

УДК 004.02, 004.94, 519.6

**САМОАДАПТИВНЫЙ АЛГОРИТМ БАКТЕРИАЛЬНОГО ПОИСКА****Ершов Николай Михайлович<sup>1</sup>, Полуян Сергей Владимирович<sup>2</sup>**

<sup>1</sup>Кандидат физико-математических наук, доцент;  
ГБОУ ВО МО «Университет «Дубна»,  
Институт системного анализа и управления;  
141980, Московская область, г. Дубна, ул. Университетская, 19;  
e-mail: ershovnm@gmail.com.

<sup>2</sup>Аспирант;  
ГБОУ ВО МО «Университет «Дубна»,  
Институт системного анализа и управления;  
141980, Московская область, г. Дубна, ул. Университетская, 19;  
e-mail: svpoluyan@gmail.com.

*Алгоритмы роевой оптимизации, рассматриваемые в данной работе, основаны на моделировании коллективного поведения в колониях живых организмов – муравьев, бактерий, пчел и т.д. Настоящая работа посвящена описанию нового подхода к построению самоадаптивных алгоритмов роевой оптимизации, в которых происходит автоматическая настройка части параметров алгоритма в процессе его выполнения. Идея построения самоадаптивного эволюционного алгоритма заключается в том, что на фоне основного алгоритма оптимизации запускается вспомогательный генетический алгоритм, целью работы которого является настройка параметров базового алгоритма, обеспечивающая максимально возможную скорость его сходимости. Приводятся результаты численного исследования самоадаптивного алгоритма бактериального поиска на примере решения стандартных тестовых задач непрерывной оптимизации.*

**Ключевые слова:** методы роевого интеллекта, оптимизация, генетические алгоритмы.

**SELF-ADAPTIVE BACTERIAL FORAGING ALGORITHM****Ershov Nikolay<sup>1</sup>, Poluyan Sergey<sup>1</sup>**

<sup>1</sup>Candidate of Science in Physics and Mathematics, assistant professor;  
Dubna State University,  
Institute of the system analysis and management;  
141980, Dubna, Moscow reg., Universitetskaya str., 19;  
e-mail: ershovnm@gmail.com.

<sup>2</sup>PhD student;  
Dubna State University,  
Institute of the system analysis and management;  
141980, Dubna, Moscow reg., Universitetskaya str., 19;  
e-mail: ershovnm@gmail.com.

*Swarm optimization algorithms, considered in this paper, are based on the modeling of collective behavior in the colonies of living organisms - the ants, bacteria, bees, etc. The present work is devoted to the description of a new approach to building self-adaptive swarm optimization algorithms, with automatic adjustment of the algorithm parameters during its execution. The idea of constructing self-adaptive evolutionary algorithm is following – in the background of the main optimization algorithm we run auxiliary genetic algorithm, which purpose is the adjustment of the parameters of the basic algorithm, providing the maximum possible speed of convergence. The results of the numerical analysis of the self-adaptive version of bacterial foraging algorithm for standard test problems of continuous optimization are described.*

**Keywords:** swarm intelligence, optimization, genetic algorithms.

## Введение

В основе работы практически всех алгоритмов роевой оптимизации [1] лежит, как правило, достаточно простая идея. Например, в алгоритме бактериального поиска [2], каждое решение ведет себя подобно бактерии, движущейся под действием тех или иных химических соединений (хемотаксис) – до тех пор, пока концентрация полезных веществ возрастает, бактерия движется прямолинейно с постоянной скоростью, как только концентрация начинает падать, бактерия меняет случайным образом направление своего движения.

Однако эта простая идея при реализации обрастает разнообразными вспомогательными правилами (эвристиками), каждое из которых управляется одним или несколькими параметрами. Это приводит к тому, что конечный алгоритм оказывается зависимым от большого числа параметров, которые влияют на эффективность этого алгоритма. К сожалению, настроить такие параметры универсальным образом чаще всего не удается.

Это значит, что для каждого отдельного набора входных данных оказывается возможным подобрать такую комбинацию параметров, которая обеспечит лучшую сходимость алгоритма. Настоящая работа посвящена описанию подхода к построению самоадаптивных алгоритмов роевой оптимизации, в которых происходит автоматическая настройка части параметров алгоритма в процессе его эволюции.

### 1. Простой алгоритм бактериального поиска

В качестве тестового алгоритма была выбрана упрощенная версия алгоритма бактериального поиска для задачи непрерывной оптимизации, в котором по сравнению с классическим вариантом отсутствуют шаги саморепликации и рассеивания. На каждой итерации мы пытаемся изменить каждое решение в заданном направлении, если этот шаг приводит к улучшению решения, то мы его принимаем, если нет, то выбирается новое случайное направление. Таким образом, вектор направлений оказывается связанным с каждым отдельным решением. Длина  $\delta$  этого вектора, однако, является глобальным параметром, т.е. общим для всей популяции. Известно, что этот параметр очень сильно влияет на сходимость алгоритма – даже при решении одной конкретной задачи на разных этапах желательно использовать разные его значения.

С этой целью применяются такие известные методики, как адаптивное изменение шага (аналогично методу *R-propagation* [3] обучения нейронных сетей) или имитация отжига [4], когда на начальном этапе длина  $\delta$  устанавливается большой, а затем постепенно уменьшается практически (в пределе) до нуля.

### 2. Схема самоадаптации

Основная идея построения самоадаптивного эволюционного заключается в том, что на фоне основного алгоритма (в нашем случае это алгоритм бактериального поиска) запускается вспомогательный генетический алгоритм [5]. Целью работы генетического алгоритма является настройка параметров базового алгоритма, обеспечивающая максимально возможную скорость его сходимости. Заметим, что такой алгоритм не является гибридным, т.к. только базовая его часть отвечает собственно за оптимизацию заданной функции.

Для случая простого бактериального алгоритма и параметра  $\delta$  схема самоадаптации выглядит следующим образом. Во-первых, параметр  $\delta$  становится локальным, т.е. привязанным к каждому отдельному решению (особи популяции). В начальный момент времени значения этого параметра устанавливаются случайным образом. Затем, каждые  $\tau$  итераций основного алгоритма производится одна итерация генетического алгоритма, включающая в себя три шага – отбор, скрещивание и мутацию.

Для выполнения отбора используется турнирная схема, две случайно выбранные особи сравниваются по параметру  $\Delta F$ , который представляет собой изменение целевой функции  $F$  данной особи за промежуток времени  $\tau$ . С вероятностью  $p_m$  выбирается лучшее значение в паре, в противном случае

выбирается худшее значение. Старые значения параметра  $\delta$  обеих особей заменяются выбранным значением-победителем.

Для рекомбинации параметров применяется вариант арифметического скрещивания. Для этого выбирается пара особей с номерами  $i$  и  $j$ . Новые значения параметров  $\delta_i$  и  $\delta_j$  определяются усреднением старой пары значений:

$$\delta_i = p_c \delta_i + (1 - p_c) \delta_j \text{ и } \delta_j = p_c \delta_j + (1 - p_c) \delta_i.$$

Мутация заключается в умножении текущего параметра  $\delta_i$  на случайный множитель из диапазона  $[0.5, 2]$ . Мутация применяется к каждому решению, т.к. нас интересует высокая степень адаптивности алгоритма.

### 3. Численное исследование

Описанная выше схема была реализована программно, при этом были получены следующие результаты. Исследовалась задача минимизации сферической двумерной функции:

$$F(x, y) = (x - x_0)^2 + (y - y_0)^2 + b.$$

Эта функция имеет единственный минимум в точке  $(x_0, y_0)$  равный  $b$ .

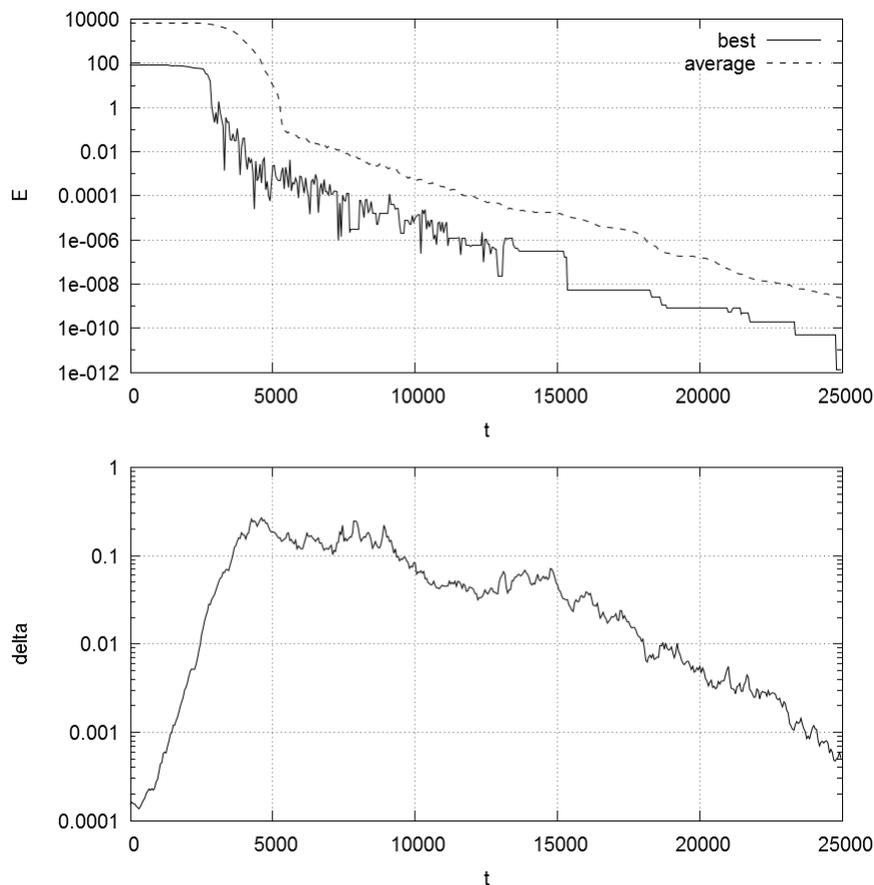


Рис. 1. Сходимость самоадаптивного бактериального алгоритма для начальной области поиска  $(x, y) \in [-100, 100]$

На рисунках 1 и 2 представлены графики сходимости (показаны среднее и лучшее значения целевой функции по популяции) и графики, показывающие среднее значение параметра  $\delta$  по популяции. Рисунок 1 соответствует начальной области поиска  $(x, y) \in [-100, 100]$ , рисунок 2 – области  $(x, y) \in [-10, 10]$ . Видно, что в обоих случаях идет процесс настройки параметра  $\delta$ , причем этот процесс происходит по-разному, в зависимости от того, насколько далеко популяция находится от глобально-

го минимума целевой функции. Для сравнения, на рисунке 3 показан график сходимости неадаптивного варианта алгоритма (с несколькими фиксированными значениями параметра  $\delta$ ) и предложенного адаптивного варианта.

Кроме того, было рассмотрено применение предложенного подхода к настройке сразу двух параметров. Для этого во втором варианте базового (бактериального) алгоритма каждой особи был приписан еще один параметр  $\theta$  (играющий роль температуры). Этот параметр используется следующим образом: в том случае, когда новое решение оказывалось хуже старого, переход к нему делается с вероятностью  $p = \exp(-\Delta F/\theta)$ . Введение такого параметра позволяет популяции не «застревать» в локальных экстремумах целевой функции – на начальном этапе температуру следует делать высокой, а затем постепенно снижать ее до нуля (классический вариант имитации отжига).

На рисунке 4 показаны сходимость алгоритма и изменение средних значений его параметров от номера итерации. Видно, что схема самоадаптации работает и в случае двух параметров.

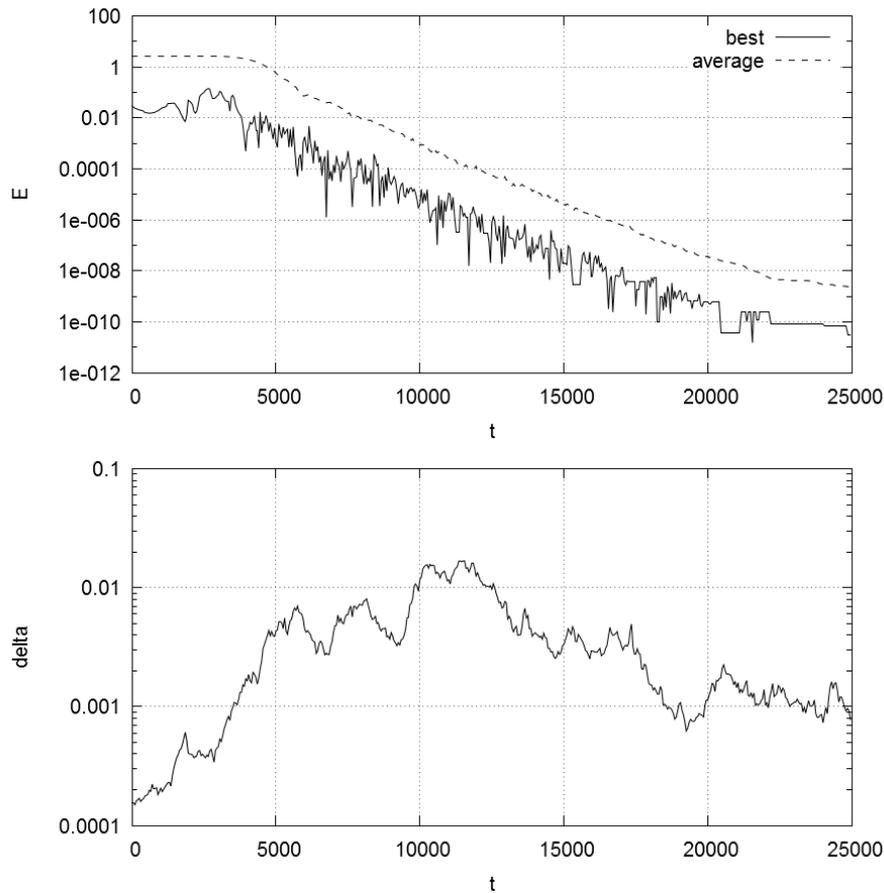


Рис. 2. Сходимость самоадаптивного бактериального алгоритма для начальной области поиска  $(x, y) \in [-10, 10]$

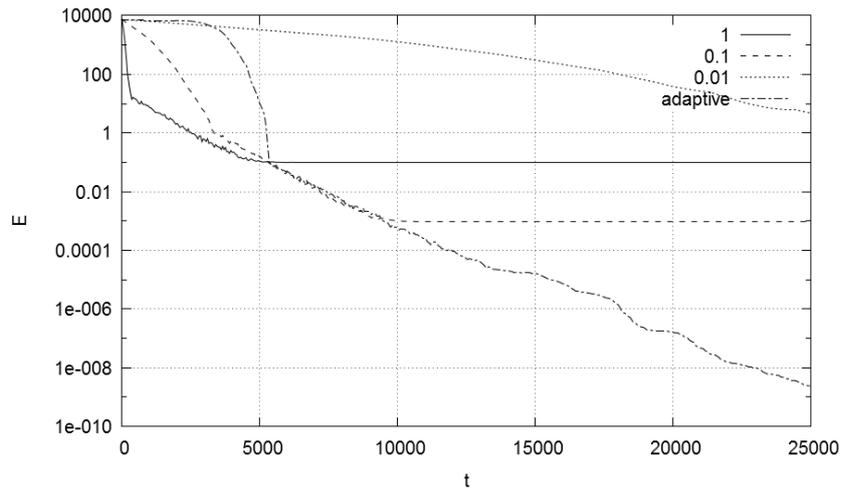


Рис. 3. Сравнение алгоритмов бактериального поиска с адаптацией и без адаптации

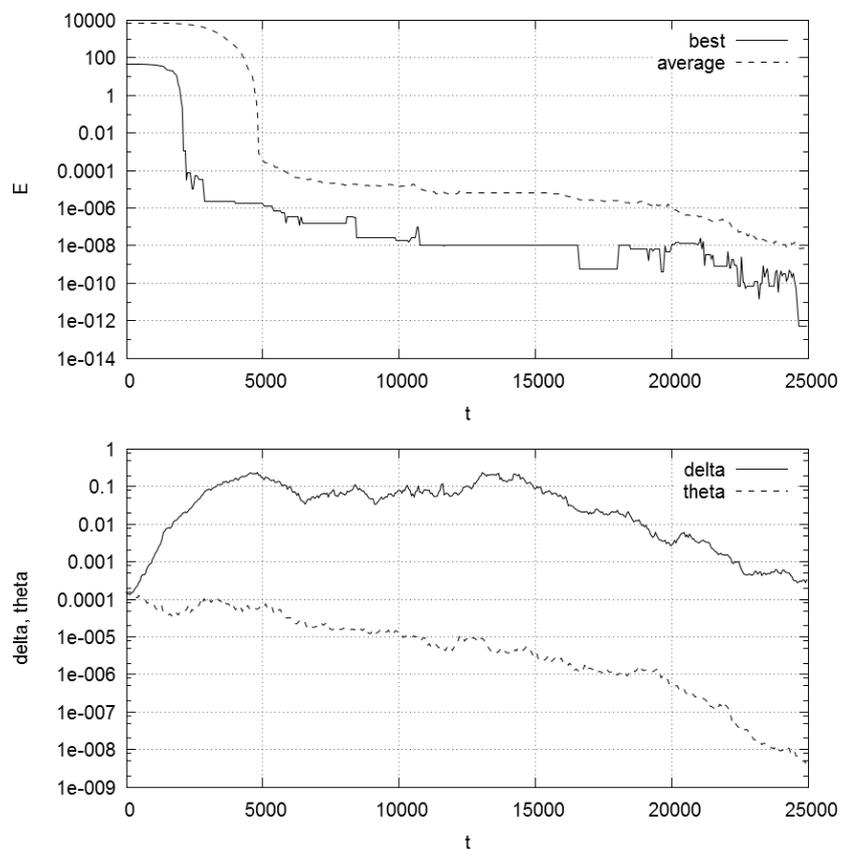


Рис. 4. Сходимость самоадаптивного бактериального алгоритма с адаптацией двух параметров

## Заключение

В результате выполненной работы получены следующие результаты:

- предложена схема автоматической настройки параметров эволюционных алгоритмов, основанная на использовании генетического алгоритма;
- рассмотрено применение этой схемы к алгоритму бактериального поиска;
- проведено численное исследование работы самоадаптивных версий рассмотренного алгоритма бактериального поиска, показавшее работоспособность предложенной схемы самоадаптации.

### *Список литературы*

1. Bonabeau E., Dorigo M., Theraulaz G. Swarm Intelligence: from Natural to Artificial Systems. — New York: Oxford University Press, Inc., 1999.
2. Passino K. M. Biomimicry of Bacterial Foraging for Distributed Optimization and Control // IEEE Control Systems Magazine. — 2002. — Vol. 22. — № 3. — Pp. 52–67.
3. Martin Riedmiller and Heinrich Braun. A direct adaptive method for faster backpropagation learning: The rprop algorithm. In IEEE INTERNATIONAL CONFERENCE ON NEURAL NETWORKS, pages 586–591, 1993.
4. S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. Science, Number 4598, 13 May 1983, 220, 4598:671–680, 1983.
5. Whitley D. A Genetic Algorithm Tutorial // Statistics and Computing. — 1994. — Vol. 4. — № 2. — Pp. 65–85.