**Приложение 2. Образец оформления статьи (доклада)**

**А.П. Карпенко**

*МГТУ им. Н.Э. Баумана, Москва, Россия*

УДК 519.6

**ИССЛЕДОВАНИЕ ЭФФЕКТИВНОСТИ АЛГОРИТМА ГЛОБАЛЬНОЙ ОПТИМИЗАЦИИ, ВДОХНОВЛЕННОГО НЕКОТОРЫМИ АСПЕКТАМИ ПОВЕДЕНИЯ ТАРАКАНОВ**

Ключевые слова: *глобальная безусловная оптимизация, популяционный алгоритм, алгоритм оптимизации роем частиц, функция Растригина, функция Экли.*

*Рассматриваем оптимизационный алгоритм нашествия тараканов (Roach Infestation Optimization, RIO), который относится к классу популяционных алгоритмов, вдохновленных живой природой. Цель работы состоит в программной реализации и исследовании эффективности алгоритма RIO для известных сложных мультимодальных тестовых функций Растригина и Экли. Особенностью исследования является поиск глобального экстремума (минимума) указанных функций в широкой области пространства поиска, в которой число локальных минимумов этих функций чрезвычайно велико.*

**Введение.** В науке и технике часто возникают задачи глобальной параметрической оптимизации, отличительной особенностью которых является наличие, часто, большого или очень большого числа локально-оптимальных решений. В силу высокой практической значимости этих задач разработано большое число алгоритмы их решения. Вслед за [1], разделяем алгоритмы глобальной оптимизации на классические и не классические (популяционные). Популяционные алгоритмы (П-алгоритмы) часто обеспечивают более высокую эффективность (например, вероятность локализации глобального экстремума целевой функции) при решении сложных многомерных мультимодальных задач глобальной оптимизации.

Известно несколько классификаций П-алгоритмов глобальной оптимизации. Обычно выделяют следующие классы этих алгоритмов: эволюционные алгоритмы, включая генетические; П-алгоритмы, вдохновленные живой природой; алгоритмы, вдохновленные неживой природой; алгоритмы, инспирированные человеческим обществом; прочие алгоритмы [1].

Рассматриваемый в данной работе оптимизационный алгоритм нашествия тараканов (Roach Infestation Optimization, RIO) [2] относится к классу алгоритмов, вдохновленных живой природой. Этому классу принадлежит большое число «кластеров» П-алгоритмов: бактериальные алгоритмы; алгоритмы, вдохновленные насекомыми, млекопитающими, птицами, земноводными, рыбами, растениями и т.д. [3]. Алгоритм RIO входит в кластер П-алгоритмов, инспирированных поведением в природе насекомых. Кроме алгоритма RIO этому кластеру принадлежат муравьиные, пчелиные, светлячковые и некоторые другие алгоритмы [3].

Алгоритм RIO разработан на основе наблюдений за поведением тараканов [4-8] и его можно считать глубокой модификацией широко известного и одного из наиболее эффективных оптимизационных алгоритмов роя частиц (Particle swarm optimization, PSO) [9, 10].

RIO моделирует три следующих аспекта поведения тараканов.

Find Darkness. Тараканы ищут самое «темное» место в пространство поиска. Полагаем, что степень темноты в рассматриваемой точке пространства поиска прямо пропорциональна значению фитнес-функция в этой точке.

Find Friends. Тараканы вероятностно ищут соседних тараканов (тараканов своей группы). В качестве метрики близости используется евклидова метрика пространства .

Find Food. Тараканы периодически становятся голодными и в поисках пищи покидают достигнутые темные места и свои группы.

В силу высокой эффективности алгоритма PSO для широкого круга задач глобальной оптимизации, особый интерес представляет исследование данной модификации этого алгоритма. Цель работы состоит в программной реализации и исследовании эффективности алгоритма RIO для известных сложных мультимодальных тестовых функций Ратригина и Экли. Особенностью исследования является поиск глобального экстремума (минимума) указанных функций в широкой области пространства поиска, в которой число локальных минимумов этих функций чрезвычайно велико

**1. Постановка задачи.** Рассматриваем детерминированную задачу глобальной безусловной минимизации

, (1)

где  ‑ размерность вектора варьируемых параметров ;  ‑ целевая функция со значениями в пространстве ; ,  ‑ искомые оптимальные решение и значение целевой функции соответственно. Полагаем, что поиск решения задачи (1) (то есть инициализация популяции) начинается в -мерном параллелепипеде , где  и неравенства понимаются покомпонентно. Полагаем, что фитнесс-функция  П-алгоритма также подлежит минимизации. Найденные П-алгоритмом приближенные значения величины  и компонентов вектора  обозначаем ,  соответственно.

Отметим, что задача глобальной условной минимизации может быть сведена к аналогичной задаче безусловной оптимизации методом штрафных функций [1].

Примем еще следующие обозначения:  ‑ число особей (тараканов) в популяции;  ‑ -мерные векторы, определяющие параллелепипед  допустимых «скоростей» особей;  ‑ вещественные константы, задающие веса составляющих «скорости»;  ‑ целочисленный порог уровня «голода» тараканов;  ‑ максимально допустимое число итераций алгоритма;  ‑ целочисленное приращение уровня «голода»;  ‑ вероятности, определяющие число «соседей» каждого из тараканов (см. ниже).

**2. Схема алгоритма.**

1)*Инициализация*

1.1) Инициализация пользователем свободных параметров алгоритма , рекомендуемые значения которых имеют следующие значения: ; ; ;  ; .

1.2) Инициализация начальных положений  и «скоростей»  особей популяции *S*. Равномерно случайно распределяем особей  в области , а их скорости ‑ в области .

1.3) Инициализация начального уровня *голода* тараканов , . Полагаем величины  равномерно случайно распределенными в интервале .

2) *Эволюция популяции* (схема одной итерации алгоритма)

2.1 Вычисляем значения элементов -матрицы , имеющих смысл текущих евклидовых расстояний  между особями популяции: .

2.2) С использованием матрицы  вычисляем текущее среднее расстояние между двумя особями популяции

.

2.3) Для каждой из особей  выполняем следующие действия.

а) Среди текущих следов особи  выбираем наилучшее положение  такое, что

.

б) Определяем множество текущих соседей  особи  (*группу* таракана ) в пространстве  по формуле

. (2)

в) Среди следов текущих соседей  особи  с вероятностью  выбираем наилучше положение :

.

Здесь вероятность  определяет формула

 (3)

г) Если текущий уровень голода таракана  не превышает заданный уровень , то есть если , то применяем рандомизирующий трех шаговый оператор перемещения особи 

; (4)

.

д) Если уровень голода того же таракана превышает , то есть если  то выполняем следующие действия:

‑ перемещаем особь  в случайную точку **, созданную в параллелепипеде *П* по правилам инициализации начальной популяции;

‑ полагаем .

2.4) Повышаем уровень голода таракана  по правилу .

3) *Завершение эволюционного процесса*.

Условием окончания поиска является достижение числа итераций эволюционного процесса, равного .

Отметим, что поскольку в определении множество соседей особи  используется ограничение , в это множество не входит особь  (см. (2)). Отсюда следует, что наилучшее положение  не может совпадать с наилучшим положением .



Поведения *Find Darkness*, *Find Friends* моделируют формула (4) и первые два слагаемых формулы (5) или третье слагаемое формулы (5) соответственно. Правило поведения *Find Food* формализует п. 2.3д схемы алгоритма. Таракан  перемещается в пространстве поиска по правилам *Find Darkness*, *Find Friends* если его текущий уровень голода не превышает заданный уровень . В противном случае этот таракан перемещается по правилам *Find Food.*



**Программное обеспечение и организация вычислительных экспериментов.** Алгоритм нашествия тараканов реализован на языке программирования Python и IDE PyCharm. Каждый из тараканов является объектом класса *Cockroach*, а рой ‑ объектом класса *Swarm*. Использованы указанные выше рекомендованные значения свободных параметров алгоритма.

Для тестирования алгоритма и разработанного программного обеспечения использована сферическая функция, минимум которой достигается в точке и равен, очевидно, нулю.



Исследование эффективности алгоритма RIO выполнено на двух тестовых функциях с известными точными положениями  и значениями  глобального минимума [1]:

‑ функции Растригина;

‑ функция Экли.

Во всех случаях инициализация начальной популяции осуществлена в кубе, так что  (см. ниже).

Принято, что фитнесс-функция совпадает с целевой функцией, то есть что имеет место равенство . Критерием окончания поиска является стагнация вычислительного процесса: если на протяжении  итераций значение фитнесс-функции улучшается менее чем на , то поиск прекращается. Используем значение величины , равное 150.

Вычислительные эксперименты выполнены для размерностей пространства поиска, равных 2,4,8,16,32 и 64. Для каждого из экспериментов применен метод мультистарта с числом запусков алгоритма, равным .

Рассматриваем следующие индикаторы эффективности алгоритма:

1) среднее число итераций алгоритма ;

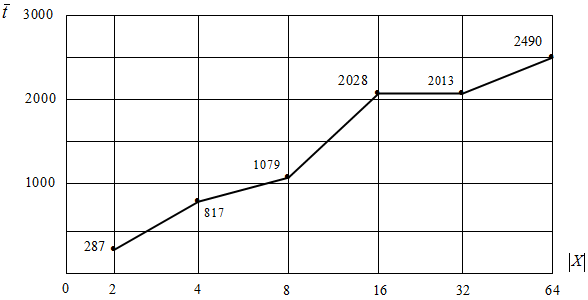
2) оценка математического ожидания  достигнутой абсолютной точности в пространстве значений целевой функции ‑ среднее по мультистарту значение величины ;

3) аналогичная оценка математического ожидания  достигнутой значения абсолютной точности в пространстве поиска – оценка среднего по мультистарту значения величины ;

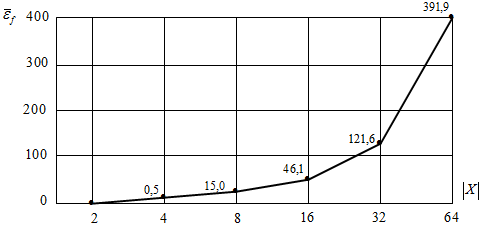
4) оценка вероятности локализации глобального экстремума , где  ‑ число успешных стартов алгоритма, то есть стартов, в которых был выполнен критерий локализации глобального экстремума.

Здесь  ‑ заданная константа;  – длина ребра начального куба  с центром в точке . Для всех рассматриваемых размерностей вектора варьируемых параметров  эта длина принята равной *l=*200, то есть компоненты  векторов  во всех случаях выбирались так, чтобы это равенство было справедливым.

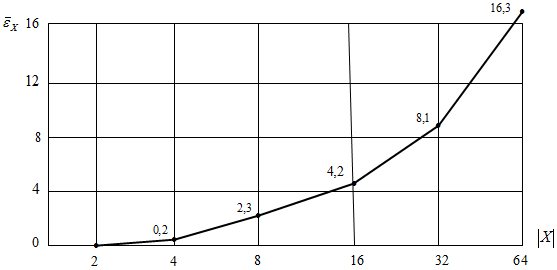
**Результаты вычислительных экспериментов и их обсуждение.** На Рисунке 1 представлены некоторые результаты исследования эффективности для функции Растригина.



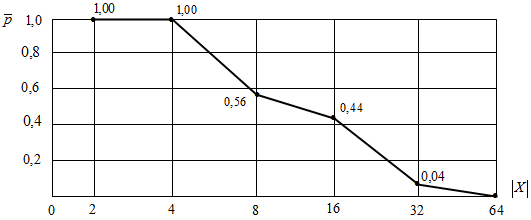
а) Число итераций  в функции размерности пространства поиска 



б) Средняя достигнутая точность по значениям целевой функции в зависимости от 



в) Средняя достигнутая точность положения результирующей точки в пространстве поиска в функции 



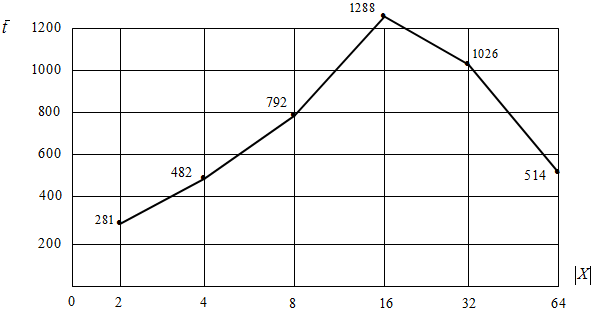
г) Оценка вероятности локализации глобального минимума в зависимости от 

Рисунок 1 ‑ Результаты вычислительных экспериментов: функция Растригина

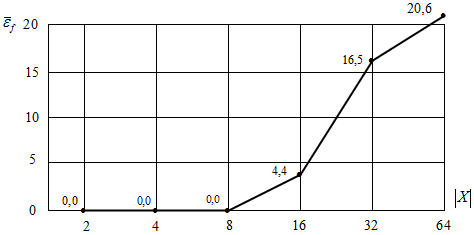
Прежде, чем анализировать представленные результаты, отметим, что из-за большого размера начального куба  (длина ребра =200), число локальных минимумов функции Растригина (равно, как и функции Экли) в этой области чрезвычайно велико и быстро растет с ростом размерности  пространства поиска [1]. Потому отыскание глобального минимума этих функций в указанной области представляет собой экстремально сложную задачу для любого из популяционных алгоритмов, особенно, при .

Результаты, представленные на Рисунках 1а ‑ 1в, ожидаемо показывают, что с ростом размерности задачи  быстро растет число итераций  до стагнации вычислительного процесса, а также погрешности , . Интегрально, эффективность алгоритма и реализующей его программы характеризует Рисунок 1г, из которого следует, что до  обеспечивается приемлемая вероятность локализации глобального минимума функции Растригина. Напомним, что нами использовано число особей популяции, равное , и метод мультистарта с числом стартов . Повысить вероятность локализации глобального минимума, естественно, за счет увеличения вычислительных затрат на поиск, можно путем увеличения числа особей и/или числа стартов.

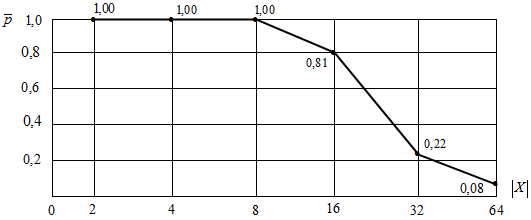
На Рисунке 2, аналогичным Рисунку 1, представлены результаты исследования эффективности для функции Экли.



а) 



б) 



в) 

Рисунок 2 ‑ Результаты вычислительных экспериментов: функция Экли

В силу близости ландшафтов функций Растригина и Экли, качественно, представленные результаты для функции Экли близки результатам исследования для функции Растригина. Отметим неожиданное уменьшение числа итераций  при размерности вектора варьируемых параметров , превышающем 16. Причина этого эффекта не понятна и требует самостоятельного исследования. Оценка вероятности  локализации глобального минимума в данном случае оказывается более высокой, чем в случае функции Растригина: сохраняется равной единице вплоть до , уменьшаясь до приемлемой величина при . Как и в предыдущем случае, повысить эффективность поиска можно путем увеличения числа особей и/или число стартов алгоритма.

**Заключение.** Легко видеть, что формулы (4), 5) близки по структуре к формулам, которые определяют эволюцию особей в алгоритме роя части PSO [1]. Основное отличие заключается в том, что при вычислении вектора скорости особи (см. (5)) используется не лучший результат всей популяции, как в алгоритме PSO, а лучший результат группы этой особи. Такое решение имеет свои позитивные и негативные стороны. Тот факт, что в алгоритме RIO не приходится отыскивать глобально лучшее решение, может позволить, например. эффективно распараллелить этот алгоритм для вычислительных систем с распределенной памятью. С другой стороны, отказ в алгоритме RIO от использования метрики близости в топологическом пространстве в пользу метрики пространства поиска  снижает диверсификационные свойства алгоритма. Можно предположить, что именно поэтому авторам алгоритма RIO пришлось ввести правило поведения *Find Food*, призванное предотвратить преждевременную стагнацию вычислительного процесса. Кроме того, использование метрики пространства  вынуждает на каждой итерации алгоритма RIO вычислять евклидовы расстояния между всеми парами особей популяции, что требует значительных вычислительных затрат и глобальной информации о всей популяции.

Для диверсификации поиска алгоритм RIO использует в механизме образования групп особей  следующий прием. В качестве точки  выбирается не обязательно самая лучшая точка среди следов всех соседей  особи , но одна из лучших таких точек. Выбор происходит с вероятностью , зависящей от размера группы (см. (3)). В основе правила определения вероятности  лежат биологические предпосылки [4].

Поясним, наконец, оригинальный механизм диверсификации поиска, реализующий правило поведения особей *Find Food.* С каждым тараканом  алгоритм связывает его целочисленный уровень «голода» , который на этапе инициализации полагаем распределенным равномерно случайно в заданном интервале. Значение величины на каждом шаге эволюции популяции увеличиваем на фиксированную целочисленную величину до выполнения условия , где  ‑ заданный уровень. При этом по правилам инициализации начальной популяции перемещаем таракана в случайную точку областипоиска и обнуляем его уровень голода*.* Такой механизм диверсификации поиска можно интерпретировать, как частичную «встряску» популяции [1].



Результаты представленного исследования позволяют утверждать, что оптимизационный алгоритм нашествия тараканов RIO перспективен для поиска экстремума сложных многоэкстремальных функций. Как почти всякий популяционный алгоритм, алгоритм RIO имеет значительное число свободных параметров, так что эффективность этого алгоритма применительно к той или иной задаче глобальной оптимизации может быть значительно повышена за счет его параметрической настройки [1]. В развитии работы авторы предполагают выполнить такую настройку алгоритма RIO, а также реализовать широкое сравнительное исследование этого алгоритма и алгоритма роя частиц PSO.

**Литература**

1. Карпенко А.П. Современные алгоритмы поисковой оптимизации. Алгоритмы вдохновленные природой. Москва: Издательство МГТУ им. Н.Э. Баумана; 2014.
2. Havens T.C. et al. Roach infestation optimization. *In: Proceedings of the 2008 IEEE Swarm Intelligence Symposium,* St. Louis, MO, USA*.* 2008: 21–23.
3. Bo Xing, Wen-Jing Gao. Innovative Computational Intelligence: A Rough Guide to 134 Clever Algorithms. Springer International Publishing Switzerland; 2014.
4. Jeanson R. et al. Selforganized aggregation in cockroaches. *Animal Behaviour*. 2005; 69: 169–180.
5. Halloy J. et al. Social integration of robots into groups of cockroaches to control self-organizined choices. *Science*. 2007; 318(5853): 1155-1158.
6. Ame J. at al. Collegial decision making based on social amplification leads to optimal group formation. *Proc. Natl. Acad. Sci*. 2006; 103(15): 5835–5840.
7. Garnier S. et al. Collective decision-making by a group of cockroach-like robots. Proc. *2005 IEEE Swarm Intelligence Symposium (SIS 2005)*. ‑ Pasadena, CA, USA. 2005; 233-240.
8. Watanabe H., Mizunami M. Pavolv’s cockroach: Classical conditioning of salivation in an insect. *PLoS ONE*. 2007; 2(6): 529.
9. Kennedy J., Eberhardt R. Particle swarm optimization. *Proceedings of the IEEE* *Int. Conf. on Neural Networks,* Piscataway, NJ. 1995; 1942–1948.
10. Clerc M. Particle Swarm Optimization. Newport Beach, CA: ISTE USA, 2006.

**A.P. Karpenko**

Bauman Moscow State Technical University (National research university of technology), Moscow, Russian Federation

**EFFECTIVENESS RESEARCH OF A GLOBAL OPTIMIZATION ALGORITHM INSPIRED BY SOME ASPECTS OF COCKROACH BEHAVIOUR**

Keywords: *global unconstrained optimization, population based algorithm, particle swarm optimization algorithm, Rastrigin function, Ackley function.*

**Abstract.** *Consider the Roach Infestation Optimization (RIO) algorithm, which belongs to the class of population-based algorithms inspired by wildlife. The RIO algorithm was proposed in 2008 and can be considered a deep modification of the well-known and one of the most effective particle swarm optimization (PSO) algorithms. The interest in the RIO algorithm is due to the fact that, due to the high efficiency of the PSO algorithm for a wide range of global optimization problems, the study of the modification of this algorithm, which is represented by the RIO algorithm, is of particular interest. The purpose of the work is to implement software and study the efficiency of the RIO algorithm for the well-known complex multimodal test functions of Ratrigin and Ackley. A feature of the study is the search for a global extremum (minimum) of these functions in a wide region of the search space, in which the number of local minima of these functions is extremely large. We present the formulation of the considered global optimization problem, as well as a description of the RIO algorithm, a distinctive feature of which is the use not of the original designations of the authors of this algorithm, but of the unified designations that we use when considering other population algorithms. We describe the software that implements the algorithm and the organization of computational experiments to study its effectiveness. Finally, the article presents the research results showing the high prospects of the RIO algorithm for solving global optimization problems.*

Сведения об авторах:

**Карпенко Анатолий Павлович** – доктор физико-математических наук, профессор, заведующий кафедрой САПР МГТУ им. Н.Э. Бауман, [apkarpenko@mail.ru](mailto:apkarpenko@mail.ru)