

МЕТАЭВРИСТИЧЕСКИЙ МЕТОД ДЕФОРМАЦИЙ ДЛЯ РЕШЕНИЯ ЗАДАЧ КОМБИНАТОРНОЙ ОПТИМИЗАЦИИ

Леонид Гуляницкий

Аннотация: *Исследуются проблемы разработки приближенных алгоритмов комбинаторной оптимизации. Приводится формализация понятия комбинаторных объектов, позволяющая строго формально определить понятие задачи комбинаторной оптимизации. Предлагается метаэвристический алгоритм комбинаторной оптимизации, названный H -методом, метаэвристика которого использует аналогии с известным в недифференцируемой непрерывной оптимизации методом Нелдера-Мида. В качестве встроенной процедуры использован G -алгоритм – один из алгоритмов стохастического локального поиска. Обсуждаются вопросы практической реализации алгоритмов предложенного H -метода. Приводятся результаты вычислительного эксперимента по решению квадратичных задач о назначениях из известной библиотеки.*

Ключевые слова: *комбинаторная оптимизация, метаэвристики, популяционные методы, стохастический локальный поиск, квадратичная задача о назначении.*

Введение

Среди прикладных проблем моделирования видное место занимают задачи выбора оптимальных решений в дискретных (в частности, конечных) пространствах, чем обуславливаются потребности в исследовании и внедрении моделей и методов комбинаторной оптимизации (КО). Для решения возникающих сложных задач КО, а также для решения задач повышенной размерности необходима разработка новых общих вычислительных схем (в том числе с распараллеливанием вычислений) и исследование их эффективности. Поскольку большинство практически важных задач КО относится к числу NP -трудных, основное внимание уделяется разработке и исследованию приближенных алгоритмов. Прогресс в повышении производительности вычислительной техники привел в последние годы к появлению так называемых метаэвристических (гибридных) методов оптимизации, применение которых направлено на повышение точности решения задач КО [1,2].

Метаэвристические методы основаны на использовании двух техник: общая схема строится на базовом методе (метаэвристике), в которую включается та или иная встроенная процедура. Большинство разработанных метаэвристических методов относятся к классу популяционных алгоритмов [2], т.е. алгоритмов, в которых, в отличие от траекторных, на каждой итерации обрабатывается не один, а сразу несколько вариантов решения.

В работе [3] предложен алгоритм комбинаторной оптимизации, предназначенный для использования как на традиционных ЭВМ, так и многопроцессорных вычислительных комплексах (МВК), который назван методом деформированных многогранников (МДМ). В его вычислительной схеме осуществляется глобальное сканирование пространства решений решаемой задачи, чем повышается вероятность нахождения более точных вариантов решения.

Ниже излагается метаэвристический метод КО, названный H -методом, который в качестве метаэвристики использует схему МДМ, а встроенная процедура основана на одной из модификаций алгоритмов стохастического локального поиска.

Общая схема H -метода

В настоящее время общепринятым является определение задачи КО, предложенное Пападимитриу и Стайглицем [1,2,4]: необходимо найти $x^* \in X$ такое, что

$$x_* = \arg \min_{x \in D \subseteq X} f(x), \quad (1)$$

где X – конечное (или, возможно, счетное бесконечное) пространство решений задачи, D – его подпространство, определяемое ограничениями задачи, $f: X \rightarrow R^1$ – заданная целевая функция задачи.

Такое определение однозначно относит к задачам КО все проблемы оптимизации на конечных множествах, однако в случае бесконечных пространств не позволяет четко классифицировать задачи оптимизации по структуре элементов пространства решений. Берж [5] предложил формализовать понятие комбинаторной конфигурации следующим образом: пусть имеем m, n – натуральные, а также два множества $U = \{1, \dots, m\}$, $V = \{v_1, \dots, v_n\}$, причем на V задан некий строгий порядок $v_1 < \dots < v_n$ (т.е. V – цепь).

Определение 1. Комбинаторной конфигурацией называется отображение $\varphi: U \rightarrow V$, которое удовлетворяет некоторому комплексу ограничений Λ .

Из определения следует, что при фиксированных m и n число комбинаторных конфигураций конечно. Выбор ограничений в Λ позволяет описывать различные комбинаторные конфигурации.

Предлагается следующее обобщение схемы Бержа. Пусть заданы $Y = \{1, \dots, m\}$, Z – дискретное, в частности, конечное пространство (назовем его образующим), φ – гомоморфизм, $\varphi: Y \rightarrow Z$, удовлетворяющий некоторой системе ограничений Ω . Напомним, что под дискретным пространством понимается множество, состоящее из изолированных точек.

Определение 2. Под комбинаторным объектом κ будем понимать триаду $\kappa = (\varphi, \tilde{X}, \Omega)$, где \tilde{X} – базовое пространство.

Определение 3. Назовем комбинаторными объектами 1-го порядка такие комбинаторные объекты, у которых базовое пространство совпадает с образующим:

$$\kappa = (\varphi, X_{(1)}, \Omega),$$

где $X_{(1)} \equiv Z$.

Нетрудно убедиться, что если Z – это конечная цепь, то такие комбинаторные объекты совпадают с комбинаторными конфигурациями в смысле Бержа [5].

Определение 4. Комбинаторными объектами k -го порядка ($k > 1$) назовем комбинаторные объекты

$$\kappa = (\varphi, X_{(k)}, \Omega),$$

где $X_{(k)} \subseteq X_{(k-1)} \cup X^k$.

Возвращаясь к оптимизационной задаче (1), дадим следующее

Определение 5. Задача (1) называется задачей КО, если пространство ее решений X – это пространство, элементами которого являются комбинаторные объекты.

В ряде развитых алгоритмов КО для избежания концентрации поиска в ограниченной подобласти пространства решений задачи X и повышения точности получаемых решений используются процедуры возмущения (как в повторяющемся локальном поиске [6]) или скрещивания и мутации, как в генетических алгоритмах (ГА) или меметических алгоритмах (МА) [1]. Заметим, что подобные процедуры порождают

подмножества вариантов решения, которые не согласованы с топологией пространства X . В то же время, примером подобного согласования является упомянутый МДМ. Осуществляемое в нем использование специальных отрезков дает возможность синтезировать поиск в окрестностях и глобальное сканирование пространства решений X , причем процедура сканирования, в отличие от общих операторов возмущения или рекомбинации в большинстве других метаэвристических методов, определенная конкретно.

На основе синтеза идей МДМ и G -алгоритмов [7] (а также подхода на основе популяций, который был использован, например, при разработке ГА и фронтальных алгоритмов [8]), предлагается схема гибридного алгоритма, названного H -методом [9].

В алгоритмах МДМ существенную роль играет понятие d -отрезка $/x, y/$, соединяющего произвольные две точки $x, y \in X$, $X=(X, d)$ – метрическое пространство с метрикой d [8].

Определение 6. Назовем d -отрезком $/x, y/$ упорядоченную совокупность точек $x_i \in X$, $i=1, \dots, k$, которые удовлетворяют условию: $d(x, x_i) + d(x_i, y) = d(x, y)$ для всех $i=1, \dots, k$, причем $x_1 = x$, $x_k = y$, а $d(x, x_i) < d(x, x_{i+1})$, $i=1, \dots, k-1$; при этом не существует точки $z \in X$ такой, что $d(x_i, z) + d(z, x_{i+1}) = d(x, x_{i+1})$, $z \neq x_i$, $z \neq x_{i+1}$, $i=1, \dots, k-1$.

Определение 7. d -интервалом $\langle x, y \rangle$ назовем упорядоченную совокупность $/x, y/ \setminus \{x, y\}$.

Дальше будем рассматривать такие пространства вариантов решения X , для которых $d(x, y) = b \cdot h$ (b – натуральное, $h > 0$), причем если $d(x, y) > h$, то интервал $\langle x, y \rangle \neq \emptyset$. Без ущерба для общности положим, что $D \equiv X$, т.е. X будет играть роль пространства допустимых решений.

Вычислительная схема H -метода в терминах эволюционных вычислений представлена на рис., где **G_Search** – процедура G -алгоритма [7], осуществляющего поиск локального оптимума (не исключается возможность использования и другого метода поиска локально оптимального типа).

```

procedure  $H(x)$ ;
begin
   $h := 0$ ;  $P^0 := \emptyset$ ;
  for  $i:=1$  to  $m$  do
     $x :=$  некоторый начальный вариант решения;
     $x :=$  G_Search ( $x$ );
     $P^0 := P^0 \cup x$ ;
  end for;                                     {сформирована начальная популяция  $P^0$ }
  repeat
     $P := P^1$ ;
    for  $i:= 1$  to  $k$  do
      ОтборДляВариации ( $x, y \in P$ );
      ПостроениеПолуинтервала  $\langle x, x^\infty / : y \in \langle x, x^\infty / \ \& \ f(x) > f(y)$ ;
       $z := \arg \min \{f(u) : u \in \langle x, x^\infty / \setminus L_s(y), y \in \langle x, x^\infty / \}$ ;
       $z :=$  G_Search ( $z$ );
       $P := P \cup z$ ;
    end for;                                     {сформирована временная популяция из  $m+k$  точек}

```

```

for  $i:=1$  to  $l$  do
  ОтборДляМутации ( $x \in P$ );
   $z:=$ Мутация ( $x$ , предыстория);
   $z:=G\_Search$  ( $z$ );
   $P:=P \cup z$ ;
end for;                                     {сформирована временная популяция из  $m+k+l$  точек}
 $P^{h+1} :=$  ОтборПопуляции ( $P$ );
 $h :=h+1$ ;
until не выполняется условие завершения;
 $x := \arg \min \{f(u): u \in P\}$ ;
return  $x$ ;
end

```

Рис. Схема H -метода

Основные параметры H -метода:

m – число особей в популяции (в терминах эволюционных вычислений);

k – количество пар особей, выбираемых для исследования (проведения полуинтервалов от текущей точки до максимально удаленной и поиска минимума на этих прямых);

l – количество особей, которые подлежат мутации;

s – радиус метрической окрестности $L_s(y)$, точки которой, находящиеся на построенном полуинтервале, исключаются из рассмотрения (в большинстве случаев s – натуральное);

x^∞ – точка, максимально удаленная в пространстве вариантов решений от исходной точки x (диаметрально противоположная в пространстве X точка).

Кроме численности популяции m , другие параметры могут изменяться от итерации к итерации, то есть определяться динамически с учетом хода процесса поиска экстремумов.

Как и в алгоритме повторяющегося локального поиска или МА, предлагаемый метод оперирует с локальными экстремумами. Их множество P играет роль, аналогичную популяции в ГА или МА – поэтому использованные в нем три процедуры отбора могут реализовываться по аналогии с эволюционными алгоритмами. Принципиальное отличие алгоритма H -метода – глобальный характер поиска в пространстве решений X путем нахождения субоптимального решения исходной задачи на основе решения подзадачи вида:

$$z = \arg \min_{u \in \langle x, x^\infty \rangle \setminus L_s(y)} f(u), \quad (2)$$

где $y \in \langle x, x^\infty \rangle$.

Вопросы реализации ключевых аспектов вычислительной схемы

При практическом применении алгоритмов H -метода было отмечено, что поскольку выбранные точки x и y являются локальными экстремумами, то часто может возникнуть ситуация, когда $z=y$. Во избежание этого и вводится параметр $s > 0$ – значение радиуса метрической окрестности $L_s(y)$, точки которой исключаются из рассмотрения в подзадачах (2). Понятно, что для большинства комбинаторных пространств целесообразно выбирать $s \geq 1$, исключая тем самым возможность определения точки y в качестве субоптимального решения в указанной подзадаче.

Как и в алгоритмах локального поиска, при решении подзадач (2) можно избежать повторного вычисления значений целевой функции: стоит вычислять лишь разность $\Delta = f(x) - f(y)$, учитывая тот факт, что отрезок составляют лишь соседние точки – во многих случаях вычисления разности имеет существенно более низкую трудоемкость.

В отличие от операторов скрещивания или других неструктурированных операторов рекомбинации предлагаемый подход не позволяет "стягиваться" точкам текущей популяции P , поэтому отпадает необходимость в диверсификации результатов, которая обычно производится в ГА или МА.

После этапов вариации и мутации образуется временное множество P , количество точек в которой ограничено величиной $m+k+l$. Задачей процедуры *ОтсевПопуляции* является уменьшение числа точек в P снова к величине m .

Использование предложенной схемы позволяет порождать семейство алгоритмов КО. Заметим, что при $m=1, k=0, l > 0$ получаем схему повторяющегося локального поиска [6], а при $l = 0$ – алгоритм без использования возмущений.

Отметим также, что механизм порождения субоптимального решения на полуинтервале позволяет провести отдаленную аналогию с алгоритмом рассеянного поиска [10]: его можно представлять как действие специфического оператора рекомбинации, который формирует потомка от трех родителей – точек $x, y, x^\infty \in X$.

Другой путь порождения алгоритмов – за счет альтернативных способов конкретизации (с учетом опыта применения и специфики решаемой задачи) следующих аспектов общей схемы метода, приведенной на рис.

1) Правило отбора очередной пары точек $x, y \in P$ для вариации:

- выбор таких точек, которые отвечают наилучшему и наихудшему значению функции пригодности среди точек P ;
- случайный выбор точек с вероятностью, пропорциональной пригодности или другим показателям;
- выбор "наихудшей точки" x и случайный выбор точки y , причем при этом следует учитывать и удаленность точек x и y – как в пространстве решений задачи, так и по значениям целевой функции.

Во всех случаях целесообразно, как показал опыт решения задач КО, в качестве начальной точки x выбирать ту точку из отобранной пары, у которой "худшее значение" целевой функции. Иными словами, для задач минимизации более эффективно поиск осуществляется в случае, когда выполняется неравенство $f(x) > f(y)$.

2) *Отбор для мутации*: в самом распространенном случае используется случайный отбор l точек. Сама мутация обычно заключается в возмущении нескольких компонентов заданного варианта решения, выбранных случайным образом.

3) Стратегия формирования новой популяции (*Отбор Популяции*), которая должна в итоге состоять из m точек, из образованной временной популяции объемом $m+k+l$ точек:

- отбор m самых пригодных точек;
- включение потомков на основе критерия Метрополиса;
- замена всех m родителей лучшими из $k+l$ потомков, $m \leq k+l$.

При определении функции пригодности для конкретной точки из P кроме значения целевой функции уместно дополнительно учитывать еще и такие показатели:

- число процедур отбора, в которых данная точка была выбрана для вариации;
- число потомков, которые уже отбирались в обновленные популяции;
- "время жизни", т.е. число осуществленных итераций, в течение которых точка не выбывала из множества P .

4) Критерием завершения вычислительного процесса может быть:

- превышение заданного числа H_{max} обновлений популяции P (алгоритм завершает работу, если $h > H_{max}$);
- стабилизация рекордного или среднего значения целевой функции в точках множества P :

$$\bar{f} = \frac{\sum_{x \in P} f(x)}{m};$$

- стремление к нулю разброса значений целевой функции в точках множества P : $\sigma = \max\{f(x): x \in P\} - \min\{f(x): x \in P\}$;
- достижение ситуации, когда все пары точек из очередной популяции уже принимали участие в формировании полуинтервалов – если используется однозначный способ построения метрических полуинтервалов $\langle x, x^\infty \rangle$.

5) Способ построения полуинтервалов $\langle x, x^\infty \rangle$. Рассматриваемые конечные комбинаторные метрические пространства X имеют две такие характерные особенности.

Во-первых, для произвольного $x \in P$ "максимально удаленной" является точка, удовлетворяющая условию: $x^\infty = \max \{d(x, y): y \in X\}$, где $d(x, y)$ – заданная на пространстве X метрика. Отметим, что часто это значение равно диаметру пространства X .

Во-вторых, во многих комбинаторных метрических пространствах между двумя несоседними точками $x, y \in P$ бывает возможным построение не одного, а сразу нескольких d -отрезков. Поэтому при реализации H -алгоритмов следует различать случай, когда из множества возможных d -отрезков $\langle x, y \rangle$, $d(x, y) > h$, всегда по конкретному правилу построения выбирается только один отрезок, или когда могут рассматриваться/строиться все возможные отрезки. В зависимости от этого может модифицироваться и правило "*Отсев Для Вариации*": если рассматривается один возможный полуинтервал $\langle x, x^\infty \rangle$, причем $u \in \langle x, x^\infty \rangle$, то вершины x и u после их выбора следует отметить как "отработанные", поскольку их повторный выбор становится нецелесообразным. Если же имеется возможность строить все или несколько возможных полуинтервалов $\langle x, x^\infty \rangle$, то повторный отбор некоторой пары x, u становится возможным, если при этом будет обеспечено построение именно нового полуинтервала.

Отметим, что при решении задач КО большой размерности процедура оптимизации вдоль полуинтервала может оказаться достаточно трудоемкой. В таких случаях можно ввести дополнительный параметр алгоритма, который бы определял часть полуинтервала $\langle x, x^\infty \rangle$, подлежащую просмотру в задачах (2) – то

есть определял бы максимально допустимую удаленность точек x и x^∞ в смысле метрического расстояния.

Следует особо отметить, что аналогичный параметр обязательно следует вводить в том случае, когда пространство X является бесконечным, поскольку тогда точка x^∞ действительно устремляется в бесконечность, а полуинтервал превращается в луч, а задача (2) будет иметь бесконечное число вариантов решения.

Вычислительный эксперимент

Поскольку теоретические исследования алгоритмов КО крайне редко позволяют получать результаты, пригодные при решении практических задач, принято анализировать показатели эффективности путем проведения вычислительных экспериментов. С этой целью часто используют "классические" модели КО – прежде всего, задачу коммивояжера и квадратичную задачу о назначениях [1]. В [11] приведены результаты вычислительного эксперимента по сравнению метаэвристического метода G-ГА, основанного на синтезе (гибридизации) схемы G-алгоритмов и ГА, с разными модификациями ГА (в том числе, близкими к МА [1,12]) при решении известных задач коммивояжера. Результаты этого эксперимента продемонстрировали преимущество метода G-ГА, что послужило основанием для использования его в нашем эксперименте по решению серии более трудоемких квадратичных задач о назначениях, заимствованных из известной Интернет-библиотеки [13].

Для решения названных задач был отобран специализированный комбинированный алгоритм, разработанный специально для решения задач на перестановках на основе схем ГА и локальной оптимизации (обозначение КГА), а также лидер упомянутого эксперимента – алгоритм G-ГА [11]. В табл. приведены результаты решения десяти квадратичных задач о назначениях из [13]. Здесь f_* – известное значение целевой функции в точке глобального минимума, f – найденное соответствующим алгоритмом значение целевой функции, δ – относительная погрешность алгоритма (%), t – время счета на ПЭВМ класса Pentium-II (с). В реализованном алгоритме H -метода выбиралось $m=10$, а мутации не производились ($I = 0$). Для квадратичных задач о назначениях меньшей размерности (19–20) лучшие результаты продемонстрировали ГА, но уже для $n > 20$ проявилось преимущество H -метода, который чаще находил точное решения. Из алгоритмов на базе генетических точнее был G-ГА, причем получаемая им точность была высшей по отношению к результатам КГА более чем в два раза.

Использование специально разработанной процедуры автонастройки параметров позволило в дальнейшем H -методу достичь оптимальных значений практически во всех задачах (в задаче LIPA40A – за 150 с, в задачах CHR20A и CHR20C – за 7 и 5 с соответственно).

Таблица 1. Результаты решения КЗН

Название задачи	n	f_*	КГА			G-ГА			H-метод		
			f	δ	t	f	δ	t	f	δ	t
ELS19	19	1721254	1721254	0	4	1721254	0	6	1721254	0	18
CHR20A	20	2192	2192	0	2	2192	0	6	2396	9	5
CHR20B	20	2298	2298	0	3	2298	0	5	2498	9	6
CHR20C	20	14142	14142	0	5	14142	0	7	16872	19	17

NUG21	21	2438	2447	0.4	5	2438	0	8	2438	0	6
NUG22	22	3596	3614	0.5	8	3598	0.1	7	3596	0	6
LIPA40A	40	31538	31779	0.8	80	31629	0.3	174	31649	0.4	103
LIPA40B	40	476581	485132	1.8	152	480056	0.7	132	476581	0	28
LIPA50A	50	62093	62891	1.3	612	62467	0.6	668	62589	0.8	325
LIPA50B	50	1210244	1246510	3.0	694	1227348	1.4	649	1210244	0	185

Заклучение

Предложенный H -метод может быть использован для решения широкого круга задач комбинаторной оптимизации, поскольку он изложен при весьма общих предположениях о решаемой задаче КО. Его вычислительная схема эффективно распараллеливается, что подтверждено итогами вычислительного эксперимента по решению задач КО на кластерном МВК СКИТ, разработанном в Институте кибернетики им. В.М. Глушкова НАН Украины [14].

Важной целью дальнейших теоретических исследований может стать получение условий сходимости и трудоемкости алгоритмов метода, предназначенных для решения конкретных классов задач КО.

Список литературы

1. Hoos H.H., Stützle T. Stochastic Local Search: Foundations and Applications. – San Francisco: Morgan Kaufmann Publ., 2005. – 658 p.
2. Blum C., Roli A. Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison // ACM Computing Surveys. – 2003. – 35, No. 3. – P. 268–308.
3. Гуляницький Л.Ф. Метод деформаций в дискретной оптимизации // Исследование операций и АСУ. – 1989. – Вып. 34. – С. 30–33.
4. Пападимитриу Х., Стайглиц К. Комбинаторная оптимизация. Алгоритмы и сложность. – М.: Мир, 1985. – 512 с.
5. Berge C. Principes de combinatoire. – Paris: Dunod, 1968. – 146 p.
6. Lourenço H. R., Martin O., Stützle T. Iterated local search // Handbook of Metaheuristics: International Series in Operations Research & Management Science, vol. 57 (Eds. F. Glover and G. Kochenberger). – Norwell: Kluwer Academic Publishers, MA, 2002. – P. 321–353.
7. Гуляницький Л.Ф. Решение задач комбинаторной оптимизации алгоритмами ускоренного вероятностного моделирования // Компьютерная математика. – Киев: Ин-т кибернетики им. В.М. Глушкова НАН Украины, 2004. – №1. – С. 64–72.
8. Сергиенко И.В., Гуляницький Л.Ф. Фронтальные алгоритмы для многопроцессорных ЭВМ // Кибернетика. – 1981. – №6. – С.1–4.
9. Гуляницький Л.Ф. Один гібридний алгоритм комбінаторної оптимізації // Abstract of Int. Ukrainian-Polish Workshop "Problems of Stochastic and Discrete Optimization" (May 10-15, 2005, Kaniv, Ukraine). – Kaniv, 2005. – P. 63–65.
10. Glover F. Scatter Search and Path Relinking // New Ideas in Optimization (Eds. D.Corne, M.Dorigo, F.Glover). – McGraw Hill, 1999. – P. 297-316.
11. Гуляницький Л.Ф. Разработка гибридных методов дискретной оптимизации на основе G -алгоритмов // Компьютерная математика. – Киев: Ин-т кибернетики им. В.М. Глушкова НАН Украины, 2005. №1. С. 143–151.
12. Haupt R.L., Haupt S.E. Practical Genetic Algorithms (2nd Ed.). – Hoboken: John Wiley&Sons, 2004. – 253 p.
13. QAPLIB //www.opt.math.tu-graz.ac.at/qaplib

-
14. Гуляницкий Л.Ф., Гложик Ю.С. Параллельный метод деформируемых многогранников для решения задач комбинаторной оптимизации // Искусственный интеллект. – 2005. – 4. – С.130–139.

Об авторе

Леонид Гуляницкий (Hulianytskyi) – д.т.н., Институт кибернетики им. В.М.Глушкова НАН Украины, пр-т Глушкова, 40, Киев, 03680, Украина. e-mail: lh_dar@hotmail.com