

# Структурно-параметрическая адаптация генетического алгоритма

Ю. В. Минаева

В статье рассматривается методика комплексной адаптации генетического алгоритма, позволяющая автоматизировать выбор варианта генетического оператора из заданного множества и произвести настройку его фактических параметров. Подобный подход позволит увеличить универсальность вычислительного алгоритма и уменьшить время эволюции за счет использования тех методов, которые уже показали свою эффективность при решении данной задачи. Для реализации процедуры параметрической настройки операторов предлагается использовать динамически формируемый набор правил, связывающих значения параметров операторов со статистическими характеристиками их операндов.

*Ключевые слова:* оптимизация, генетический алгоритм, генетические операторы, динамическая адаптация.

## 1. Введение

При проектировании сложных технических и экономических систем часто возникает задача оптимального выбора внутренних характеристик системы, описывающих ее структуру или поведение. Для увеличения эффективности и скорости реализации процедур оптимального поиска разработано множество методов, однако практически все они имеют ограничения, связанные с характером математической модели исследуемой системы. Наиболее универсальными с этой точки зрения являются алгоритмы управляемого перебора, основанные на процессах эволюционного развития биологических популяций. Одним из таких методов является генетический алгоритм (ГА).

Потенциальные решения в ГА представляются в виде популяции хромосом, каждая из которых имеет в своем составе набор генов. Смыслом существования всех хромосом в популяции является повышение степени индивидуальной полезности за счет нахождения более оптимальных значений целевой функции. Для этого хромосомы постоянно обновляют свой генотип путем многократного применения операторов селекции, скрещивания и мутации [1].

Для каждого генетического оператора (ГО) к настоящему времени разработано множество вариантов реализации. Эффективность применения конкретного варианта сильно зависит от класса задачи оптимизации, этапа ее решения, а также от выбора его параметров. С учетом широкого спектра решаемых с помощью ГА задач практически невозможно произвести отбор варианта оператора и его параметрическую адаптацию априорно.

Для решения проблемы большой зависимости эффективности работы алгоритма от его параметров используются следующие подходы [2, 3]:

- многократное выполнение прогона алгоритма с различными значениями параметров;
- динамическая адаптация значений параметров в процессе оптимизации;
- использование для подбора управляемых параметров «мета-алгоритма», в качестве которого также может выступать ГА.

Поскольку выполнение ГО требует больших затрат времени и ресурсов, то применение первого и третьего подходов, использующих несколько экземпляров метода, накладывает ограничения на вычислительные мощности систем и не подходит для решения задач, связанных с оперативной обработкой данных. Второй подход является наиболее перспективным,

но в этом случае необходима разработка алгоритма, учитывающего влияние всех или некоторых параметров на работу метода в целом.

## 2. Методы адаптации вычислительного алгоритма

В [4–6] выделяют следующие способы реализации динамического управления параметрами вычислительного алгоритма:

- детерминированный – параметры алгоритма изменяются в соответствии с заданным временным графиком без анализа текущего результата;
- эксплицитный – к алгоритму дополнительно добавляется корректирующий блок, работающий по принципу обратной связи и изменяющий параметры алгоритма в зависимости от результатов, полученных на текущей итерации;
- имплицитный – управляемые параметры кодируются, добавляются в качестве новых генов в хромосомы и подвергаются в процессе решения скрещиванию и мутации.

Наиболее перспективными являются эксплицитная и имплицитная формы адаптации, однако первая требует разработки соответствующих методов динамической коррекции, а реализация второй влечет за собой увеличение вычислительной сложности, во-первых, из-за большей длины хромосомы, и во-вторых, из-за необходимости выполнения различных генетических операторов над частями хромосомы с данными и с параметрами алгоритма.

Большинство исследовательских работ в области эксплицитной адаптации посвящено проблеме выбора параметров ГО. Чаще всего предлагается подвергать динамической коррекции вероятности мутации и кроссовера [7–10], размер выборки для турнирной селекции [4, 11–13] и размер популяции [14–18].

Однако с учетом увеличения разнообразия и сложности оптимизационных задач, а также большого количества уже предложенных вариантов ГО задача априорного выбора одного оператора оказывается очень трудоемкой и требует от специалиста, во-первых, достаточного объема накопленных знаний об особенностях решаемых задач и ГО, а во-вторых, проведения большого количества экспериментов для обоснования выбора.

Различные подходы к автоматизации динамического выбора одного варианта из заданного множества операторов рассматриваются в работах [19, 20]. В качестве критериев выбора авторы предлагают использовать вероятности вызова операторов, пересчитываемые после каждого шага в зависимости от успешности использования выбранного варианта ГО.

Для дальнейшего повышения адаптационных свойств ГА имеет смысл проводить не только автоматизированный выбор оператора (т.е. структурную адаптацию алгоритма), но и иметь возможность настройки фактических параметров ГО. Такая комплексная адаптация алгоритма может являться существенным фактором повышения гибкости и универсальности поисковых процедур.

Для проверки эффективности комплексной адаптации ГА проведем ее экспериментальное исследование. В качестве способа реализации параметрической адаптации будем использовать зависимость между значениями параметров операторов и такими статистическими характеристиками их операндов, как среднее квадратическое отклонение, расстояние Хэмминга, частотное распределение и др., предложенную в [21].

### 3. Методика комплексной адаптации ГА

Предлагается следующая методика комплексной структурно-параметрической адаптации ГА.

Пусть  $T = \{\tau_1, \dots, \tau_{n_T}\}$  – допустимые типы операторов ГА, где  $n_T$  – число типов. Каждый тип операторов имеет несколько вариантов реализации, т.е.  $\tau_i = \{t_{i1}, \dots, t_{in_{\tau_i}}\}$ , где  $n_{\tau_i}$  – количество вариантов оператора типа  $\tau_i$ ,  $i = 1, \dots, n_T$ . В общем случае каждый вариант исполнения  $t_{ij}$  характеризуется кортежем  $\langle P_{ij}, D_{ij}, R_{ij} \rangle$ , где  $P_{ij}$  – вероятность вызова  $j$ -й альтернативы  $i$ -го оператора,  $j = 1, \dots, n_{\tau_i}$ ,  $D_{ij}$  – список формальных параметров оператора,  $R_{ij} = (C_{ij}, V_{ij})$  – совокупность правил, ставящих в соответствие множеству статистических характеристик операндов  $C_{ij}$  множество наборов фактических параметров оператора  $V_{ij}$ .

Алгоритм комплексной адаптации ГА включает следующие шаги:

1. Перед началом работы алгоритма вероятностям вызова вариантов операторов присваиваются одинаковые значения, а множество правил обнуляется:

$$P_{ij}^0 = \frac{1}{n_{\tau_i}}, \quad R_{ij} = \emptyset, \quad \forall i = 1, \dots, n_T, \quad \forall j = 1, \dots, n_{\tau_i},$$

Устанавливается номер итерации  $k = 1$ .

2. Выбор варианта ГО  $t \in \tau_i$  осуществляется по методу рулетки на основании значений  $P_{ij}^k$ .

3. Определение множества характеристик операндов  $\tilde{c}$  оператора  $t$  в зависимости от числа используемых операндов  $p$ :

$$\tilde{c} = \begin{cases} d, & \text{если } p \leq 2, \\ \sigma, & \text{если } p > 2, \end{cases}$$

где  $d$  – расстояние Хэмминга между хромосомами-операндами,  $\sigma$  – среднеквадратическое отклонение для набора операндов.

4. В списке правил  $R_{ij}$  производится поиск такой пары  $(c_m, v_m)$ , для которой  $c_m \in [\tilde{c} - \delta_c, \tilde{c} + \delta_c]$ ,  $\forall m = 1, \dots, n_{R_{ij}}$ ,  $n_{R_{ij}}$  – длина списка,  $\delta_c$  – допустимое отклонение для  $\tilde{c}$ . Если такой элемент найден, то фактические параметры  $\tilde{v}$  варианта оператора  $t$  генерируются по нормальному закону распределения со средним значением  $v_m$  и дисперсией  $\delta_v$  и сама пара  $(c_m, v_m)$  удаляется, иначе набор параметров генерируется случайным образом по равномерному закону распределения:

$$\tilde{v} = \begin{cases} \text{norm}(v_m, \delta_v), & \text{если } c_m \in R_{ij}, \\ \text{random}(v_{\min}, v_{\max}), & \text{если } c_m \notin R_{ij}, \end{cases}$$

где  $v_{\min}$  и  $v_{\max}$  – минимальное и максимальное допустимые значения для параметров оператора соответственно.

5. Реализация оператора  $t$  с набором параметров  $\tilde{v}$ .

6. Эффективность оператора  $t$  определяется по формуле:

$$E_t = \frac{\bar{F}(X_{\text{out}})}{\bar{F}(X_{\text{in}})},$$

где  $X_{\text{in}} = \{x_{\text{in}_1}, \dots, x_{\text{in}_{n_{X_{\text{in}}}}}\}$  – входное множество хромосом емкостью  $n_{X_{\text{in}}}$ ,

$X_{\text{out}} = \{x_{\text{out}_1}, \dots, x_{\text{out}_{n_{X_{\text{out}}}}}\}$  – выходное множество емкостью  $n_{X_{\text{out}}}$ ,  $\bar{F}(X_{\text{in}})$  и  $\bar{F}(X_{\text{out}})$  –

средняя полезность входного и выходного множества соответственно, определяются следующим образом:

$$\bar{F}(X_{in}) = \sum_{i=1}^{n_{X_{in}}} f(x_{in_i}) / n_{X_{in}}, \quad \bar{F}(X_{out}) = \sum_{i=1}^{n_{X_{out}}} f(x_{out_i}) / n_{X_{out}}.$$

В том случае, если вариант  $t$  показал себя эффективным, происходит добавление нового правила, связывающего характеристики ГО с его фактическими параметрами:

$$R_{ij} = \begin{cases} R_{ij} \cup (\tilde{c}, \tilde{v}), & \text{если } E_t > 1, \\ R_{ij}, & \text{если } E_t \leq 1. \end{cases}$$

7. В зависимости от эффективности применения варианта оператора динамически изменяется вероятность его вызова  $P_{ij}$ :

$$P_{ij}^k = \begin{cases} P_{ij}^{k-1} + \delta P, & \text{если } E_{ij} > 1, \\ P_{ij}^{k-1} - \delta P, & \text{если } E_{ij} \leq 1, \end{cases}$$

где  $E_{ij}$  – эффективность  $j$ -й альтернативы  $i$ -го оператора,  $\delta P$  – поправка к значению вероятности,  $k$  – номер итерации.

Для корректной работы алгоритма для вероятностей вызова вариантов операторов установлены граничные значения  $P_{\max}$  и  $P_{\min}$ :

$$P_{ij}^k = \begin{cases} P_{\min}, & \text{если } P_{ij}^k < P_{\min}, \\ P_{ij}^k, & \text{если } P_{\min} \leq P_{ij}^k \leq P_{\max}, \\ P_{\max}, & \text{если } P_{ij}^k > P_{\max}. \end{cases}$$

Для сохранения постоянной суммы фактическая поправка к вероятности вызова равномерно распределяется между остальными вариантами  $i$ -го оператора с обратным знаком:

$$P_{im}^k = P_{im}^{k-1} - \frac{P_{ij}^k - P_{ij}^{k-1}}{n_{\tau_i} - 1}, \quad \forall m = 1, \dots, n_{\tau_i}, \quad m \neq j.$$

8. Если выполнены все операторы, то останов работы алгоритма, иначе присвоить  $k = k + 1$  и перейти к шагу 2.

Рассмотрим работу алгоритма при  $n_{\tau} = 3$  для адаптации операторов селекции, скрещивания и мутации. Множества исходных данных для алгоритма имеют вид:

$$T = \{\text{селекция, кроссовер, мутация}\},$$

$$\tau_1 = \{\text{метод рулетки, ранговая, турнирная}\},$$

$$\tau_2 = \{\text{одноточечный, двухточечный, равномерный}\},$$

$$\tau_3 = \{\text{инвертирование, перестановка}\}.$$

$$D_{11} = D_{12} = \emptyset, \quad D_{13} = \{\text{объем выборки}\},$$

$$D_{21} = D_{22} = \{\text{позиции точек деления}\}, \quad D_{23} = \{\text{маска кроссовера}\},$$

$$D_{31} = D_{32} = D_{33} = \{\text{номера битов}\}.$$

При расчете эффективности входным множеством для оператора селекции будет само множество  $X$ , для скрещивания – множество, состоящее из отобранных родительских хромосом, для мутации – хромосома, которая подвергнется мутации. Выходным множеством для селекции будет множество родительских хромосом, для кроссовера – полученные в результате скрещивания потомки, для мутации – сама хромосома.

Для оценки эффективности предлагаемого алгоритма сравним его работу с каноническим генетическим алгоритмом (КГА) и алгоритмом автоматического выбора операторов (АВОГА), предложенным в [21]. Используемые тестовые функции представлены в табл. 1.

Таблица 1. Набор тестовых функций для проверки алгоритма

Название	Формула	Интервал
Сферическая функция	$F(x) = \sum_{i=1}^n x_i^2$	$x \in (-5, 5)$
Функция Растригина	$F(x) = 10n + \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i))$	$x \in (-5, 5)$
Функция Розенброка	$F(x) = \sum_{i=1}^{n-1} \left( 100(x_{i+1} - x_i^2)^2 + (x_i - 1) \right)$	$x \in (-2.5, 2.5)$
Функция Швевеля	$F(x) = 418,9829n + \sum_{i=1}^n \left( -x_i \cdot \sin(\sqrt{ x_i }) \right)$	$x \in (-525, 525)$

Для всех алгоритмов популяция состояла из 50 хромосом и имела максимальное время жизни в 500 поколений. Для предлагаемого алгоритма использовались следующие дополнительные параметры:  $P_{\max} = 0.9$ ,  $P_{\min} = 0.1$ ,  $\delta P = 0.1$ ,  $\delta_v = 1$  для номеров битов и  $\delta_v = 0.1$  для вероятностей выбора значения бита в маске кроссовера,  $\delta_c = 1$  для расстояния Хэмминга и  $\delta_c = 0.1\sigma$  для среднеквадратического отклонения  $\sigma$ . Средние значения эффективности и времени решения вначале находились для каждой функции за 100 прогонов, а затем усреднялись для всех тестовых функций. Результаты экспериментов приведены в табл. 2.

Таблица 2. Эффективность различных вариантов ГА

Название алгоритма	Эффективность	Время эволюции (количество поколений)
Канонический ГА	0.76	420
ГА с автоматическим выбором оператора	0.87	374
ГА с комплексной адаптацией	0.89	328

По результатам эксперимента можно сделать вывод, что эффективность предлагаемого алгоритма превосходит КГА и не уступает АВОГА, а время решения меньше, чем у остальных алгоритмов. Основным преимуществом алгоритма является отсутствие необходимости в априорной настройке параметров, что делает полезным его применение в тех случаях, когда такая настройка затруднена или малоэффективна, а также когда пользователь не является экспертом в данной области.

#### 4. Заключение

В данной статье была предложена методика комплексной структурно-параметрической адаптации генетического алгоритма, основанная на выборе наиболее эффективных вариантов генетических операторов и применении процедуры их параметрической настройки. В

основе такого подхода лежит зависимость между значениями фактических параметров операторов и статистическими характеристиками операндов. Подобная схема решения позволяет использовать каждый вариант оператора в таких состояниях вычислительного процесса, при которых его эффективность будет наиболее высока. Рассмотренная методика была апробирована на нескольких стандартных тестовых функциях. По результатам тестирования можно сделать вывод, что комплексная адаптация позволяет повысить надежность процесса поиска глобального оптимума и уменьшить время эволюции по сравнению с канонической версией алгоритма и методом автоматического выбора генетических операторов без их параметрической адаптации.

## Литература

1. Емельянов В. В., Курейчик В. В., Курейчик В. М. Теория и практика эволюционного моделирования. М.: Физматлит. 2003. 432 с.
2. Nunez-Letamendia L. Fitting the control parameters of a genetic algorithm: An application to technical trading systems design // *European Journal of Operational Research*. 2007. Vol. 179, № 3. P. 847–868.
3. Harik G. R., Lobo F. G. A parameter-less genetic algorithm // *Proceedings of the Genetic and Evolutionary Computation Conference*. 1999. P. 258–265.
4. Eiben A. E., Schut M. C., De Wilde A. R. Boosting genetic algorithms with self-adaptive selection // *Proceedings of the IEEE Congress on Evolutionary Computation*. 2006. P. 1584–1589.
5. Hinterding R., Michalewicz Z., Eiben A. E. Adaptation in Evolutionary Computation: a Survey // *Proceedings of the Fourth IEEE Conference on Evolutionary Computation*. 1997. P. 65–69.
6. Meyer-Nieberg S., Beyer H.-G. Self-adaptation in evolutionary algorithms // *Studies in Computational Intelligence*. 2007. Vol. 54. P. 47–75.
7. Laoufi A., Hadjeri S., Hazzab A. Adaptive probabilities of crossover and mutation in genetic algorithms for power economic dispatch // *International Journal of Applied Engineering Research*. 2006. Vol. 1, № 3. P. 393–408.
8. Lin W.-Y., Lee W.-Y., Hong T.-P. Adapting crossover and mutation rates in genetic algorithms // *Journal of Information Science and Engineering*. 2003. Vol. 19. P. 889–903.
9. Yang S. Adaptive crossover in genetic algorithms using statistics mechanism // *Proceedings of the 8th International conference on Artificial life*. 2002. P. 182–185.
10. Tabarzad M. A., Lucas C., Hamzeh A. Statistical genetic algorithm // *International Journal of Computer, Electrical, Automation, Control and Information Engineering*. 2008. Vol. 2, № 2. P. 483–487.
11. Whitley D. The GENITOR algorithm and selection pressure: Why rank-based allocation of reproductive trials is best // *Proceedings of the Third International Conference on Genetic Algorithms*. 1989. P. 116–121.
12. Goldberg D. E., Deb K. A comparative analysis of selection schemes used in genetic algorithms // *Foundations of Genetic Algorithms*. 1991. P. 69–93.
13. Back T., Fogel D. B., Michalewicz Z. *Evolutionary Computation1: Basic Algorithms and Operators*. IOP Publishing ltd. 2000. 339 p.
14. Цой Ю. Р., Спицын В. Г. Исследование генетического алгоритма с динамически изменяемым размером популяции // *Труды Международной научно-технической конференции «Интеллектуальные системы (IEEE AIS'05)»*. Научное издание. М.: Изд-во физико-математической литературы. 2005. С. 241–246.
15. Eiben A. E., Marchiori E., Valko V. A. Evolutionary algorithms with on-the-fly population size adjustment // *Parallel Problem Solving from Nature VIII*. 2004. Vol. 3242. P. 41–50.

16. Arabas J., Michalewicz Z., Mulawka J. GAVAPS — a genetic algorithm with varying population size // Proc. of the First IEEE International Conference on Evolutionary Computation. 1994. P. 73–78.
17. Back T., Eiben A. E., Vaart N. A. L. An empirical study on GAs without parameters // Parallel Problem Solving from Nature VI. 2000. Vol. 1917. P. 315–324.
18. Goldberg D. E., Deb K., Clark J. H. Genetic algorithms, noise, and the sizing of populations // Complex Systems. 1992. № 6. P. 333–362.
19. Семенкина М. Е. Самоадаптивные эволюционные алгоритмы проектирования информационных технологий интеллектуального анализа данных // Искусственный интеллект и принятие решений. 2013. № 1. С. 13–23.
20. Hilding F. G., Ward K. Automated operator selection on genetic algorithms // Proceedings of the 9th international conference on Knowledge-Based Intelligent Information and Engineering Systems. 2005. № 4. P. 903–909.
21. Shamsaei R., Hamzeh A., Rahmani A. Adaptive genetic algorithms based on learning classifier systems // 9'th Computer Society of Iran Computer Conference. 2004.

*Статья поступила в редакцию 29.03.2016;  
переработанный вариант — 06.04.2016*

**Минаева Юлия Васильевна**

старший преподаватель кафедры систем автоматизированного проектирования и информационных систем Воронежского государственного технического университета,  
тел. (473) 243-77-04

**Structural-parametric adaptation of genetic algorithm**

**Y. Minaeva**

The article considers the methodology of genetic algorithm complex adaptation which allows us to automate choice of genetic operator from given set and to adapt its actual parameters. Such approach allows us to increase computational algorithm universality and to decrease evolution time because of using the methods which have already shown their efficiency in similar task solution. Dynamic set of rules connecting operator parameter values with their operand statistical characteristic is proposed for parametric adaptation procedure realization.

*Keywords:* optimization, genetic algorithm, genetic operators, dynamic adaptation.