Вполне полиномиальная рандомизированная аппроксимационная схема на основе эволюционного алгоритма*

Еремеев А.В.

10 июля 2010 г.

Для класса задач дискретной оптимизации, удовлетворяющих условиям существования вполне полиномиальной аппроксимационной схемы Г. Воигенгера, предложена вполне полиномиальная рандомизированная аппроксимационная схема, основанная на эволюционном алгоритме.

1 Введение

Эволюционные алгоритмы (ЭА), такие как генетические алгоритмы, эволюционные стратегии и алгоритмы генетического программирования получили широкое применение при решении практических задач оптимизации [5, 13, 14, 17, 18]. Характерной особенностью этих алгоритмов является имитация процесса эволюции биологической популяции, где особи соответствуют пробным точкам в пространстве решений задачи, а приспособленность особей к условиям окружающей среды соответствует значениям целевой функции в этих точках. Процесс случайного поиска в ЭА эвристически направляется значениями целевой функции в просмотренных точках пространства решений. Новые решения-потомки строятся посредством случайных операторов, модифицирующих полученные ранее пробные точки.

Теоретически ЭА до сих пор изучены недостаточно, несмотря на активные исследования. Обзоры литературы по теории ЭА могут быть найдены в [6, 19, 21]. Особый интерес с точки зрения теоретического анализа представляет сравнение ЭА с классическими методами дискретной оптимизации. В последние годы новые результаты получены при анализе ЭА для таких базовых задач дискретной оптимизации как задача об остовном дереве минимального веса [16], задача о наибольшем паросочетании [10] и задача о минимальном разрезе в графе [15]. Известны полиномиальные в среднем ЭА для поиска приближенных решений NP-трудных задач, например, для задачи о покрытии множества [9]. В [20] предложена общая схема ЭА с полиномиальным в среднем временем поиска решения для задач оптимизации на матроидах.

*Работа поддержана грантом РФФИ 07-01-00410.

Для ряда задач дискретной оптимизации известны эволюционные алгоритмы, имеющие среднее время получения оптимального решения, близкое к трудоемкости алгоритмов динамического программирования (ДП). В частности, такие алгоритмы предложены для задачи о кратчайшем пути в графе [22], задачи коммивояжера [23], задачи об одномерном булевом рюкзаке [2] и целого класса задач [8], включающего задачи многокритериальной оптимизации. Возможность построения приближенных решений с использованием ЭА, аналогичного предложенному в [8], рассматривается в настоящей работе.

Под *ρ-приближенным решением* понимается решение, значение целевой функции которого не более чем в ρ раз превышает оптимум, если рассматривается задача минимизации, или не более чем в ρ раз уступает оптимуму, если рассматривается задача максимизации. Алгоритм, находящий ρ-приближенное решение, когда задача разрешима, будем называть *ρ-приближенным алгоритмом*.

Семейство $(1+\varepsilon)$-приближенных алгоритмов с временной сложностью, полиномиально зависящей от длины входа задачи и от $1/\varepsilon$ при любом $0 < \varepsilon < 1$, называется *вполне полиномиальной аппроксимационной схемой*. В [24] Г. Войгенгером предложены условия, при которых для достаточно широкого класса задач, на основе уже известного алгоритма ДП может быть построена вполне полиномиальная аппроксимационная схема. В этот класс, например, входит задача о булевом одномерном рюкзаке и такие задачи теории расписаний, как задача минимизации средневзвешенного времени завершения работ на фиксированном числе параллельных машин разной производительности, задача минимизации средневзвешенного отклонения от заданного срока завершения работ на одной машине, задача минимизации взвешенного суммарного запаздывания деталей и др.

В настоящей работе для класса задач, удовлетворяющих условиям Г. Войгенгера, на основе ЭА [8] построено семейство алгоритмов, являющееся *вполне полиномиальной рандомизированной аппроксимационной схемой*. Под вполне полиномиальной рандомизированной аппроксимационной схемой здесь и далее подразумевается семейство вероятностных алгоритмов при всевозможных $0 < \varepsilon < 1$ с временной сложностью, полиномиально зависящей от длины входа задачи и от $1/\varepsilon$, дающих $(1+\varepsilon)$-приближенное решение с вероятностью не менее $3/4$. Заметим, что вероятность отсутствия $(1+\varepsilon)$-приближенного решения с ростом числа независимых испытаний вероятностного алгоритма убывает экспоненциально, поэтому вместо $3/4$ в данном случае может использоваться любая константа из интервала (0,1).

2 Постановка задачи и используемые подходы

Пусть Π – задача комбинаторной оптимизации (см., например, [1]), I обозначает исходные данные ее индивидуальной задачи, $\mathcal{D}_I$ – множество допустимых решений, $m_I : \mathcal{D}_I \to \mathrm{Z}_+$ – целевая функция (здесь и далее Z_+ обозначает множество неотрицательных целых чисел). Оптимальное значение целевой функции для индивидуальной задачи I обозначим через $OPT(I) = \max_{y \in \mathcal{D}_I} m_I(y)$, если рассматривается задача максимизации, либо $OPT(I) = \min_{y \in \mathcal{D}_I} m_I(y)$ в случае задачи минимизации. Сделаем дополнительное предположение относительно структуры входных данных Π: вход I состоит из n векторов $X_1, \dots, X_n \in \mathrm{Z}_+^{\alpha}$ и целочисленные компоненты $x_{1k}, \dots, x_{\alpha k}$ каждого из векторов

X_k представлены в двоичной кодировке. Размерность α может зависеть от конкретного входа задачи. Таким образом, длина входа $|I|$ составляет $\Theta\left(n+\alpha+\log\left(\sum_{k=1}^{n}\sum_{i=1}^{\alpha}x_{ik}\right)\right)$.

Рассмотрим задачу П с точки зрения возможности построения алгоритма динамического программирования для нее. При этом будем пользоваться подходом, предложенным Г. Воегингером [24].

Предположим, что для задачи П существует точный алгоритм ДП, работа которого состоит из n стадий. На k-й стадии ДП, $k = 1,\dots,n$, обрабатывается входной вектор X_k и строится множество состояний $\mathcal{S}_k$ (эти состояния в некотором смысле соответствуют частичным решениям задачи П). Каждый элемент множества $\mathcal{S}_k$ – вектор вида $S = (s_1,\dots,s_\beta) \in \mathrm{Z}_+^\beta$. Размерность β фиксирована для задачи П и не зависит от индивидуальной задачи I.

Пусть конечное множество $\mathcal{F}$ отображений перехода от состояний текущей стадии $k-1$ к состояниям следующей стадии k, имеющих вид $F : \mathrm{Z}_+^\alpha \times \mathrm{Z}_+^\beta \to \mathrm{Z}_+^\beta$, а конечное множество $\mathcal{H}$ состоит из вспомогательных функционалов вида $H : \mathrm{Z}_+^\alpha \times \mathrm{Z}_+^\beta \to Q$. Здесь Q – множество рациональных чисел. Первый аргумент указанных отображений далее будет представлять собой k-ый вектор исходных данных X_k, а второй аргумент – вектор состояния из уже построенной стадии $\mathcal{S}_{k-1}$. Для каждого отображения $F \in \mathcal{F}$ имеется соответствующее ему отображение $H_F \in \mathcal{H}$.

На этапе инициализации ДП множество состояний $\mathcal{S}_0$ формируется как некоторое конечное подмножество пространства Z_+^β. На стадии k, $k = 1,\dots,n$ пространство состояний $\mathcal{S}_k$ вычисляется на основе множества $\mathcal{S}_{k-1}$:

$$\mathcal{S}_k = \{F(X_k, S) : \ F \in \mathcal{F}, \ S \in \mathcal{S}_{k-1}, \ H_F(X_k, S) \le 0\}. \tag{1}$$

Отображения из $\mathcal{H}$ служат для отбрасывания некоторых неперспективных состояний (как правило, они выражают некоторые ограничения задачи П).

Назовем *допустимой траекторией* последовательность состояний $S_0,\dots,S_n$, для которой при каждом $k = 1,\dots,n$ выполняется $S_k \in \mathcal{S}_k$, $S_k = F_k(X_k, S_{k-1})$, $F_k \in \mathcal{F}$ и $H_{F_k}(X_k, S_{k-1}) \le 0$. Далее будем предполагать, что всякая допустимая траектория $S_1,\dots,S_n$ представляет некоторое допустимое решение задачи П, т.е. $(S_1,\dots,S_n) \in \mathcal{D}_I$, а значение целевой функции при этом может быть вычислено на элементах последнего множества состояний $\mathcal{S}_n$ посредством функции $G : \mathrm{Z}_+^\beta \to \mathrm{Z}_+$ т.е. $m_I(S_1,\dots,S_n) = G(S_n)$.

Сделанное ранее предположение о том, что ДП доставляет точное решение задачи П можно выразить формально:

$$OPT(I) = \min\{G(S) : S \in \mathcal{S}_n\}, \tag{2}$$

если П – задача минимизации, либо

$$OPT(I) = \max\{G(S) : S \in \mathcal{S}_n\}, \tag{3}$$

если это задача максимизации.

2.1 Достаточные условия существования вполне полиномиальной аппроксимационной схемы

При построении вполне полиномиальных аппроксимационных схем для «прореживания» множества состояний на каждой стадии ДП могут исключаться некоторые состояния, близкие к уже построенным. Одной из первых работ, где использовался данный подход, является [12]. Другие примеры его использования могут быть найдены в [3, 7, 24].

Для формализации понятия близости состояний, следуя [24], предположим что задан *вектор степеней* $D = (d_1, \dots, d_\beta) \in \mathrm{Z}_+^\beta$, не зависящий от индивидуальной задачи I. При заданном вещественном масштабирующем множителе $\Delta > 1$ будем полагать, что состояние $S = (s_1, \dots, s_\beta)$ (D, Δ)-*близко* к состоянию $S' = (s'_1, \dots, s'_\beta)$, если

$$\Delta^{-d_\ell} s_\ell \le s'_\ell \le \Delta^{d_\ell} s_\ell, \quad \ell = 1, \dots, \beta.$$

Введем обозначение $\mathcal{L}_0$ для множества индексов $1 \le \ell \le \beta$, таких что $d_\ell = 0$, и пусть $\mathcal{L}_1 = \{1, \dots, \beta\} \backslash \mathcal{L}_0$.

Вычислительная сложность ДП, как правило, сокращается за счет исключения из рассмотрения «неперспективных» состояний, например, в точных алгоритмах ДП для этого чаще всего используется принцип Беллмана. Аналогичные принципы используются в алгоритме «киевский веник» (см., например, [4]). В настоящей работе, подобно [24], исключение «неперспективных» состояний осуществляется посредством отношения линейного предпорядка $\preceq_{\mathrm{qua}}$ на множестве состояний, с учетом их близости. При таком «прореживании» на каждой стадии k, $k = 1, \dots, n$, могут удаляться только те из состояний, которые уступают какому-либо уже построенному на этой стадии состоянию в смысле отношения $\preceq_{\mathrm{qua}}$ и при этом (D, Δ)-близки к нему.

Для большей общности результата [24] наряду с линейным предпорядком $\preceq_{\mathrm{qua}}$ на множестве состояний рассматривается частичный порядок $\preceq_{\mathrm{dom}}$, такой что $\preceq_{\mathrm{qua}}$ является некоторым его продолжением, т.е. из $S \preceq_{\mathrm{dom}} S'$, где $S, S' \in \mathrm{Z}_+^\beta$, следует $S \preceq_{\mathrm{qua}} S'$. Будем говорить, что $S \in Z^\beta$ *доминируется* состоянием $S' \in Z^\beta$, если $S \preceq_{\mathrm{dom}} S'$. Если $S \in \mathcal{S} \subseteq Z^\beta$ такое, что ни для какого $S' \in \mathcal{S}$ не выполняется $S \preceq_{\mathrm{dom}} S'$, тогда S будем называть *недоминируемым* в $\mathcal{S}$.

Следующие четыре условия [24] обеспечивают существование вполне полиномиальной аппроксимационной схемы. Условия C1 и C2 связывают свойства функций из $\mathcal{F}$ и $\mathcal{H}$ с отношениями $\preceq_{\mathrm{dom}}$ и $\preceq_{\mathrm{qua}}$:

C1. Для любых $\Delta > 1$, $F \in \mathcal{F}$, $X \in \mathrm{Z}_+^\alpha$, $S, S' \in \mathrm{Z}_+^\beta$,
(i) если S (D, Δ)-близко к S' и $S \preceq_{\mathrm{qua}} S'$, то либо $F(X, S) \preceq_{\mathrm{qua}} F(X, S')$ и $F(X, S)$ (D, Δ)-близко к $F(X, S')$, либо $F(X, S) \preceq_{\mathrm{dom}} F(X, S')$;
(ii) если $S \preceq_{\mathrm{dom}} S'$, то $F(X, S) \preceq_{\mathrm{dom}} F(X, S')$.

C2. Для любых $\Delta > 1$, $H \in \mathcal{H}$, $X \in \mathrm{Z}_+^\alpha$, $S, S' \in \mathrm{Z}_+^\beta$,
(i) если S (D, Δ)-близко к S' и $S \preceq_{\mathrm{qua}} S'$, то $H(X, S') \le H(X, S)$;
(ii) если $S \preceq_{\mathrm{dom}} S'$, то $H(X, S') \le H(X, S)$.

Заметим, что ввиду произвольности $\Delta > 1$, пункт (i) условия C2 сводится к требованию выполнения неравенства $H(X, S') \le H(X, S)$ для всех $H \in \mathcal{H}$, $X \in \mathrm{Z}_+^\alpha$ и $S, S' \in \mathrm{Z}_+^\beta$, $S \preceq_{\mathrm{qua}} S'$, таких что $s_\ell = s'_\ell$ для всех $\ell \in \mathcal{L}_0$.

Далее, условие C3 связывает свойства функции G с отношениями доминирования и близости состояний на последнем множестве состояний $\mathcal{S}_n$:

C3.

(i) Существует $g \in Z_+$, зависящее только от функции G и вектора D, такое что для любых $\Delta > 1$ и $S, S' \in Z^\beta_+$, если S (D, Δ)-близко к S' и $S \preceq_{\text{qua}} S'$, то в случае минимизационной задачи, $G(S') \le \Delta^g G(S)$, а в случае задачи максимизации, $\Delta^{-g} G(S) \le G(S')$.

(ii) Для любых $S, S' \in Z^\beta_+$, если $S \preceq_{\text{dom}} S'$, то в случае минимизационной задачи, $G(S') \le G(S)$, а в случае максимизационной задачи, $G(S') \ge G(S)$.

Наконец, последнее условие C4 обеспечивает полиномиальную вычислимость функций и полиномиальную ограниченность некоторых множеств:

C4.

(i) Функции $F \in \mathcal{F}$, $H \in \mathcal{H}$ и G, а также отношение $\preceq_{\text{qua}}$ вычислимы за полиномиальное относительно длины входных данных задачи время.

(ii) Мощность $|\mathcal{F}|$ ограничена полиномом от длины входных данных.

(iii) $\mathcal{S}_0$ вычислимо за полиномиальное время относительно длины входных данных.

(iv) Существует полином $\pi_1(n, \log_2 |I|)$, такой что компоненты состояний множеств $\mathcal{S}_k$, $k = 1, \dots, n$ суть целые числа, не превышающие $e^{\pi_1(n, \log_2 |I|)}$. Кроме того, для всех $\ell \in \mathcal{L}_0$ мощность множества значений, которые может принимать такая координата $|\{s_\ell : (s_1, \dots, s_\ell, \dots, s_\beta) \in \mathcal{S}_k\}|$ ограничена полиномом $\pi_2(n, \log_2 |I|)$.

Пусть задача П разрешима алгоритмом ДП по рекуррентной формуле (1), выполнено условие (2) или (3) и существуют частичный порядок $\preceq_{\text{dom}}$, линейный предпорядок $\preceq_{\text{qua}}$ и вектор степеней D, удовлетворяющие условиям C1-C4. Тогда будем говорить, что П является DPB-задачей (от английского термина *DP-benevolent problem* [24]).

2.2 Вполне полиномиальная аппроксимационная схема

Пусть L – достаточно большая величина, выбранная для данной индивидуальной задачи I и погрешности ε (выбор величины L будет обсуждаться далее). Рассмотрим алгоритм ДП$_\Delta$, представляющий собой модификацию ДП с «прореживанием» множества состояний на каждой стадии. При описании алгоритма будем пользоваться разбиением множества $B(L, \Delta) = Z^\beta_+ \cap [0, \Delta^L]^\beta$ на параллелепипеды следующего вида: пусть параллелепипед $\mathcal{B}_{(i_1, \dots, i_\beta)}$ с индексом $(i_1, \dots, i_\beta) \in Z^\beta_+$ содержит все целочисленные точки $S = (s_1, \dots, s_\beta)$, удовлетворяющие условиям:

$$s_\ell \in \begin{cases} \{0\} & \text{если} \quad i_\ell = 0; \\ [\Delta^{i_\ell - 1}, \Delta^{i_\ell} - 1] & \text{если} \quad 0 < i_\ell < L; \\ [\Delta^{i_\ell - 1}, \Delta^{i_\ell}] & \text{если} \quad i_\ell = L \end{cases} \tag{4}$$

для ℓ таких, что $\ell \in \mathcal{L}_1$, и

$$s_\ell = i_\ell \tag{5}$$

при $\ell \in \mathcal{L}_0$.

Алгоритм ДП$_\Delta$

1. Инициализировать $\mathcal{T}_0 := \mathcal{S}_0$.
2. Для k от 1 до n выполнять:
3. Положить $\mathcal{T}_k := \emptyset$.
4. Для каждого $S \in \mathcal{T}_{k-1}$ и каждого $F \in \mathcal{F}$ выполнять:
5. Если $H_F(X_k, S) \le 0$ и в параллелепипеде $\mathcal{B}_{(i_1,\dots,i_\beta)}$, содержащем $F(X_k, S)$, не существует такого $S' = (s'_1, \dots, s'_\beta) \in \mathcal{T}_k$, что $F(X_k, S) \preceq_{\text{qua}} S'$, то добавить $F(X_k, S)$ в $\mathcal{T}_k$ и удалить из $\mathcal{T}_k$ все состояния $S'' \in \mathcal{B}_{(i_1,\dots,i_\beta)} \cap \mathcal{T}_k$, такие что $S'' \preceq_{\text{qua}} F(X_k, S)$.
6. Конец цикла по S и F.
7. Конец цикла по k.
8. Выбрать $\min\{G(S) : S \in \mathcal{T}_n\}$ в случае задачи минимизации, либо $\max\{G(S) : S \in \mathcal{T}_n\}$ в случае задачи максимизации.

Для простоты изложения в приведенном алгоритме не описывается процедура обратного хода для восстановления допустимой траектории по выбранному на шаге 8 элементу $S \in \mathcal{T}_n$. Отсутствует и дополнительная информация, сохраняемая при каждом переходе для этой процедуры. При необходимости алгоритм может быть дополнен этими элементами стандартным образом.

Как показано в [24], для получения вполне полиномиальной аппроксимационной схемы в ДП$_\Delta$ достаточно положить

$$\Delta = 1 + \frac{\varepsilon}{2gn}, \tag{6}$$

$$L = \left\lceil \frac{\pi_1(n, \log_2 |I|)}{\ln \Delta} \right\rceil. \tag{7}$$

При этом L полиномиально ограничена от длины входа и от $1/\varepsilon$ (см. лемму 2 в приложении). Основной результат [24] формулируется следующим образом.

Теорема 1 *Пусть Π – DPB-задача. Тогда семейство алгоритмов ДП$_\Delta$ с выбором параметров Δ и L согласно (6) и (7) представляет собой вполне полиномиальную аппроксимационную схему.*

3 Эволюционный алгоритм

Предлагаемый здесь эволюционный алгоритм EA$_\Delta$ аналогичен эволюционному алгоритму [8] с основным отличием в правиле добавления особей в популяцию. В качестве особей рассматриваются пары вида $a = (k, S)$, составленные из номера стадии k и элемента пространства состояний $S \in B(L, \Delta)$. Обозначим множество всевозможных особей популяции через A, $A \subseteq \{0, \dots, n\} \times B(L, \Delta)$. Потребуем, чтобы все особи популяции были различны. Множество всевозможных популяций в таком случае есть 2^A.

Опишем EA$_\Delta$ в предположении, что Π является DPB-задачей. Алгоритм начинает работу с исходной популяции $\mathcal{P}_0 = \{0\} \times \mathcal{S}_0$, для построения которой используется полиномиальная процедура, существующая согласно условию C4 (iii). В ходе работы EA$_\Delta$

порождается последовательность популяций $\mathcal{P}_t \subseteq \{t\} \times B(L, \Delta),\ t = 1, 2, \ldots, \tau$. Значение τ для критерия остановки будет выбрано позднее.

Для создания новых особей используются вероятностные операторы селекции $sel : 2^A \to A$ и мутации $mut : A \to A$. Из заданной популяции $\mathcal{P}$ оператор селекции $sel(\mathcal{P})$ выбирает одну особь случайным образом с равномерным распределением вероятностей. При действии оператора мутации $mut(a)$ к особи $a = (k, S)$ применяется функция перехода F, случайно выбранная из $\mathcal{F}$ с равномерным распределением вероятностей, и полагается $mut((k, S)) = (k+1, F(X_{k+1}, S))$. Новая особь $(k+1, F(X_{k+1}, S))$ добавляется в популяцию, если $H_F(X_{k+1}, S) \leq 0$ и если эта особь не уступает уже имеющимся в популяции $\mathcal{P}_t$ в смысле следующего отношения $\succeq_{\mathrm{EA}}$. Предпорядок $\succeq_{\mathrm{EA}}$ определен на основе линейного предпорядка $\succeq_{\mathrm{qua}}$ таким образом: $(k', S') \succeq_{\mathrm{EA}} (k, S)$ тогда и только тогда, когда $k' = k$, $S' \succeq_{\mathrm{qua}} S$, и существует параллелепипед $\mathcal{B}_{(i_1,\ldots,i_\beta)}$, такой что $S, S' \in \mathcal{B}_{(i_1,\ldots,i_\beta)}$. Схема алгоритма имеет следующий вид.

Алгоритм EA$_\Delta$
1. Инициализировать $\mathcal{P}_0$.
2. Для t от 1 до τ выполнять:
3. Положить $\mathcal{P}_t := \mathcal{P}_{t-1}$.
4. Построить $a := mut(sel(\mathcal{P}_t))$. Пусть $a = (k, S)$.
5. Если $H_F(S) \leq 0$ и не существует $a' \in \mathcal{P}_t$, такого что $a \preceq^{\mathrm{EA}} a'$,
6. то положить $\mathcal{P}_t := (\mathcal{P}_t \setminus \{a' \in \mathcal{P}_t \mid a' \preceq^{\mathrm{EA}} a\}) \cup \{a\}$.
7. Конец цикла по t.
8. Выбрать $\min\{G(S) : (n, S) \in \mathcal{P}_\tau\}$ в случае задачи минимизации,
либо $\max\{G(S) : (n, S) \in \mathcal{P}_\tau\}$ в случае задачи максимизации.

Из схемы алгоритма вытекает, что для любого $k \in \{1, \ldots, n\}$ особи вида (k, S) в любой популяции $\mathcal{P}_t,\ t = 0, \ldots, \tau$, представляют только состояния $S \in \mathcal{S}_k$, т.к. состояния из $\mathrm{Z}^\beta \backslash \mathcal{S}_k$ не могут быть построены с помощью операторов sel и mut за k итераций.

Восстановление допустимой траектории по выбранному S на шаге 8 EA$_\Delta$ не описывается, но при необходимости это может быть выполнено стандартным способом, также как и в алгоритме ДП$_\Delta$.

4 Вполне полиномиальная рандомизированная аппроксимационная схема

Без потери общности, в настоящем параграфе будем полагать, что Π – задача минимизации. Для произвольного $S \in \mathcal{S}_k$ обозначим через $\theta(k, S)$ случайную величину, равную номеру итерации EA$_\Delta$, на которой впервые получена особь (k, T), такая что T (D, Δ^k)-близко к S и $S \preceq_{\mathrm{qua}} T$. Заметим, что по схеме EA$_\Delta$, начиная с итерации $\theta(k, S)$ в популяции всегда можно найти некоторую особь T, удовлетворяющую данному условию.

Следующая лемма показывает, что для любого недоминируемого $S \in \mathcal{S}_k$ в среднем за полиномиально ограниченное от длины входа и от $1/\varepsilon$ число итераций будет получена особь (k, T), такая что T (D, Δ^k)-близко к S и $S \preceq_{\mathrm{qua}} T$. Здесь и далее математическое ожидание обозначается символом $E[\cdot]$.

Лемма 1 *Пусть* Π *является DPB-задачей. Тогда для любого* $k = 0, \ldots, n$ *и всякого недоминируемого в* $\mathcal{S}_k$ *состояния* S *выполняется неравенство* $E[\theta(k,S)] \leq nk(L\pi_2(n, \log_2 |I|))^{\beta}|\mathcal{F}|$.

Доказательство. Воспользуемся индукцией по k. При $k = 0$ утверждение леммы выполняется тривиально. Рассмотрим произвольное недоминируемое состояние S в $\mathcal{S}_k$. Пусть $k > 0$ и утверждение леммы верно для $\mathcal{P}_{k-1}$.

Относительно алгоритма ДП в [24] доказано (см. лемму 3 в приложении), что для S найдется недоминируемое в $\mathcal{S}_{k-1}$ состояние $S^{\#}$ и отображение $F^{\#} \in \mathcal{F}$, такие что $F^{\#}(X_k, S^{\#}) = S$, и $H_{F^{\#}}(X_k, S^{\#}) \leq 0$. Отметим, что предположение индукции дает оценку сверху для математического ожидания числа итераций $\theta(k-1, S^{\#})$ до получения особи $(k-1, T^{\#})$, такой что $T^{\#}$ (D, Δ^{k-1})-близко к $S^{\#}$ и $S^{\#} \preceq_{\mathrm{qua}} T^{\#}$. Назовем мутацию с применением $F^{\#}$ к особи $(k-1, T^{\#})$ *успешной мутацией*.

Согласно условию C2 (i) имеем:

$$H_{F^{\#}}(X_k, T^{\#}) \leq H_{F^{\#}}(X_k, S^{\#}) \leq 0,$$

а по предположению C1 (i) либо (а) $F^{\#}(X_k, T^{\#})$ будет (D, Δ^{k-1})-близко к S и $S \preceq_{\mathrm{qua}} F^{\#}(X_k, T^{\#})$, либо (б) $S \preceq_{\mathrm{dom}} F^{\#}(X_k, T^{\#})$.

В случае (а), после успешной мутации, популяция будет содержать элемент $(k, F^{\#}(X_k, T^{\#}))$, либо некоторый другой элемент (k, T'), такой что T' попадает в тот же параллелепипед $\mathcal{B}_{(i_1,\ldots,i_\beta)}$, что и $F^{\#}(X_k, T^{\#})$ и кроме того $T' \succeq_{\mathrm{qua}} F^{\#}(X_k, T^{\#})$. То есть, после указанной мутации популяция будет содержать особь (k, T), такую что T (D, Δ)-близко к $F^{\#}(X_k, T^{\#})$ и $F^{\#}(X_k, T^{\#}) \preceq_{\mathrm{qua}} T$. Далее, т.к. $F^{\#}(X_k, T^{\#})$ (D, Δ^{k-1})-близко к S, то из определения близости состояний вытекает, что T (D, Δ^k)-близко к S. Кроме того, $S \preceq_{\mathrm{qua}} F^{\#}(X_k, T^{\#}) \preceq_{\mathrm{qua}} T$, следовательно, $S \preceq_{\mathrm{qua}} T$. То есть, в случае (а) успешная мутация обеспечивает присутствие искомого представителя для S в популяции на стадии k.

В случае (б), после успешной мутации будет получен элемент (k, S), т.к. S – недоминируемое состояние, а $F^{\#}(X_k, T^{\#})) \succeq_{\mathrm{dom}} S$. В результате, популяция будет содержать элемент (k, S), либо некоторый другой элемент (k, T'), такой что T' попадает в тот же параллелепипед $\mathcal{B}_r$, что и S и кроме того $T' \succeq_{\mathrm{qua}} S$. То есть, после указанной мутации популяция будет содержать особь (k, T), такую что T (D, Δ)-близко к S и $S \preceq_{\mathrm{qua}} T$. Очевидно, при этом, T также (D, Δ^k)-близко к S. Таким образом, успешная мутация обеспечивает присутствие искомого представителя для S в популяции на стадии k.

Для завершения доказательства остается оценить математическое ожидание числа испытаний θ^* до осуществления успешной мутации при условии наличия особи $a^{\#} = (k-1, T^{\#})$ в текущей популяции $\mathcal{P}_t$. Заметим, что по схеме алгоритма вероятность успешной мутации $p^* = (n \cdot |\{(k-1, S') \in \mathcal{P}_t\}| \cdot |\mathcal{F}|)^{-1}$, при этом

$$|\mathcal{P}_t| = \sum_{k'=1}^{n} |\{(k', S') \in \mathcal{P}_t\}| \leq \sum_{k'=1}^{n} |\{(i_1, \ldots, i_\beta) : \mathcal{B}_{(i_1,\ldots,i_\beta)} \cap \mathcal{S}_{k'} \neq \emptyset\}|. \tag{8}$$

Рассмотрим слагаемое в правой части (8) при любом фиксированном k'. Для каждого $\ell \in \mathcal{L}_1$ индекс параллелепипеда i_ℓ может принимать не более L значений. Кроме того, ввиду условия C4 (iv), для каждого $\ell' \in \mathcal{L}_0$ координата $s_{\ell'}$ в состояниях из множества $\mathcal{S}_{k'}$ может принимать не более $\pi_2(n, \log_2 |I|)$ значений.

Таким образом, правая часть неравенства (8) не превышает $nL^{|\mathcal{L}_1|}\pi_2(n, \log_2 |I|)^{|\mathcal{L}_0|} \le n(L\pi_2(n, \log_2 |I|))^\beta$, а значит, $E[\theta^*] = 1/p^* \le n(L\pi_2(n, \log_2 |I|))^\beta |\mathcal{F}|$. Утверждение леммы для стадии k вытекает из того, что $E[\theta(k, S)] = E[\theta(k-1, T^\#)] + E[\theta^*]$. Лемма 1 доказана.

Полученная в лемме 1 оценка $E[\theta(k, S)]$ служит основой для выбора критерия остановки алгоритма EA_Δ. Положим

$$\tau = 4n^2(L\pi_2(n, \log_2 |I|))^\beta |\mathcal{F}|. \tag{9}$$

Теорема 2 *Если задача* Π *является DPB-задачей, то семейство алгоритмов* EA_Δ *с выбором параметров* Δ *и* L *согласно (6) и (7) и критерием остановки (9) образует вполне полиномиальную рандомизированную апроксимационную схему.*

Доказательство. Рассуждения аналогичны доказательству основного результата в [24].

Согласно сделанным предположениям, $OPT(I) = \min\{G(S) : S \in \mathcal{S}_n\}$. Поэтому, ввиду условия C3 (ii), существует недоминируемое состояние $S^* \in \mathcal{S}_n$, такое что $OPT(I) = G(S^*)$. По лемме 1, в среднем не более чем за $n^2(L\pi_2(n, \log_2 |I|))^\beta |\mathcal{F}|$ итераций в EA_Δ вычисляется популяция, в которой найдется особь (n, T^*), такая что T^* (D, Δ^n)-близко к S^* и $S^* \preceq_{qua} T^*$. Тогда по условию C3 (i):

$$G(T^*) \le \Delta^{gn} G(S^*) = \left(1 + \frac{\varepsilon}{2gn}\right)^{gn} OPT(I) \le (1+\varepsilon) OPT(I).$$

(Последнее неравенство вытекает из того что $gn \ge 1$, поэтому $(1 + \frac{\varepsilon}{2gn})^{gn}$, как функция от ε, на отрезке $\varepsilon \in [0, 2]$ выпукла и требуемое неравенство выполнено на концах отрезка.) В таком случае алгоритм обеспечивает искомую точность аппроксимации по целевой функции и с помощью процедуры обратного хода по T^* может быть эффективно получено $(1+\varepsilon)$-приближенное решение.

При выполнении EA_Δ с критерием остановки (9), согласно неравенству Маркова, вероятность *не найти* $(1+\varepsilon)$-приближенное решение, составляет не более $1/4$, как и требуется в определении вполне полиномиальной рандомизированной аппроксимационной схемы.

Наконец, по условию C4 трудоемкость каждой итерации EA_Δ полиномиально ограничена от длины входа и $1/\varepsilon$, поэтому данное семейство алгоритмов является вполне полиномиальной рандомизированной аппроксимационной схемой. Теорема 2 доказана.

Для задач максимизации схема алгоритма EA_Δ и доказательство того, что он дает вполне полиномиальную рандомизированную аппроксимационную схему, аналогичны.

5 Заключение

В работе представлен новый подход к использованию специфики задачи в эволюционных алгоритмах. Основной результат показывает, что к эволюционным алгоритмам для

DPB-задач, наряду со сходимостью к оптимуму почти наверное (см., например, [21]), естественно предъявить новые требования в терминах вероятности эффективного получения приближенного решения.

Предложенный эволюционный алгоритм не обладает более низкими оценками трудоемкости в среднем по сравнению с оценками трудоемкости для известных детерминированных алгоритмов. Тем не менее, на практике в операторах селекции и мутации может быть эвристически учтена специфика задачи, позволяющая ускорить получение решений требуемого качества. Для оценки перспективности данного подхода необходимы дальнейшие экспериментальные исследования.

Приложение

Для замкнутости изложения прилагаются доказательства вспомогательных результатов из [24], используемых в работе.

Лемма 2 *[24] Пусть $\Delta = 1 + \varepsilon/(2gn)$, тогда величина $L = \left\lceil \frac{\pi_1(n,\log_2|I|)}{\ln \Delta} \right\rceil$ полиномиально ограничена сверху относительно длины входа и от $1/\varepsilon$, и всякое состояние из множества $\mathcal{S}_k$, $k = 1, \dots, n$, принадлежит одному из параллелепипедов $\mathcal{B}_{(i_1,\dots,i_\beta)}$, заданных условиями (4) и (5).*

Доказательство. Согласно условию C4 (iv), существует полином $\pi_1(n, \log_2 |I|)$, такой что компоненты состояний множеств $\mathcal{S}_k$, $k = 1, \dots, n$, суть целые числа, не превышающие $e^{\pi_1(n,\log_2|I|)} \le \Delta^L$. Таким образом, любое состояние $S \in \mathcal{S}_k$ принадлежит одному из параллелепипедов $\mathcal{B}_{(i_1,\dots,i_\beta)}$. Заметим, что $\ln \Delta \ge (\Delta-1)/\Delta$ (для этого достаточно сравнить производные функций $\ln \Delta$ и $(\Delta-1)/\Delta$). Следовательно, $1/\ln \Delta \le 1+1/(\Delta-1) = 1+2gn/\varepsilon$, то есть, $L = \left\lceil \frac{\pi_1(n,\log_2|I|)}{\ln \Delta} \right\rceil$ полиномиально ограничено от длины входа и от $1/\varepsilon$. Лемма 2 доказана.

Лемма 3 *[24] Пусть S – недоминируемое состояние в $\mathcal{S}_k$ для некоторого $1 \le k \le n$, тогда для S найдется недоминируемое в $\mathcal{S}_{k-1}$ состояние $S^\#$ и отображение $F^\# \in \mathcal{F}$, такие что $F^\#(X_k, S^\#) = S$, и $H_{F^\#}(X_k, S^\#) \le 0$.*

Доказательство. Предположим, состояние S попадает в множество $\mathcal{S}_k$, когда $F^\# \in \mathcal{F}$ применяется к X_k и некоторому состоянию из $\mathcal{S}_{k-1}$. Пусть $H^\# \in \mathcal{H}$ – отображение, соответствующее $F^\#$. Рассмотрим непустое подмножество $\mathcal{S}^\diamond \subseteq \mathcal{S}_{k-1}$,

$$\mathcal{S}^\diamond = \{S^\diamond : S^\diamond \in \mathcal{S}_{k-1},\ H^\#(X_k, S^\diamond) \le 0,\ F^\#(X_k, S^\diamond) = S\}. \tag{10}$$

Пусть $S^\#$ – недоминируемое состояние в $\mathcal{S}^\diamond$ (такое состояние всегда существует, т.к. $\preceq_{\mathrm{dom}}$ – частичный порядок и $\mathcal{S}^\diamond$ конечно). Допустим, $S^\#$ доминируется в $\mathcal{S}_{k-1}$ некоторым состоянием $S' \in \mathcal{S}_{k-1} \backslash \mathcal{S}^\diamond$. Тогда по условиям C2 (ii) и (10),

$$H^\#(X_k, S') \le H^\#(X_k, S^\#) \le 0.$$

Следовательно, $F^\#(X_k, S')$ принадлежит $\mathcal{S}_k$ и доминирует S согласно условию C1 (ii). Кроме того, $F^\#(X_k, S') \ne S$, т.к. $S' \notin \mathcal{S}^\diamond$. Но это противоречит предположению о том, что S – недоминируемое состояние в $\mathcal{S}_k$. Следовательно, такого S' не существует, и $S^\#$ – недоминируемое состояние в $\mathcal{S}_{k-1}$. Лемма 3 доказана.

Список литературы

[1] **Гэри М., Джонсон Д.** Вычислительные машины и труднорешаемые задачи. М.: Мир, 1982.

[2] **Еремеев А.В.** О связи динамического программирования и многокритериальных эволюционных алгоритмов. Препринт. ОмГУ, Омск: 2008.

[3] **Ковалев М.Я., Шафранский Я. М.** Построение ε-приближенных алгоритмов оптимизации функции на последовательно конструируемых множествах // *Журн. выч. матем. и матем. физ.* 1986. Т. 26, N 7, С. 1006–1018.

[4] **Моисеев Н.Н., Иванилов Ю.П., Столярова Е.М.** Методы оптимизации. М.: Наука, 1978.

[5] **Норенков И.П.** Эвристики и их комбинации в генетических методах дискретной оптимизации // *Информационные технологии.* 1999. N 1, С. 2–7.

[6] **Beyer H.-G., Schwefel H.-P., Wegener I.** How to analyse evolutionary algorithms // *Theor. Comp. Sci.* 2002. Vol. 287, Issue 1, P. 101–130.

[7] **Chauhan S.S., Eremeev A.V., Romanova A.A., Servakh V.V., Woeginger G.J.** Approximation of the supply scheduling problem // *Oper. Res. Lett.* 2005. Vol. 33, N 3. P. 249–254.

[8] **Doerr B., Eremeev A., Horoba C., Neumann F., Theile M.** Evolutionary algorithms and dynamic programming // Proc. of Genetic and Evolutionary Computation Conf. (GECCO), New York: ACM Press, 2009. P. 771–777.

[9] **Friedrich T., Hebbinghaus N., Neumann F., He J., Witt C.** Approximating covering problems by randomized search heuristics using multi-objective models // Proc. of Genetic and Evolutionary Computation Conf. (GECCO), New York: ACM Press, 2007. P. 797–804.

[10] **Giel O., Wegener I.** Evolutionary algorithms and the maximum matching problem // Proc. Symp. on Theoretical Aspects of Computer Science (STACS). Springer, 2003. P. 415–426.

[11] **Held, M., Karp, R. M.** A dynamic programming approach to sequencing problems // *J. of Soc. for Indust. and Appl. Math.* 1962. V. 10. P. 196–210.

[12] **Ibarra O.H., Kim C.E.** Fast approximation algorithms for the knapsack and sum of subset problems // *J. of the ACM.* 1975. V. 22, N 4. P. 463–468.

[13] **Koza J.R.** Genetic Programming II: Automatic Discovery of Reusable Programs (Complex Adaptive Systems). Cambridge, MA: MIT Press, 1994.

[14] **Michalewicz Z.** Genetic Algorithms + Data Structures = Evolution Programs. Springer-Verlag, 1996.

[15] **Neumann F., Reichel J., and Skutella M.** Computing minimum cuts by randomized search heuristics // Proc. of Genetic and Evolutionary Computation Conf. (GECCO), New York: ACM Press, 2008. P. 779–786.

[16] **Neumann F., Wegener I.** Randomized local search, evolutionary algorithms, and the minimum spanning tree problem // *Theor. Comp. Sci.* 2007. V. 378, N 1, P. 32–40.

[17] **Rechenberg I.** Evolutionsstrategie: Optimerung Technischer Systeme nach Prinzipen der Biologischen Evolution. Stuttgart: Formann-Holzboog Verlag, 1973.

[18] **Reeves C.R.** Genetic algorithms for the operations researcher // *INFORMS Journ. on Comput.* 1997. V. 9, N 3, P. 231–250.

[19] **Reeves C.R., Rowe J.E.** Genetic Algorithms: Principles and Perspectives. Norwell, MA: Kluwer, 2002.

[20] **Reichel J., Skutella M.** Evolutionary algorithms and matroid optimization problems // Proc. Genetic and Evolutionary Computation Conf. (GECCO). New York: ACM Press, 2007. P. 947–954.

[21] **Rudolph, G.** Finite Markov chain results in evolutionary computation: A tour d'horizon // *Fundamenta Informaticae.* 1998. V. 35, N 1-4, P. 67–89.

[22] **Scharnow J., Tinnefeld K., Wegener I.** The analysis of evolutionary algorithms on sorting and shortest paths problems // *Journ. Math. Mod. and Alg.* 2004. V. 3, P. 349–366.

[23] **Theile M.** Exact solutions to the travelling salesperson problem by a population-based evolutionary algorithm // Proc. European Conf. on Evolutionary Computation in Combinatorial Optimisation (EvoCOP). LNCS V. 5482. Springer, 2009. P. 145–155.

[24] **Woeginger G.J.** When does a dynamic programming formulation guarantee the existence of a fully polynomial time approximation scheme (FPTAS)? // *INFORMS Journal on Computing.* 2000. V. 12, Issue 1. P. 57–74.