

УДК 519.7

## ГЕНЕТИЧЕСКИЙ АЛГОРИТМ С ТУРНИРНОЙ СЕЛЕКЦИЕЙ КАК МЕТОД ЛОКАЛЬНОГО ПОИСКА \*)

А. В. Еремеев

**Аннотация.** Найдены достаточные условия, при которых популяционный генетический алгоритм с турнирной селекцией впервые посещает локальный оптимум в среднем за полиномиально ограниченное время. Показано, что эти условия выполняются на классе задач с гарантированными локальными оптимумами при подходящем выборе параметров алгоритма.

**Ключевые слова:** генетический алгоритм, локальный поиск, приближённое решение.

### Введение

Генетический алгоритм (ГА) предложен Холландом [15] и представляет собой рандомизированный эвристический метод поиска экстремума, основанный на аналогии с генетическими механизмами в живой природе и использующий эволюционирующую популяцию пробных решений. Различные его модификации получили широкое применение в исследовании операций, распознавании образов, искусственном интеллекте и др. областях [2, 6, 16]. Несмотря на многочисленные экспериментальные исследования этих алгоритмов теоретический анализ их работоспособности находится на начальном этапе [13].

В настоящей статье генетические алгоритмы исследуются применительно к отысканию локальных оптимумов задач комбинаторной оптимизации, а точнее, задач NP-оптимизации [4, 9]. Особое внимание уделяется выделению тех ситуаций, когда ГА имеет полиномиально ограниченное в среднем время поиска локального оптимума. Здесь и далее *полиномиально ограниченной* называется величина, для которой существует некоторый полином относительно длины исходных данных задачи, ограничивающий данную величину сверху.

---

\*) Исследование выполнено при финансовой поддержке Российского фонда фундаментальных исследований (проект 10-01-00598) и президиума РАН (программа фундаментальных исследований № 2, проект № 227).

Мотивацией исследования служит тот факт, что ГА зачастую относят к классу методов локального поиска (см., например, [5, 8]), поэтому представляет интерес детальное изучение случаев, когда работоспособность ГА объясняется сходством его поведения с локальным поиском.

Статья имеет следующую структуру. Разд. 1 содержит определение класса задач NP-оптимизации, а также некоторые связанные с ним базовые определения и описание исследуемого ГА. В разд. 2 получена оценка среднего времени первого достижения локального оптимума, в разд. 3 с использованием найденной оценки исследуется среднее время получения оптимума в двух специальных семействах задач. В разд. 4 оценка из разд. 2 применяется к классу задач с гарантированными локальными оптимумами. Разд. 5 содержит заключительные замечания.

## 1. Постановка задачи и описание алгоритма

**1.1. Задачи NP-оптимизации.** Определение задачи NP-оптимизации [4, 9] формализует понятие комбинаторной или переборной задачи. Через  $\{0, 1\}^*$  обозначим множество строк из нулей и единиц произвольной длины, через  $\mathbb{N}$  — множество натуральных чисел, а через  $|S|$  — длину строки  $S \in \{0, 1\}^*$ .

**Определение 1.** *Задача NP-оптимизации* — это тройка  $\Pi = (I, \text{Sol}, f_x)$ , где  $I \subseteq \{0, 1\}^*$  — множество индивидуальных задач из  $\Pi$  и выполнены следующие условия:

(i) существует детерминированная машина Тьюринга, распознающая принадлежность строки исходных данных  $x$  множеству  $I$  за полиномиально ограниченное время;

(ii)  $\text{Sol}(x) \subseteq \{0, 1\}^{n(x)}$  — множество допустимых решений индивидуальной задачи  $x \in I$ , причём для некоторого полинома  $\text{poly}$  размерность пространства решений  $n(x) \leq \text{poly}(|x|)$ ; кроме того, для входа  $x$  и строки  $\mathbf{x} \in \{0, 1\}^*$  за полиномиально ограниченное время возможно определить принадлежность  $\mathbf{x} \in \text{Sol}(x)$ ;

(iii) для  $x \in I$  за полиномиально ограниченное время вычислима целевая функция  $f_x : \text{Sol}(x) \rightarrow \mathbb{N}$ , которую требуется максимизировать (если  $\Pi$  — задача NP-максимизации) или минимизировать (если  $\Pi$  — задача NP-минимизации).

**Определение 2.** Задача NP-оптимизации называется *полиномиально ограниченной*, если существует полином от  $|x|$ , ограничивающий значения  $f_x(\mathbf{x})$ ,  $\mathbf{x} \in \text{Sol}(x)$ .

Приближённый алгоритм для задачи NP-минимизации (NP-максимизации) имеет оценку точности  $\rho \geq 1$ , если он доставляет решение, превышающее оптимум по целевой функции не более чем в  $\rho$  раз (меньшее, чем оптимум не более чем в  $\rho$  раз) для любой индивидуальной задачи, имеющей решение.

**1.2. Окрестности и локальные оптимумы.** Пусть для любого элемента  $\mathbf{y} \in \text{Sol}(x)$  определена окрестность  $\mathcal{N}_x(\mathbf{y}) \subseteq \text{Sol}(x)$ . Совокупность  $\{\mathcal{N}_x(\mathbf{y}) : \mathbf{y} \in \text{Sol}(x)\}$  называется *системой окрестностей* [9]. Здесь и далее предполагается, что отображение  $\mathcal{N}_x$  вычислимо за полиномиально ограниченное время.

**Определение 3.** Если для  $\mathbf{x} \in \text{Sol}(x)$  при любом  $\mathbf{y} \in \mathcal{N}_x(\mathbf{x})$  выполняется неравенство  $f_x(\mathbf{y}) \leq f_x(\mathbf{x})$  в случае задачи максимизации или  $f_x(\mathbf{y}) \geq f_x(\mathbf{x})$  в случае задачи минимизации, то решение  $\mathbf{x}$  называется *локальным оптимумом в системе окрестностей*  $\mathcal{N}_x$ .

Пусть  $D(\cdot, \cdot)$  — некоторая метрика, заданная на элементах из  $\text{Sol}(x)$ . Множество  $\mathcal{N}_x(\mathbf{x}) = \{\mathbf{y} : D(\mathbf{x}, \mathbf{y}) \leq k\}$ ,  $\mathbf{x} \in \text{Sol}(x)$ , называется *системой окрестностей радиуса  $k$ , порождённой метрикой  $D(\cdot, \cdot)$* .

Алгоритм локального поиска начинает свою работу с некоторого допустимого решения. Далее на каждой итерации алгоритма происходит переход от текущего решения к новому допустимому решению в его окрестности, имеющему лучшее значение целевой функции, чем текущее решение. Процесс продолжается, пока не будет достигнут локальный оптимум. Способ выбора нового решения в окрестности текущего решения зависит от специфики конкретного алгоритма локального поиска.

Ввиду того, что из контекста далее всегда ясно, о какой индивидуальной задаче  $x$  идёт речь, для краткости обозначений символ  $x$ , как правило, будем опускать.

**1.3. Генетический алгоритм.** ГА предложен в [15] как алгоритм, имитирующий адаптацию популяции к окружающей среде, которая задана функцией приспособленности особей. Впоследствии ГА стал активно использоваться и как метод оптимизации, где функция приспособленности определяется целевой функцией задачи.

В соответствии с общепринятым подходом будем предполагать, что при  $\mathbf{x} \in \text{Sol}$ , функция приспособленности имеет вид  $\Phi(\mathbf{x}) = \varphi(f(\mathbf{x}))$ , где  $\varphi$  — некоторая строго монотонно возрастающая функция, если решается задача на максимум, или строго монотонно убывающая функция в случае задачи минимизации. Если  $\mathbf{x} \notin \text{Sol}$ , то функция приспособленности  $\Phi(\mathbf{x})$  принимает значение, меньшее, чем при любом допустимом

решении, что соответствует штрафу за нарушение ограничений задачи.

Процесс работы ГА представляет собой последовательную смену популяций (поколений), состоящих из  $N$  особей. Здесь и далее под особью понимается элемент  $\mathbf{x}$  пространства решений  $B = \{0, 1\}^n$ . Чем больше приспособленность особи, тем больше шансов она имеет оставить потомков в следующем поколении. Особи каждого следующего поколения строятся на основе особей текущей популяции некоторым случайным образом под действием операторов селекции  $\text{Sel} : B^N \rightarrow \{1, \dots, N\}$ , мутации  $\text{Mut} : B \rightarrow B$  и кроссинговера  $\text{Cross} : B \times B \rightarrow B \times B$ . Указанные операторы в общем случае могут рассматриваться как вычислимые за полиномиально ограниченное время рандомизированные процедуры [3]. Если задан аргумент такой процедуры, то распределение вероятностей на её выходе не зависит от предшествующей работы алгоритма.

Популяцию поколения  $t \geq 0$  обозначим через  $X^t = (\mathbf{x}^{1t}, \dots, \mathbf{x}^{Nt})$ . Способ нумерации особей в популяции не имеет значения. Итерацией ГА является переход от  $X^t$  к  $X^{t+1}$ . Для удобства описания алгоритма предположим, что  $N$  чётно. Приведём схему популяционного генетического алгоритма с турнирной селекцией (см., например, [16]). ГА, соответствующий этой схеме, будет обозначаться через GA.

#### ГЕНЕТИЧЕСКИЙ АЛГОРИТМ GA

ШАГ 1. Положить  $t := 0$ .

ШАГ 2. Для  $k$  от 1 до  $N$  выполнять:

ШАГ 2.1. Построить случайным образом особь  $\mathbf{x}^{k,0}$ .

Итерация  $t$

ШАГ 3. Для  $k$  от 1 до  $N/2$  выполнять шаги 3.1–3.3.

ШАГ 3.1. Селекция: выбрать особи  $\mathbf{x} := \mathbf{x}^{\text{Sel}(X^t),t}$ ,  $\mathbf{y} := \mathbf{x}^{\text{Sel}(X^t),t}$ .

ШАГ 3.2. Кроссинговер: построить  $(\mathbf{x}', \mathbf{y}') := \text{Cross}(\mathbf{x}, \mathbf{y})$ .

ШАГ 3.3. Мутация: положить  $\mathbf{x}^{2k-1,t+1} := \text{Mut}(\mathbf{x}')$ ,  
 $\mathbf{x}^{2k,t+1} := \text{Mut}(\mathbf{y}')$ .

ШАГ 4. Положить  $t := t + 1$ .

ШАГ 5. Если условие остановки не выполнено, то идти на шаг 3, иначе на шаг 6.

ШАГ 6. Результатом работы ГА является лучшая из найденных особей за все итерации.

На шаге 2 формируется начальная популяция  $X^0$ , элементы которой генерируются некоторой детерминированной или рандомизирован-

ной процедурой, например, в соответствии с равномерным распределением на  $B$ .

Оператор селекции особей имеет то же значение, что и естественный отбор в природе. Действие этого оператора состоит в выборе номера родительской особи для построения очередного потомка. В настоящей статье исследуется широко используемый оператор турнирной селекции [14]. При действии этого оператора из популяции извлекаются  $s$  особей с равномерным распределением и в качестве родителя среди них выбирается лучшая по приспособленности. Параметр  $s$  называется *размером турнира*.

Размеры популяции  $N$  и турнира  $s$ , вообще говоря, зависят от индивидуальной задачи  $x$ , и их выбор может существенно влиять на скорость сходимости популяции к решениям приемлемого качества (см., например, [10, 12]).

Условие остановки (шаг 5) можно сформулировать по-разному: например, по достижению заданной приспособленности, по достижению заданного числа итераций или по прошествии заданного числа итераций без улучшения рекорда целевой функции. Для удобства анализа будем считать, что условие остановки никогда не выполняется.

Будем предполагать, что в результате кроссинговера с вероятностью не менее  $\varepsilon$ ,  $0 < \varepsilon \leq 1$ , образуются особи  $(\mathbf{x}', \mathbf{y}') = \text{Cross}(\mathbf{x}, \mathbf{y})$ , хотя бы одна из которых не уступает по приспособленности родительским особям  $\mathbf{x}, \mathbf{y} \in B$ , т. е.

$$\mathbf{P}\{\max\{\Phi(\mathbf{x}'), \Phi(\mathbf{y}')\} \geq \max\{\Phi(\mathbf{x}), \Phi(\mathbf{y})\}\} \geq \varepsilon \quad (1)$$

при любых  $\mathbf{x}, \mathbf{y} \in B$ , причём константа  $\varepsilon$  не зависит от  $x$ .

**1.4. Примеры операторов мутации и кроссинговера.** Наибольший интерес представляют несколько вариантов мутации и кроссинговера, широко используемых в ГА и моделирующих процессы рекомбинации и мутации в живой природе. Рассмотрим наиболее известные операторы мутации  $\text{Mut}^*$  и кроссинговера  $\text{Cross}^*$  классического ГА [15].

Результат кроссинговера  $(\mathbf{x}', \mathbf{y}') = \text{Cross}^*(\mathbf{x}, \mathbf{y})$  при действии на родительские решения  $\mathbf{x} = (x_1, \dots, x_n)$ ,  $\mathbf{y} = (y_1, \dots, y_n)$  с вероятностью  $P_c$  формируется в виде

$$\mathbf{x}' = (x_1, \dots, x_\chi, y_{\chi+1}, \dots, y_n), \quad \mathbf{y}' = (y_1, \dots, y_\chi, x_{\chi+1}, \dots, x_n),$$

где случайная координата  $\chi$  выбрана с равномерным распределением от 1 до  $n - 1$ . С вероятностью  $1 - P_c$  родительские особи копируются

без изменений, т. е.  $\mathbf{x}' = \mathbf{x}$ ,  $\mathbf{y}' = \mathbf{y}$ . Этот оператор  $\text{Cross}^*$  принято называть *одноточечным кроссинговером*. Для него условие (1) выполняется при  $\varepsilon = 1 - P_c$ , если  $P_c < 1$  — константа, не зависящая от  $x$ . Условие (1) выполняется с  $\varepsilon = 1$ , если один из двух потомков — решение задачи оптимальной рекомбинации родительских решений (см., например, [10]).

При действии  $\text{Mut}^*$  вычисляется особь  $\mathbf{x}' = \text{Mut}^*(\mathbf{x})$ , где каждому биту  $x'_i$ ,  $i = 1, \dots, n$ , независимо от других с заданной вероятностью  $P_m$  присваивается значение  $1 - x_i$ , а с вероятностью  $1 - P_m$  — значение  $x_i$ .

Степень воздействия кроссинговера и мутации регулируется параметрами  $P_c$  и  $P_m$ , которые, вообще говоря, могут зависеть от  $x$ . Увеличение вероятности мутации до 0.5 превращает ГА в простой случайный перебор. Уменьшение  $P_m$  до нуля приводит к малому разнообразию в популяции и может вызвать «зацикливание» ГА, когда на каждой итерации генерируются лишь ранее встречавшиеся особи.

## 2. Среднее время достижения локального оптимума

Пусть имеется задача NP-максимизации  $\Pi = (I, \text{Sol}, f_x)$  и задано некоторое семейство окрестностей  $\mathcal{N}$ . Пусть  $h = |\{f(\mathbf{x}) : \mathbf{x} \in \text{Sol}\}| - 1$ , т. е.  $h$  — число всех неоптимальных значений целевой функции  $f$ . Тогда, начиная с любого решения, локальный поиск достигает локального оптимума не более чем за  $h$  улучшающих целевую функцию итераций. Пусть  $L$  обозначает минимальную вероятность достижения решения в пределах окрестности:

$$L = \min_{\mathbf{x} \in \text{Sol}, \mathbf{x}' \in \mathcal{N}(\mathbf{x})} \mathbf{P}\{\text{Mut}(\mathbf{x}) = \mathbf{x}'\}.$$

Чем выше величина  $L$ , тем больше согласованность оператора мутации  $\text{Mut}$  с системой окрестностей  $\mathcal{N}$ . Численность популяции  $N$ , размер турнира  $s$  и величину  $L$  будем рассматривать как функции от исходных данных задачи  $x$ . Пусть  $e$  — число Эйлера.

**Лемма 1.** Если  $X^0$  содержит допустимое решение,  $s \geq rN$ ,  $r > 0$ ,  $h > 1$ ,  $L > 0$  и

$$N \geq \frac{2(1 + \ln h)}{L\varepsilon(1 - 1/e^{2r})}, \quad (2)$$

то (i) ГА посещает локальный оптимум к итерации  $h$  с вероятностью не менее  $1/e$ , и (ii) локальный оптимум достигается не позднее, чем за  $eh$  итераций ГА в среднем.

**ДОКАЗАТЕЛЬСТВО.** Заметим, что в начальной популяции особь с наибольшей приспособленностью является допустимым решением, так как

приспособленность особей, отвечающих недопустимым решениям, всегда меньше приспособленности решений из допустимой области. Пусть событие  $E_k^{t+1}$ ,  $k = 1, \dots, N/2$ , состоит в выполнении следующих трёх условий:

- (а) из популяции  $X^t$  при построении  $k$ -й пары потомков следующего поколения выбирается решение  $\mathbf{x}_*^t$  наибольшей приспособленности в  $X^t$ ;
- (б) при построении  $k$ -й пары потомков посредством кроссинговера один из них имеет приспособленность не менее  $\Phi(\mathbf{x}_*^t)$  (пусть для определённости это  $\mathbf{x}'$ );
- (с) оператор мутации, применённый к  $\mathbf{x}'$ , осуществляет переход в наилучшее по приспособленности решение в окрестности  $\mathcal{N}(\mathbf{x}')$ , т. е.

$$\Phi(\text{Mut}(\mathbf{x}')) = \max_{\mathbf{y} \in \mathcal{N}(\mathbf{x}')} \Phi(\mathbf{y}).$$

Обозначим через  $p$  вероятность наступления хотя бы одного из событий  $E_k^{t+1}$ ,  $k = 1, \dots, N/2$ , при известной популяции  $X^t$ . Найдём оценку  $\lambda \leq p$ , не зависящую от выбора  $X^t$ . Согласно схеме GA  $\mathbf{P}\{E_1^{t+1}\} = \dots = \mathbf{P}\{E_{N/2}^{t+1}\}$ , обозначим эту вероятность через  $q$ . Ввиду независимости событий  $E_k^{t+1}$ ,  $k = 1, \dots, N/2$ , при фиксированной  $X^t$  имеем

$$p \geq 1 - (1 - q)^{N/2} \geq 1 - e^{-qN/2}.$$

Оценим снизу вероятность  $q$ :

$$q \geq L\varepsilon(1 - (1 - 1/N)^{2s}).$$

Однако  $(1 - 1/N)^{2s} \leq (1 - 1/N)^{2rN} \leq 1/e^{2r}$ , поэтому

$$q \geq L\varepsilon(1 - 1/e^{2r}) = Lc,$$

где  $c = \varepsilon(1 - 1/e^{2r})$ . Далее воспользуемся тем, что из последнего неравенства и из (2) вытекает

$$N \geq \frac{2}{L\varepsilon(1 - 1/e^{2r})} \geq 2/q. \quad (3)$$

Оценим снизу вероятность  $p$ . Сначала заметим, что  $1 - z/e \geq e^{-z}$  при любом  $z \in [0, 1]$  и положим  $z = e^{-qN/2+1}$ . Тогда ввиду неравенств (3) и  $z \leq 1$  получим  $p \geq \exp\{-e^{1-qN/2}\} \geq \exp\{-e^{1-LcN/2}\}$ .

От анализа потомков фиксированной популяции  $X^t$  перейдём к случайной последовательности популяций  $X^0, X^1, \dots$ . Заметим, что  $\lambda^h$  является оценкой снизу для вероятности достичь локальный оптимум за серию из не более  $h$  итераций, улучшающих значение рекорда целевой

функции. Действительно, пусть  $A_t = E_1^t + \dots + E_{N/2}^t$ ,  $t = 1, 2, \dots$ . Тогда

$$\mathbf{P}\{A_1 \& \dots \& A_h\} = \mathbf{P}\{A_1\} \prod_{t=1}^{h-1} \mathbf{P}\{A_{t+1} \mid A_1 \& \dots \& A_t\} \geq \lambda^h. \quad (4)$$

Итак, положим  $\lambda = \exp\{-e^{1-LcN/2}\}$ . Снова воспользовавшись условием (2), получаем оценку снизу для вероятности достичь локальный оптимум за серию из не более  $h$  улучшающих рекорд итераций:

$$\lambda^h = \exp\{-he^{1-LcN/2}\} \geq \exp\{-he^{-\ln h}\} = 1/e.$$

Утверждение (i) леммы доказано.

Для оценки среднего времени получения локального оптимума рассмотрим последовательность серий по  $h$  итераций в каждой. Пусть событием  $D_i$ ,  $i = 1, 2, \dots$ , является отсутствие локального оптимума в популяции в  $i$ -й серии. При выполнении условий леммы вероятность каждого события  $D_i$ ,  $i = 1, 2, \dots$ , не превышает  $\mu = 1 - 1/e$  при любой предыстории работы алгоритма. По аналогии с (4) заключаем, что

$$\mathbf{P}\{D_1 \& \dots \& D_k\} \leq \mu^k.$$

Таким образом, если через  $\eta$  обозначить случайную величину, равную номеру первой серии, на которой локальный оптимум получен, то, пользуясь свойствами математического ожидания (см., например, [1]), получим

$$E[\eta] = \sum_{i=0}^{\infty} \mathbf{P}\{\eta > i\} = 1 + \sum_{i=1}^{\infty} \mathbf{P}\{D_1 \& \dots \& D_i\} \leq 1 + \sum_{i=1}^{\infty} \mu^i = e.$$

Следовательно, локальный оптимум достигается не позднее, чем за  $eh$  итераций GA в среднем. Лемма 1 доказана.

Пусть через  $\lceil \cdot \rceil$  обозначается округление вверх. Тогда в условиях леммы выбор

$$N = 2 \left\lceil \frac{1 + \ln h}{L\varepsilon(1 - 1/e^{2r})} \right\rceil, \quad s = \lceil rN \rceil \quad (5)$$

обеспечивает получение локального оптимума в GA за  $O(h)$  итераций в среднем.

Трудоёмкости операторов Mut и Cross полиномиально ограничены, а процедура турнирной селекции требует времени  $O(s) = O(N)$ . Следовательно, имеет место



**Теорема 1.** Если задача  $\Pi = (I, \text{Sol}, f_x)$  и функция  $L^{-1}(x)$  полиномиально ограничены, а популяция  $X^0$  алгоритма ГА всегда содержит допустимое решение, то при размере популяции и размере турнира, выбранных согласно (5), ГА впервые посещает локальный оптимум за полиномиально ограниченное в среднем время.

Заметим, что сделанное в разд. 1 предположение о полиномиальной вычислимости отображения  $\mathcal{N}$  не гарантирует того, что мощность окрестности полиномиально ограничена, т. е.  $|\mathcal{N}(\mathbf{x})|$  при любом  $\mathbf{x} \in \text{Sol}$  ограничена полиномом от  $|x|$ . Если отображение  $\mathcal{N}$  удовлетворяет последнему условию, то его называют *полиномиально ограниченным* [17]. Для такого  $\mathcal{N}$  существует оператор мутации  $\text{Mut}(\mathbf{x})$ , вычисляемый за полиномиально ограниченное время и задающий равномерное распределение особей-потомков на множестве  $\mathcal{N}(\mathbf{x})$  при заданном  $\mathbf{x}$ . Тогда  $1/L$  также ограничена сверху некоторым полиномом от  $|x|$ . Таким образом, теорема 1 применима ко многим известным семействам окрестностей для задач NP-оптимизации.

Пусть  $\delta(\mathbf{x}, \mathbf{y})$  обозначает расстояние Хэмминга между двоичными строками  $\mathbf{x}$  и  $\mathbf{y}$ .

**Определение 4.** Пусть  $\Pi$  — задача NP-оптимизации. Семейство окрестностей  $\mathcal{N}$  называется *k-ограниченным* [9], если для любых  $\mathbf{x} \in \text{Sol}$  и  $\mathbf{y} \in \mathcal{N}(\mathbf{x})$  выполнено  $\delta(\mathbf{x}, \mathbf{y}) \leq k$ , где  $k$  — константа.

Оператор мутации  $\text{Mut}^*$  классического ГА [15] строит строку  $\mathbf{y}$  при мутации  $\mathbf{x}$  с вероятностью  $P_m^{\delta(\mathbf{x}, \mathbf{y})} (1 - P_m)^{n - \delta(\mathbf{x}, \mathbf{y})}$ . Заметим, что вероятность  $P_m^k (1 - P_m)^{n - k}$  как функция от  $P_m \in [0, 1]$  достигает своего максимума при  $P_m = k/n$ . Следующее утверждение даёт оценку снизу вероятности  $\mathbf{P}\{\text{Mut}^*(\mathbf{x}) = \mathbf{y}\}$  для любого  $\mathbf{y} \in \mathcal{N}(\mathbf{x})$  при оптимальном в указанном смысле выборе  $P_m = k/n$ .

**Утверждение 1.** Пусть семейство окрестностей  $\mathcal{N}$  k-ограничено и  $k \leq n/2$ . Тогда при  $P_m = k/n$  для любых  $\mathbf{x} \in \text{Sol}$  и  $\mathbf{y} \in \mathcal{N}(\mathbf{x})$

$$\mathbf{P}\{\text{Mut}^*(\mathbf{x}) = \mathbf{y}\} \geq \left(\frac{k}{en}\right)^k.$$

ДОКАЗАТЕЛЬСТВО. При  $\mathbf{x} \in \text{Sol}$  и  $\mathbf{y} \in \mathcal{N}(\mathbf{x})$  имеем

$$\mathbf{P}\{\text{Mut}^*(\mathbf{x}) = \mathbf{y}\} = \left(\frac{k}{n}\right)^{\delta(\mathbf{x}, \mathbf{y})} \left(1 - \frac{k}{n}\right)^{n - \delta(\mathbf{x}, \mathbf{y})} \geq \left(\frac{k}{n}\right)^k \left(1 - \frac{k}{n}\right)^{n - k},$$

так как  $P_m = k/n \leq 1/2$ . Далее,  $\partial(1 - k/n)^{n - k} / \partial n < 0$  при  $n > k$ , кроме

того,  $(1 - k/n)^{n-k} \rightarrow 1/e^k$  при  $n \rightarrow \infty$ . Следовательно,  $(1 - k/n)^{n-k} \geq 1/e^k$ , откуда вытекает требуемое неравенство. Утверждение 1 доказано.

### 3. Среднее время получения оптимума в двух специальных семействах задач

**3.1. Семейство задач ONEMAX\*\*.** В качестве примера применения леммы 1 и утверждения 1 рассмотрим семейство одноэкстремальных задач, часто используемых при анализе эволюционных алгоритмов [11].

Определим семейство ONEMAX\* целевых функций вида  $\delta(\mathbf{x}, a)$ , где  $a \in \{0, 1\}^n$  — оптимальное решение. Тогда ONEMAX\*\* определяется как семейство функций  $\varphi \circ g$ , где  $g \in \text{ONEMAX}^*$ ,  $\varphi : \mathbb{Z}^+ \rightarrow \mathbb{R}$  — строго монотонно возрастающая функция,  $\mathbb{Z}^+$  — множество неотрицательных целых чисел, а  $\mathbb{R}$  — множество вещественных чисел. Требуется максимизировать функцию  $f \in \text{ONEMAX}^{**}$  на множестве  $\text{Sol} = B$ . Заметим, что без потери общности достаточно рассматривать  $\varphi$  с натуральными значениями. В качестве функции приспособленности для  $f \in \text{ONEMAX}^{**}$  естественно выбрать  $\Phi(\mathbf{x}) \equiv f(\mathbf{x})$ .

В системе окрестностей, порождённой метрикой Хэмминга радиуса 1, точка  $a$  является единственным локальным оптимумом, а значит, и глобальным. Согласно утверждению 1 при  $P_m = 1/n$ ,  $n > 1$ , для любых  $\mathbf{x} \in \text{Sol}$  и  $\mathbf{y} \in \mathcal{N}(\mathbf{x})$

$$\mathbf{P}\{\text{Mut}^*(\mathbf{x}) = \mathbf{y}\} \geq 1/(en).$$

По лемме 1 заключаем, что при  $s = rN$ ,  $r > 0$ , и  $N = \lceil \frac{2en(1+\ln n)}{\varepsilon(1-1/e^{2r})} \rceil$  ГА впервые посещает оптимум в среднем не позднее, чем за  $en$  итераций. Трудоемкость турнирной селекции в таком ГА составляет  $O(N)$ . Если предположить, что кроссинговер, мутация и вычисление целевой функции могут быть выполнены за некоторое время  $T$ , то ГА впервые посещает оптимум в среднем за время  $O(Tn^3 \ln^2 n)$ . В классическом ГА [15], например,  $T = O(n)$ , т. е. оптимум впервые достигается в среднем за время  $O(n^4 \ln^2 n)$ .

**3.2. Семейство задач вершинного покрытия графа  $G(\ell)$ .** Задача о наименьшем вершинном покрытии (ЗВП) формулируется следующим образом. Пусть  $G = (V, E)$  — граф с множествами вершин  $V$  и рёбер  $E$ . Подмножество  $C \subseteq V$  называется *вершинным покрытием*, если каждое ребро из  $E$  инцидентно хотя бы одной вершине из  $C$ . Требуется найти вершинное покрытие наименьшей мощности.

Рассмотрим семейство ЗВП специального вида, где граф  $G(\ell)$  состоит из  $\ell$  трёхвершинных полных подграфов, не связанных между собой,

$\ell = 1, 2, \dots$ . Очевидно, что каждый из треугольных подграфов оптимально покрывается двумя вершинами, а неоптимально – тремя. Несмотря на простоту данной задачи, как показано в [7], некоторые известные алгоритмы типа ветвей и границ имеют экспоненциально возрастающую трудоёмкость с увеличением числа вершин графа.

Пусть вершины и рёбра графа пронумерованы, а каждому ребру  $e_i$  из  $E$  при кодировке решений из множества  $\text{Sol}$  сопоставлен один бит  $\mathbf{x}_i$ ,  $i = 1, \dots, n$ ,  $n = |E|$ . Предположим, что для каждого  $e_i$  нулевое значение бита  $\mathbf{x}_i$  означает включение в множество  $C(\mathbf{x})$  вершины, инцидентной  $e_i$ , с меньшим порядковым номером, а значение  $\mathbf{x}_i = 1$  означает включение вершины с большим номером. Очевидно, при таком способе представления решений любая строка  $\mathbf{x} \in B$  кодирует допустимое решение, т. е.  $C(\mathbf{x})$  — покрытие.

В качестве функции приспособленности для ЗВП достаточно выбрать строго монотонно убывающую функцию от  $|C(\mathbf{x})|$ . Пусть  $\Phi(\mathbf{x}) = |V| - |C(\mathbf{x})|$ . В таком случае для графа  $G(\ell)$  значение  $\Phi(\mathbf{x})$  совпадает с числом компонент связности, оптимально покрытых вершинами  $C(\mathbf{x})$ .

В системе окрестностей, порождённой метрикой Хэмминга радиуса 1, любой локальный оптимум является и глобальным. Аналогично предыдущему примеру получаем  $\mathbf{P}\{\text{Mut}^*(\mathbf{x}) = \mathbf{y}\} \geq 1/en$ , где  $n = 3\ell$ . По лемме 1 при  $s = rN$ , где  $r > 0$  и  $N = \lceil \frac{6e\ell(1+\ln \ell)}{\varepsilon(1-1/e^{2r})} \rceil$ , ГА впервые посещает оптимум в среднем не позднее, чем за  $e\ell$  итераций, т. е. среднее время его работы до получения оптимума полиномиально ограничено.

#### 4. Анализ задач с гарантированными локальными оптимумами

Применим теорему 1 для оценки возможностей генетического алгоритма отыскивать приближённые решения с априорной оценкой точности.

**Определение 5.** Пусть  $\Pi$  — полиномиально ограниченная задача NP-оптимизации. Задача  $\Pi$  принадлежит классу задач с *гарантированными локальными оптимумами* (GLO) [9], если выполняются следующие два условия:

- (i) по крайней мере одно допустимое решение  $\mathbf{y}_x \in \text{Sol}$  для любого входа  $x \in I$  может быть вычислено за полиномиально ограниченное время;
- (ii) существует  $k \in \mathbb{N}$  такое, что все локальные оптимумы задачи  $\Pi$  относительно некоторого  $k$ -ограниченного семейства окрестностей имеют константную оценку точности.

Примерами задач из класса GLO являются задача о максимальной выполнимости булевой формулы, задачи о наибольшем независимом множестве, о наименьшем доминирующем множестве вершин и о наименьшем вершинном покрытии в графах со степенью вершин, ограниченной константой, а также задача о максимальном разрезе графа [9].

Если задача NP-оптимизации  $\Pi$  лежит в классе GLO, то ввиду утверждения 1 при  $k \leq n/2$  для любых  $\mathbf{x} \in \text{Sol}$  и  $\mathbf{y} \in \mathcal{N}(\mathbf{x})$  оператор мутации  $\text{Mut}^*$  при  $P_m = k/n$  удовлетворяет условию

$$\mathbf{P}\{\text{Mut}^*(\mathbf{x}) = \mathbf{y}\} \geq 1/\text{poly}(|x|),$$

где  $\text{poly}$  — некоторый полином. Вероятность  $\mathbf{P}\{\text{Mut}^*(\mathbf{x}) = \mathbf{y}\}$  ограничена снизу положительной константой при  $n/2 < k < n$ . Таким образом, из теоремы 1 вытекает

**Следствие 1.** Пусть  $\Pi$  — задача из класса GLO,  $n > k$  и популяция  $X^0$  содержит  $\mathbf{y}_x$ . Тогда ГА с оператором мутации  $\text{Mut}^*$  при  $P_m = k/n$  и параметрах  $N$  и  $s$ , выбранных согласно (5), получает решение с константной оценкой точности в среднем за полиномиально ограниченное время.

### Заключение

Основные результаты устанавливают новые свойства генетических алгоритмов с точки зрения локальной оптимизации. Полученные оценки трудоёмкости в среднем и предложенные значения настраиваемых параметров позволяют судить о тенденциях генетических алгоритмов, связанных с ростом размера турнира и численности популяции.

### ЛИТЕРАТУРА

1. Гнеденко Б. В. Курс теории вероятностей. — М: Наука, 1988. — 451 с.
2. Загоруйко Н. Г. Самообучающийся генетический алгоритм прогнозирования (GAP) // Искусств. интеллект и эксперт. системы. Вып. 160. — Новосибирск: Вычисл. системы, 1997. — С. 3–17.
3. Китаев А., Шень А., Вялый М. Классические и квантовые вычисления. — М: МЦНМО, ЧеРо, 1999. — 192 с.
4. Кочетов Ю. А. Вычислительные возможности локального поиска в комбинаторной оптимизации // Журн. вычисл. математики и мат. физики. — 2008. — Т. 48, № 5. — С. 788–807.
5. Кочетов Ю. А. Вероятностные методы локального поиска для задач дискретной оптимизации // Дискрет. математика и её приложения: Сб. лекций молодежных научных школ по дискретной математике и её приложениям. — М: Изд-во центра прикл. исслед. при мех.-мат. фак. МГУ, 2001. — С. 84–117.

6. Рутковская Д., Пилиньский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечёткие системы. — М: Горячая линия — Телеком, 2006. — 452 с.
7. Сайко Л. А. Исследование мощности  $L$ -накрытий некоторых задач о покрытии // Дискрет. оптимизация и анализ сложных систем: Сб. науч. тр. — Новосибирск: ВЦ СО АН СССР, 1989. — С. 76–97.
8. Aarts E. H. L., Lenstra J. K. Introduction // Local search in combinatorial optimization. — New York: John Wiley & Sons Ltd., 1997. — P. 1–19.
9. Ausiello G., Protasi M. Local search, reducibility and approximability of NP-optimization problems // Inf. Process. Lett. — 1995. — Vol. 54. — P. 73–79.
10. Balas E., Niehaus W. Optimized crossover-based genetic algorithms for the maximum cardinality and maximum weight clique problems // J. Heurist. — 1998. — Vol. 4, N 2. — P. 107–122.
11. Droste S., Jansen T., Wegener I. Upper and lower bounds for randomized search heuristics in black-box optimization // Theory Comput. Syst. — 2006. — Vol. 39, N 4. — P. 525–544.
12. Ereemeev A. V. Modeling and analysis of genetic algorithm with tournament selection // Proc. Artificial Evolution Conf. (Dunkerque, France, November 3–5, 1999). — Berlin: Springer-Verl., 2000. — P. 84–95. (Lect. Notes Comp. Sci.; Vol. 1829.)
13. Ereemeev A. V., Reeves C. R. Evolutionary algorithms in discrete optimization // Discrete Optimization and Operations Research Conf. (Novosibirsk, June 24–28, 2002). Abstr. — Novosibirsk, IM Press, 2000. — P. 40–45.
14. Goldberg D. E. A note on Boltzmann tournament selection for genetic algorithms and population-oriented simulated annealing // Complex Syst. — Vol. 4. — P. 445–460.
15. Holland J. Adaptation in natural and artificial systems. — Ann Arbor: Univ. Michigan Press, 1975. — 183 p.
16. Reeves C. R. Genetic algorithms for the operations researcher // INFORMS J. Comput. — 1997. — Vol. 9, N 3. — P. 231–250.
17. Rödl V., Tovey C. Multiple optima in local search // J. Algorithms. — 1987. — Vol. 8. — P. 250–259.

Еремеев Антон Валентинович,  
e-mail: eremeev@ofim.oscsbras.ru

Статья поступила  
18 июня 2011 г.

Переработанный вариант —  
2 августа 2011 г.