x_{i,t}) \quad (1)$$

Коэффициенты w , φ_p , φ_g подбираются для каждой конкретной задачи, r_p и r_g – случайные числа в диапазоне $(0;1)$, $x_{i,t}$ – координата t -ой частицы на i -ой итерации, p_i – координата лучшего решения, найденного частицей, g_i – координата лучшего решения, найденного роем частиц. Если значения параметров будут подобраны неверно, результат работы метода может оказаться абсолютно непредсказуемым. Данное утверждение можно проверить с использованием поиска минимума функции Растригина двух переменных ($f(x,y) = 20 + x^2 + y^2 - 10(\cos(2\pi x) + \cos(2\pi y))$), поверхность которой изображена на рисунке 1.

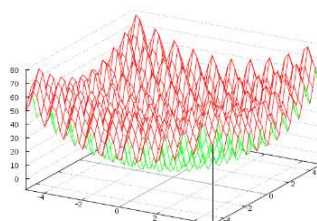


Рисунок 1 – Поверхность функции Растригина

Как известно, данная функция имеет глобальный минимум в начале координат. В таблице 1 представлен результат работы метода роя частиц, направленного на поиск минимума функции Растригина двух переменных после 100 итераций. Параметры w , φ_p , φ_g были выбраны случайным образом.

Таблица 1 – Результат работы алгоритма роя частиц

W	φ_p	φ_g	Наилучший результат
0,749	1,49	1,49	6,27512335427127E-08
0,749	2	5	2,62744870708719
0,749	1	1	9,84783810054068E-10
0,749	5	2	22,349890810727
2,749	1	1	24,0612774040452

Как видно из таблицы 1, в ряде случаев был найден глобальный минимум функции ≈ 0 . В случае некорректного подбора параметров, метод роя частиц не смог обнаружить оптимум. Таким образом, подобные алгоритмы при определенных условиях не подходят для поиска глобального экстремума функций (например, функций, меняющихся с течением времени) в связи с необходимостью постоянной настройки параметров. Целью данного исследования является разработка наиболее эффективного алгоритма для проблем поиска оптимума функций, меняющихся со временем.

Предложенный гибридный алгоритм поиска «грифов» имеет много общего с известными ранее алгоритмами роя частиц, движения косяков рыб, Монте-Карло и алгоритмом пчелиной колонии.

В методе роя частиц новая координата определяется по следующей формуле

$$x_{t+1} = x_t + v_{t+1} \quad (2)$$

В алгоритме движения косяков рыб поиск глобального экстремума состоит из несколько стадий: миграция агентов к источнику пищи, индивидуальная, инстинктивно-коллективная и коллективно-волевая стадии плавания. В индивидуальной стадии плавания частица перемещается в следующую точку пространства в случае, если значение функции в ней лучше, чем в предыдущей. Новая координата определяется следующим образом:

$$x_{t+1} = x_t + \text{Random}(0;1) \cdot \text{step} \quad (3)$$

В алгоритме Монте-Карло случайным образом генерируются точки в пространстве до тех пор, пока не будет найдена точка с учетом допустимой погрешности, или не сработает критерий останова.

Грифы – это птицы, летающие на большой высоте в поисках пищи, которую находят с помощью зрения или обоняния. Если один из них увидит добычу, он спускается вниз, а остальные птицы летят туда же [1].

Поведение грифов подпадает под законы роевого интеллекта: взаимодействуя друг с другом, они осуществляют поиск пищи. По этой причине разработанный алгоритм был назван «Гибридный оптимизационный алгоритм грифов на основе механизмов роевого интеллекта». Он состоит из рассмотренных ниже этапов.

1. Создаётся рой птиц внутри заданного пространства. Значение функции вычисляется для каждого агента. Рой состоит из 2 типов птиц: Attack Grif, которые увидев добычу, сразу летят к ней, и разведчиков, кружащих в небе в поисках пищи. Пусть в рое находится M Attack Grif и N разведчиков (подобно алгоритму пчелиной колонии[2]). Чем больше значение M , тем более точное решение находит метод за определённое число итераций в рассматриваемой области. Увеличение значения N влияет на успешность поиска более точных решений функции внутри или вне первоначальной области пространства.

2. Происходит сортировка стаи птиц по ухудшению найденных значений целевой функции. Первые M агентов в рое становятся Attack Grif, а последние N – разведчиками, вне зависимости от того, к какому типу они принадлежали ранее. Далее происходят перемещения M птиц. В зависимости от того какое значение функции было найдено каждой птицей, её перемещение будет происходить по разным алгоритмам. M грифов следует разделить на 2 части с учётом успешности найденного значения функции. Алгоритм перемещения M грифов описан в таблице 2.

Разделение птиц на 2 части было выполнено с определенной целью. Посредством работы лучшей половины грифов согласно формуле (6) достигается сужение пространства поиска. С помощью худшей половины согласно формулам (4,5) происходит как сужение, так и расширение (в зависимости от знака числа, возвращаемого функцией $\text{Random}(-1,1)$) области поиска решений. Подобно алгоритму движения косяков рыб

перемещение агентов происходит только в случае, если значение функции в новой точке лучше, чем в предыдущей.

Таблица 2 – Алгоритм перемещения M грифов

Худшая половина птиц	Лучшая половина птиц
<p>Подобно (3) из алгоритма движения косяков рыб:</p> $x_i = x_i + \text{Random}(-1;1)(x_i - x_{i+1})$ <p>(4)</p> <p>Птица переместится в следующую точку только в том случае, если значение функции в ней будет лучше, чем в предыдущей.</p> <p>Последний гриф из худшей половины будет перемещаться по следующему алгоритму:</p> $x_i^+ = \text{Random}(-1;1)(x_i^+ + x_r)$ <p>(5)</p> <p>где x_r - птица, выбранная случайным образом из M-1 первых птиц.</p>	<p>Для первой птицы из лучшей половины новая координата</p> $x_1^* = \frac{x_1 + x_2}{2}$ <p>Для второй:</p> $x_2^* = \frac{x_2 + x_3}{2}$ <p>Для n-ой:</p> $x_n^* = \frac{x_n + x_{n+1}}{2}$ <p>(6)</p> <p>x_n^* - новая координата n-ой птицы из лучшей половины.</p> <p>Агент переместится в следующую точку только в том случае, если значение функции в ней будет лучше, чем в предыдущей. В случае, если значение функции в новой точке хуже, чем в предыдущей, будет осуществлена попытка перемещения грифа по формуле (4).</p>

3. Разведчики осуществляют дополнительный поиск решений. Он может происходить по одному из двух описанных ниже алгоритмов в зависимости от условий задачи.

3.1. Если заранее известно, что глобальный экстремум располагается внутри области, в которой находились грифы на 1 шаге, то разведчики также будут посылаться в эту же область.

3.2. Если в рамках задачи точно неизвестно, где находится глобальный экстремум, необходимо использовать алгоритм расширения области поиска, который предполагает следующие шаги.

Пусть по координате x начальная область поиска находится от -5 до 5, и одна из сторон части пространства, представленной в виде n-мерного прямоугольного параллелепипеда, имеет длину $a_1 = 10$. Область поиска

необходимо равномерно расширить до величины a_2 , то есть следует к максимальной координате отрезка прибавить положительное число и такое же значение вычесть из минимальной координаты. Схематично процесс расширения представлен на рисунке 2.

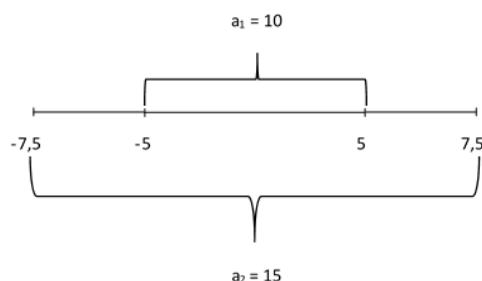


Рисунок 2 – Расширение области поиска

Для получения максимальной координаты a_2 необходимо к максимальной координате a_1 прибавить $a_1 \cdot k$, где a_1 – длина стороны a_1 , k – коэффициент расширения. Например, если $k = 0.25$, то сторона увеличится на 50%. Коэффициент k может быть подобран самостоятельно в рамках конкретной задачи; он влияет только на скорость поиска. На рисунке 3 представлены области до и после расширения.

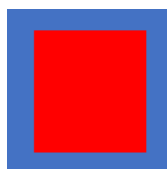


Рисунок 3 – Области до и после расширения

На рисунке 3 красным цветом представлена область, в которой изначально создавались птицы; она закрашена красным цветом. Далее на следующей итерации было проведено расширение, и разведчики производят поиск решений уже внутри области, которая закрашена синим цветом. Увеличение области поиска происходит до тех пор, пока оно не достигнет максимальных значений, заданных при создании роя птиц, после чего её необходимо сузить до первоначального значения и продолжить

работу алгоритма расширений сначала. Схема алгоритма поиска грифов приведена на рисунке 4.

Для исследования эффективности предложенного алгоритма поиска грифов был проведен сравнительный анализ работы двух его модификаций (с расширением и без расширения), метода роя частиц, алгоритма движения косяков рыб и алгоритма Монте-Карло на примере задачи глобальной оптимизации: поиска минимума функции Растригина, функции $f(x_n) = \sum x_n^2$ и функции Розенброка $f(x,y) = (1-x)^2 + 100(y-x^2)^2$, поверхности которых представлены соответственно на рисунках 1, 5 и 6. Область определения функций: $x,y \in [-20; 20]$. Для первых двух функций в алгоритме поиска грифов исследование осуществлялось при наличии 180 AttackGrif и 20 разведчиков.

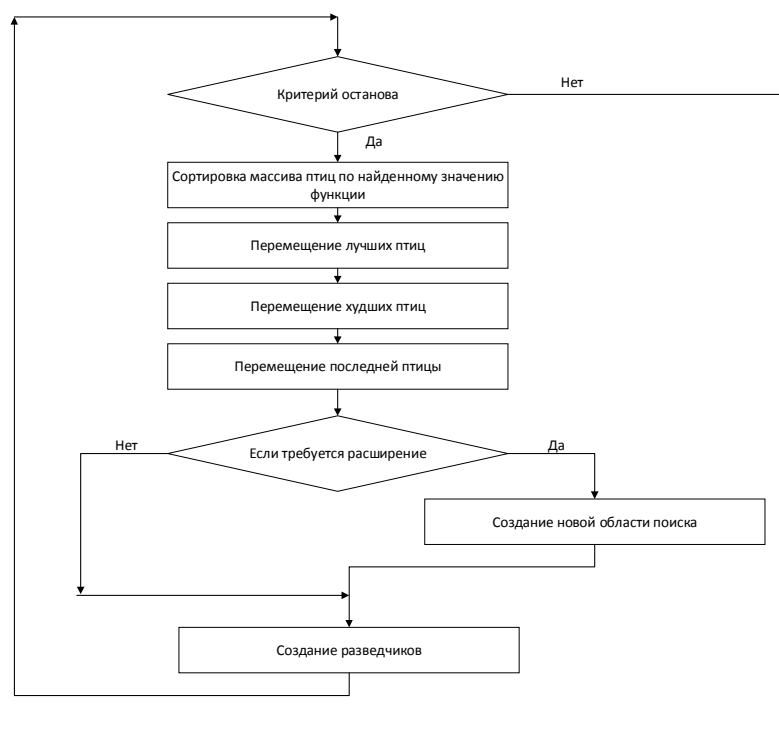


Рисунок 4 – Схема алгоритма поиска грифов

В рое частиц находилось 200 агентов. Начальные значения скоростей в рое частиц: $v_x, v_y \in [-5; 5]$, $\phi_p = \phi_g = 1.49$, $w = 0.749$.

В зависимости от точности решения, которое необходимо найти с использованием алгоритма косяков рыб и анализируемой функции, использовалось число рыб в аквариуме от 100 до 15000, максимальное число итераций от 100 до 7000, начальный шаг от 0,5 до 0,01, конечный шаг от 0,1 до 0,000000001.

Для всех трёх исследуемых алгоритмов агенты создавались в области $x, y \in [-5; 5]$.

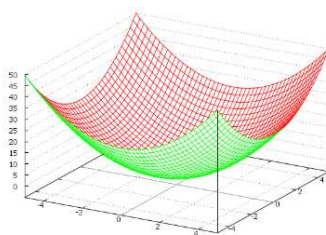


Рисунок 5 – Поверхность функции $f(x,y) = x^2 + y^2$

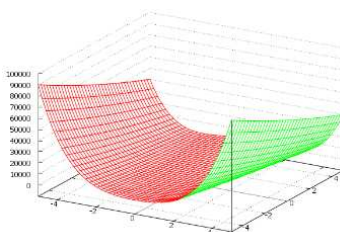


Рисунок 6 – Поверхность функции Розенброка

Результаты работы алгоритмов для поиска глобального минимума функции Растригина двух переменных в зависимости от требуемой точности приведены на рисунке 7.

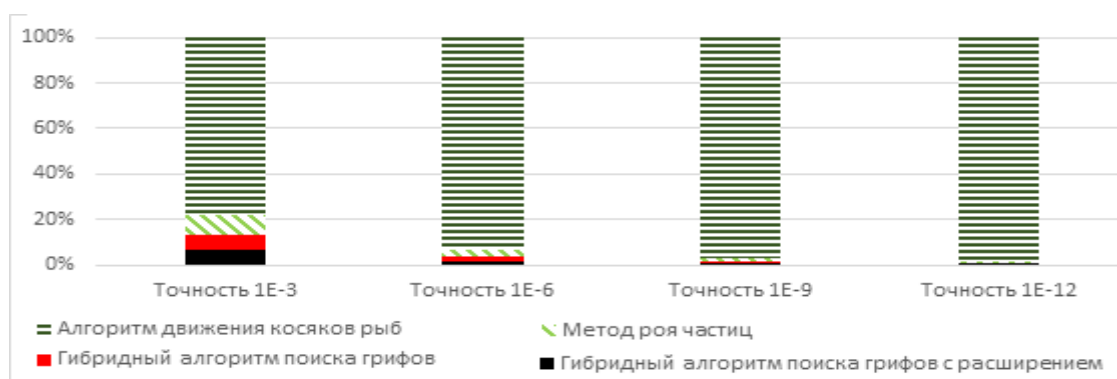


Рисунок7 – Соотношение времени поиска глобального экстремума функции Растригина 2 переменных

Как видно из рисунка 7, обе модификации алгоритма поиска грифов работают эффективнее, чем алгоритм роя частиц. Тем не менее, использование механизма расширения замедляет работу алгоритма примерно на 3%. Гибридный алгоритм поиска грифов работает на $\approx 56,61\%$ и $11920,53\%$ быстрее метода роя частиц и алгоритма движения косяков рыб соответственно при поиске экстремума функции Растригина двух переменных.

На рисунке 8 представлено время поиска глобального минимума функции $f(x,y) = x^2 + y^2$.

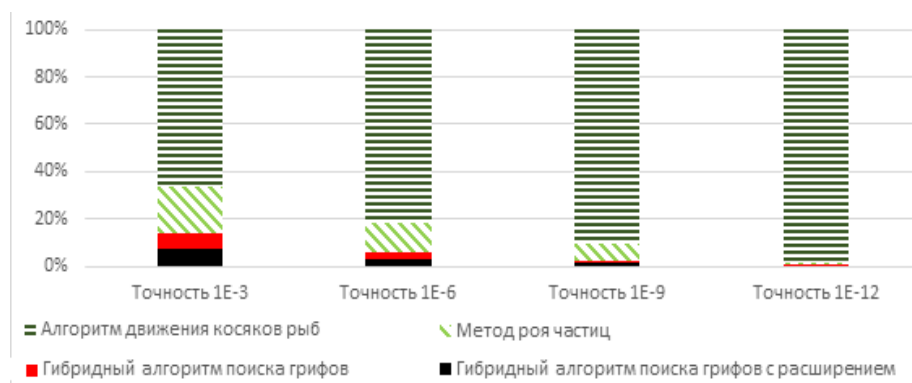


Рисунок 8 - Соотношение времени поиска глобального экстремума функции $f(x,y) = x^2 + y^2$

Скорость поиска гибридного алгоритма грифов на $\approx 297,84\%$ и $20303,41\%$ больше скорости поиска метода роя частиц и алгоритма движения косяков рыб соответственно.

На рисунке 9 представлено время поиска минимума функции Розенброка. Данная функция является достаточно сложной, поэтому в алгоритме поиска грифов для большей точности число Attack Grif следует увеличить до 800 и число разведчиков до 300.

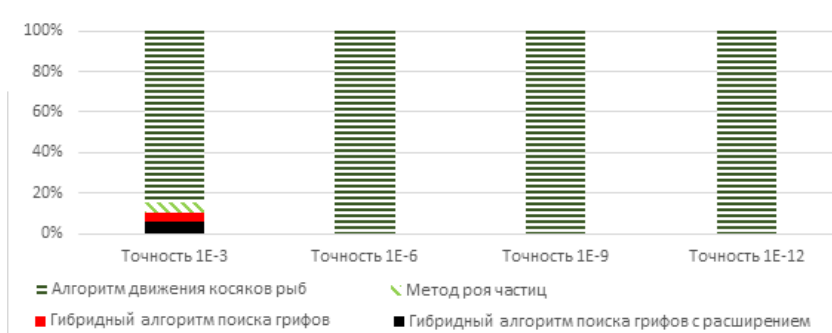


Рисунок 9 - Соотношение времени поиска глобального экстремума функции Розенброка

Как видно из рисунка 9, гибридный алгоритм поиска грифов работает на $\approx 425,16\%$ медленнее роя частиц и на $71979,43\%$ быстрее алгоритма движения косяков рыб соответственно.

При проведении предыдущих экспериментов глобальный экстремум находился внутри области, в которой на первом шаге создавались агенты. Для проверки эффективности поиска глобального экстремума вне заданной области следует сформировать исходный рой агентов таким образом, чтобы оптимум функции изначально не располагался внутри рассматриваемой части пространства. На рисунке 10 представлены данные поиска глобального экстремума функции Растригина с точностью до 10^{-6} при начальном задании агентов в области $x, y \in [-10; -5]$ и различных комбинациях значений Attack Grif, разведчиков и коэффициента расширения.



Рисунок 10 - Время поиска глобального экстремума функции Растригина двух переменных

Метод Монте-Карло относится к классическим алгоритмам оптимизации. Эффективность поиска с использованием метода Монте-Карло сильно зависит от точности решения, которое необходимо найти, и имеет слабую зависимость от вида анализируемой функции. Среднее время, затраченное на поиск глобального минимума с точностью до $1E-3$ и $1E-6$ рассмотренных выше функций, в ≈ 175 и 810493 раз соответственно больше времени, затраченного алгоритмом поиска грифов. Время поиска решений с точностью до $1E-9$ и $1E-12$ для всех рассмотренных функций превысило несколько часов.

Исходя из проведённых экспериментов, можно сделать следующие выводы:

1. Гибридный алгоритм поиска грифов имеет минимальное число входных параметров, влияющих на успешность нахождения глобального экстремума функции. При создании роя птиц необходимо указывать количество Attack Grif и разведчиков в стае и коэффициент расширения. Однако, значения входных данных влияют только на скорость поиска решения.

2. Разработанный алгоритм показал свою эффективность для поиска глобального экстремума функций, меняющихся с течением времени и имеющих достаточно сложный ландшафт.

Одним из возможных способов применения гибридного алгоритма поиска грифов является минимизация ошибки, которая возникает при настройке нейронной сети [5]. Благодаря отсутствию параметров, влияющих на успешность нахождения решения, предложенный алгоритм может применяться для анализа данных, меняющихся с течением времени, например, для анализа сетевого трафика на наличие DDoS-атак, которые являются серьёзной угрозой для сетевой безопасности [7].

Литература

1. Ильичев В.Д., Михеев А.В. Жизнь животных. Том шестой. Птицы. М.: 1986, Вып. 2, с. 120-123
2. А.П. Карпенко. Популяционные алгоритмы глобальной поисковой оптимизации. Обзор новых и малоизвестных алгоритмов // Информационные технологии. 2012. № 7, с. 1-32
3. A Novel Search Algorithm based on Fish School Behavior / Carmelo J. A. BastosFilho, Fernando B. de Lima Neto and other //International Conference on Systems, Man and Cybernetics. 2008. P. 2646-2651
4. Курейчик В.М., Кажаров А.А. Использование роевого интеллекта в решении NP-трудных задач // Известия Южного федерального университета. Технические науки. 2011. №7, с. 30-36
5. Малыгина М.П. Оценка эффективности гибридизации интеллектуальных методов на примере нейросетевой экспертной системы на основе прецедентов / Малыгина М.П., Бегман Ю.В. //Научный журнал КубГАУ [Электронный ресурс]. - Краснодар: КубГАУ, 2013. - № 86(02). - Режим доступа: <http://ej.kubagro.ru/2013/02/pdf/24.pdf>.
6. Частиков А.П., Дедкова Т.Г., Алешин А.В. Системы искусственного интеллекта. От теории к практике. Учебное пособие. - Краснодар, 1998. – 166 с.
7. Частикова В.А.Комплексная кольцевая система защиты программного обеспечения от нелицензионного использования / Частикова В.А., Остапов Д.С. //Научный журнал КубГАУ [Электронный ресурс]. - Краснодар: КубГАУ, 2012. - № 82(08). - Режим доступа: <http://ej.kubagro.ru/2012/08/pdf/47.pdf>

Referenses

1. Il'ichev V.D., Miheev A.V. Zhizn' zhiivotnyh. Tom shestoj. Pticy. M.: 1986, Vyp. 2,s. 120-123
2. A.P. Karpenko. Populjacionnyje algoritmy global'noj poiskovoj optimizacii. Obzor novyh i maloizvestnyh algoritmov // Informacionnyje tehnologii. 2012. № 7,s. 1-32
3. A Novel Search Algorithm based on Fish School Behavior / Carmelo J. A. BastosFilho, Fernando B. de Lima Neto and other //International Conference on Systems, Man and Cybernetics. 2008. P. 2646-2651
4. Kurejchik V.M., Kazharov A.A. Ispol'zovanie roevogo intellekta v reshenii NP-trudnyh zadach // Izvestija Juzhnogo federal'nogo universiteta. Tehniceskie nauki. 2011. №7, s. 30-36
5. Malyhina M.P. Ocenka jeffektivnosti gibridizacii intellektual'nyh metodov na primere nejrosetevoj jekspertnoj sistemy na osnove precedentov / Malyhina M.P., Begman Ju.V. //Nauchnyj zhurnal KubGAU [Jelektronnyj resurs]. - Krasnodar: KubGAU, 2013. - № 86(02). - Rezhim dostupa: <http://ej.kubagro.ru/2013/02/pdf/24.pdf>.
6. Chastikov A.P., Dedkova T.G., Aleshin A.V. Sistemy iskusstvennogo intellekta. Ot teorii k praktike. Uchebnoe posobie. - Krasnodar, 1998. – 166 с.
7. Chastikova V.A.Kompleksnaja kol'cevaja sistema zashhity programmno obespechenija ot nelicenzionnogo ispol'zovanija / Chastikova V.A., Ostapov D.S. //Nauchnyj zhurnal KubGAU [Jelektronnyj resurs]. - Krasnodar: KubGAU, 2012. - № 82(08). - Rezhim dostupa: <http://ej.kubagro.ru/2012/08/pdf/47.pdf>