МУЛЬТИАГЕНТНЫЙ *H*-МЕТОД В КОМБИНАТОРНОЙ ОПТИМИЗАЦИИ

Леонид Гуляницкий, Денис Гобов

***Аннотация**: Предлагается метаэвристический алгоритм комбинаторной оптимизации, построенный на основе H-метода. В основе алгоритма лежит мультиагентный подход к исследованию пространства допустимых решений. Данный подход в отличие от стандартного H-метода, в котором между двумя точками в пространства решений строится только один отрезок специального вида, предполагает построение агентами нескольких отрезков. При работе агенты используют специальную модель решаемой задачи, что позволяет учитывать наряду с содержательной информацией о задаче и опыт, накопленный на предыдущих шагах алгоритма. Эффективность предложенного подхода проиллюстрирована на основе результатов вычислительного эксперимента по решению ряда задач коммивояжера и квадратичных задач о назначениях.*

***Ключевые слова**: комбинаторная оптимизация, метаэвристики, H-метод, агенты, задача коммивояжера, квадратичная задача о назначениях.*

***ACM Classification Keywords**: G.1.6 [Numerical Analysis] Optimization – Stochastic programming, G.2.1 [Discrete Mathematics] Combinatorics – Combinatorial algorithms, I.2.8 [Artificial Intelligence]: Problem Solving, Control Methods, and Search – Heuristic methods.*

***Conference**: The paper is selected from XVth International Conference "Knowledge-Dialogue-Solution" KDS 2009, Varna, Bulgaria, June-July 2009*

Введение

Трудности, возникающие при решении задач комбинаторной оптимизации, общеизвестны. Практически все они принадлежат к классу NP-сложных задач. Вследствие быстро возрастающей вычислительной сложности при увеличении размерности существует необходимость в разработки методов и алгоритмов, которые позволяли бы получать пригодные для использования решения задачи при условии использования ограниченных вычислительных ресурсов. Одним из перспективных направлений развития алгоритмов для решения задач комбинаторной оптимизации является разработка метаэвристических (гибридных) алгоритмов. Обобщая варианты определений [1,2], можно сказать, что метаэвристики являются высокоуровневыми стратегиями для исследования пространства решений с помощью использования различных базовых методов. Большинство известных метаэвристик базируются на объединении популяционных алгоритмов – в первую очередь генетических алгоритмов – и алгоритмов поиска в окрестностях (локальный поиск, табу-поиск, имитационный отжиг) [2]. Одним из метаэврестических алгоритмов, показавших высокую производительность при решении различных задач комбинаторной оптимизации, является *H*-метод [3], построенный на основе синтеза алгоритма ускоренного вероятностного моделирования (*G*-алгоритм) и дискретного метода, названного методом деформированных многогранников. Также нельзя не отметить активное развитие мультиагентных методов интеллектуальной оптимизации. В качестве примера можно привести метод оптимизации муравьиными колониями (ОМК) [4], метод пчелиной колонии [5] и другие. Данные методы основаны на моделировании поведения насекомых, птиц, животных, поведение которых носит коллективных характер, за счет чего достигается эффект называемый коллективным интеллектом. В основе поведения данных систем лежит самоорганизация, которая представляет собой множество динамических механизмов. В соответствии с данными механизмами система регулируется на глобальном уровне за счет взаимодействия ее компонентов на нижнем уровне без прямого взаимодействия.

Ниже предлагается метаэвристический алгоритм, построенный на основе *H*-метода и использующий мультиагентный подход для исследования пространства решений.

Общая схема *H*-метода

Под задачей комбинаторной оптимизации далее будем понимать поиск такого $x \in \Omega$, при котором заданная целевая функция *f(x)* достигает оптимума, где Ω – некоторое локально конечное пространство.

Основная идея *H*-метода состоит в следующем. Формируется начальное множество решений, которое играет роль, отдаленно напоминающую роль многогранника в методе Нелдера-Мида [6]. Из этого множества некоторым образом выбираются несколько пар точек, для каждой из которых строится полуинтервал в пространстве вариантов решения задачи, проходящий через выбранные точки. Далее на нем ищется наилучшая с точки зрения целевой функции точка. Найденный вариант решения передается в качестве начального приближения для некоторого метода локальной оптимизации ***Local_search***. В результате таких действий находится совокупность улучшающих точек. Эти точки включатся в многогранник вместо "наихудших" с точки зрения некоторого выбранного критерия оптимальности, и описанная процедура повторяется до выполнении критерия остановки.

В данном методе существенную роль играет понятие полуинтервала, которое вводится на основании *d*-отрезка [7]. Пусть на пространстве Ω определена метрика *d*.

Определение 1. *d*-отрезком /x, y/, который соединяет две произвольные точки x, $y \in \Omega$, называется упорядоченная совокупность точек $x_i \in \Omega$, $i=1,...,k$, которые удовлетворяют условиям:

1) $d(x, x_i)+d(x_i, y)=d(x, y)$, $i=1,...,k$,
2) $x_1=x$, $x_k=y$, а $d(x, x_i)< d(x, x_{i+1})$, $i=1,...,k-1$,
3) не существует такой точки $z \in \Omega$, что $d(x_i, z) + d(z, x_{i+1}) = d(x_i, x_{i+1})$, $z \neq x_i$, $z \neq x_{i+1}$, $i=1,...,k-1$.

В дальнейшем будем рассматривать такие пространства, для которых $/x, y/ \setminus \{x, y\} \neq \varnothing$, если $d(x, y)>h$, где $h=\inf(d(x, y))$ для $x \neq y$ (в работе [7] они названы регулярными в метрике *d*).

Определение 2. Полуинтервал $< x, y /$ определим как совокупность $< x, y / = / x, y / \setminus \{x\}$.

Приведем более формально вычислительную схему *H*-метода, используя терминологию эволюционных вычислений (рис.).

В предлагаемом алгоритме используются параметры:

m – число вершин многогранника (число особей в популяции– в терминах популяционных алгоритмов);

k – количество пар вершин, которые выбираются для построения полуинтервалов;

l – количество вершин, которые подвергаются мутации;

ОтборПопуляции – процедура отбора, например, $(m+(k+l))$ – стратегия: выбор *m* точек из $(m+k+l)$ точек популяции;

L (*y*) – заданная окрестность точки $y \in \Omega$.

procedure *H*-Algorithm (x);
begin
$h := 0;\ P^0 = \varnothing;$
for *j*:=1 **to** *m* **do**
x:=ГенерацияДопустимогоРешения;
$P^h := P^h \cup$ ***Local_search***$(x);$
endfor; { формирование начальной популяции }
repeat
$P := P^h$;
for *i*:=1 **to** *k* **do**
ОтборДляВариации $(x, y \in P)$;
$z := \arg\min\{ f(u) : u \in < x, x^\infty / \setminus L(y) \,\&\, y \in < x, x^\infty / \};$
$P^h := P^h \cup$ ***Local_search***$(z);$
endfor; {сформирована популяция из (*m*+*k*) точек}
for *i*:=1 **to** *l* ***d*o**
ОтборДляМутации $(x \in P)$;
x:=Мутация (*x*);
$P^h := P^h \cup$ ***Local_search***$(x);$
endfor; {сформирована популяция из *m*+*k*+*l* точек}
$P^h :=$ ОтборПопуляции(P);
$h := h+1$;
until не выполняется условие останова;
$x := \arg\min\{f(y) : y \in P\}$;
return *x*;
end.

Рисунок. Схема *H*-метода

Мультиагентный *H*-метод

Важную роль для *H*-метода играет способ построения полуинтервалов $< x, x^\infty /$ в комбинаторных пространствах. Такие пространства имеют две следующие характерные особенности. Во-первых, для произвольного $x \in \Omega$ "максимально удаленной" является точка, удовлетворяющая условию: $x^\infty = \max\{d(x,u) : u \in \Omega\}$, где $d(x,u)$ – метрика на Ω. Отметим, что зачастую это значение равняется диаметру пространства Ω. Во-вторых, между двумя не соседними точками $x, y \in P$ можно построить не один, а несколько *d*-отрезков. Поэтому при реализации *H*-метода необходимо отличать случай, когда из множества допустимых *d*-отрезков $/ x, y /$, $d(x,y) > 2$, всегда по некоторому конкретному правилу/способу построения выбирается только один *d*-отрезок, или когда могут рассматриваться все возможные *d*-отрезки. Вариант с одним возможным *d*-отрезком приводит к быстрой сходимости алгоритма, тогда как вариант с рассмотрением всех возможных *d*-отрезков приводит к более качественному анализу пространства в определенной области. Однако, исходя из схемы алгоритмы, можно сказать, что второй вариант будет отличаться повышенной трудоемкостью, так как для каждого полуинтервала, построенного на основании *d*-отрезка, будет осуществляться поиск наилучшей точки с последующим применением к ней алгоритма локального поиска. Для решения этой проблемы и использования преимуществ мультиагентных методов предлагается строить агентами несколько полуинтервалов одновременно, используя при этом специальную модель задачи, и выбирать лучшую точку с учетом всех полуинтервалов. В роли агента выступает алгоритм пошагового построения полуинтервала, который при выборе следующей точки из допустимого множества использует определенное вероятностное правило.

Рассмотрим пространство перестановок, как одно из наиболее часто используемого пространства в задачах комбинаторной оптимизации. Вероятность выбора для точки x^k очередной точки $y^{ij} \in < x, x^{\infty}$ / такой, что $y^{ij} = \omega^{ij} x^k$, где ω^{ij} - оператор транспозиции (или подобный ему в случае другой метрики), с помощью которого генерируются соседние для x^k точки, будем рассчитывать на основании:

1. эвристических проблемно-зависимых значений, например $\Delta_{ij} = \dfrac{f(x^k) - f(y^{ij})}{f(x^k)}$;
2. степени «желательности» оператора ω, которая рассчитывается на основании модели задачи и использования накопленного опыта.

Для расчета степени «желательности» оператора ω^{ij} транспозиции для перестановки $x = (x_1, \ldots, x_n)$ были предложены два варианта.

Модель задачи представляется в виде матрицы (τ_{ij}) размерности $n \times n$ (*n* – размерность задачи), где элемент τ_{ij} определяет «желательность» следования *j* после *i* (в варианте решения задачи).

Вариант 1. Транспозиция $i \leftrightarrow j$ компонентов x_i и x_j перестановки *x* будет определять следующую величину степени «желательности»:

$$\nabla_{ij} = \tau_{x_{i-1}x_j} + \tau_{x_j x_{i+1}} + \tau_{x_{j-1}x_i} + \tau_{x_i x_{j+1}} - \tau_{x_{i-1}x_i} - \tau_{x_i x_{i+1}} - \tau_{x_{j-1}x_j} - \tau_{x_j x_{j+1}} .$$

Данный метод расчета «желательности» имеет определенные аналогии с расчетом вероятности включения очередной вершины в маршрут на основании количества феромона в ОМК.

Вариант 2. Транспозиция $i \leftrightarrow j$ компонентов x_i и x_j перестановки *x* будет определять следующую величину степени «желательности»: $\nabla_{ij} = \tau_{x_i j} + \tau_{x_j i} - \tau_{x_i i} - \tau_{x_j j}$. Данный метод расчета «желательности» имеет определенные аналогии с известным алгоритмом оценки распределения (Estimation of Distribution Algorithm) [8].

Оба варианта предполагают одинаковую процедуру обновления значений матрицы (τ_{ij}). Она схожа с процедурой обновления феромонных следов в циклическом ОМК и состоит из двух этапов:

1. увеличение пропорционально найденному решению элементов матрицы, которые определенным образом ассоциируются с найденным решением;
2. применение уменьшающего коэффициента ко всем элементам (τ_{ij}) (испарение феромона).

Пусть α и β – степени значимости величин Δ_{ij} и ∇_{ij} , а $J_{st} = \{(s,t) : y^{st} \in \Pi(x^k)\}$. Тогда выбор оператора транспозиции ω^{ij} , который осуществляет создание очередной точки из допустимого множества $\Pi(x^k)$ для продолжения построения полуинтервала, будет определяться на основании вероятности p_{ij}, для вычисления которой предлагается следующее выражение:

$$p_{ij} = \min\left\{1, e^{\alpha\Delta_{ij}+\beta\nabla_{ij}} / \sum_{(s,t):y^{st}\in\Pi\left(x^k\right)} e^{\alpha\Delta_{st}+\beta\nabla_{st}}\right\}.$$

Несложно увидеть, что данное выражение может быть преобразовано к следующему виду:

$$p_{ij} = \min\left\{1, 1/\sum_{(s,t):y^{st}\in\Pi\left(x^k\right)} e^{\alpha\left(\Delta_{st}-\Delta_{ij}\right)+\beta\left(\nabla_{st}-\nabla_{ij}\right)}\right\}. \quad (1)$$

Опишем схему предложенного алгоритма, который назовем мультиагентным *H*-методом или *MH*-методом.

1. Инициализация (τ_{ij}): $\tau_{ij} = \tau_{\min}$, $i, j = \overline{1,n}$. *v*:= <количество агентов >

   Другие начальные действия для обычного *H*-метода.

2. Очередная итерация:

2.1 Построение и обработка полуинтервала каждый агентом $u \in \{1,...,v\}$:

a) строим полуинтервал: на каждом шаге с вероятностью p_{ij}, рассчитанной по (1), выбираем следующую точку из $\Pi\left(x^k\right)$;

b) находим наилучшую точку $\tilde{x}^u$ на построенном полуинтервале.

2.2 Выбор лучшей (или по правилу Метрополиса) точки $\tilde{x}$ из множества $\tilde{x}^u$.

2.3 Шаги обычного *H*-метода.

2.4 Обновление (τ_{ij}).

3. Если критерий завершения не выполняется, переход на п.2.

Вычислительный эксперимент

Для анализа практической эффективности предложенного алгоритма был проведен вычислительный эксперимент по решению задач коммивояжера (ЗК) и квадратичных задач о назначениях (КЗН). Задачи были взяты из библиотек QAPLIB [9] и TSPLIB [10], в которых для каждой задачи приведено наилучшее известное на данный момент решение. Это позволило не только получить сравнительные оценки по сравнению с *H-VNSG* алгоритмом (одной из самых успешных реализаций *H*-метода), но и оценить точность работы *MH*-метода.

Для всех алгоритмов использовались следующие значения параметров: *m*=25, *k*=5, *l*=0. Радиус окрестности *L(y)* равнялся 2. Для *MH*-метода коэффициент испарения ρ=0.5, а α=1, β=3, *v*=10. В качестве алгоритма локального поиска использовался, как и *H-VNSG* алгоритме, модифицированный алгоритм локального поиска с переменными окрестностями [11]. С целью получения статистически достоверных результатов, для каждой задачи проводилась серия экспериментов с одинаковыми параметрами алгоритма и разными начальными приближениями. Объем серии для каждой задачи составлял 50. Результаты решения ряда задач приведены в табл.1 и табл.2. Здесь *N* – размерность задачи, *f** - лучшее известное решение задачи, *ε* – средняя относительная погрешность алгоритма (%), ε_{min} – лучшее значение ε в серии экспериментов, t – среднее время, за которое было найдено лучшее решение в этой серии.

Анализ полученных результатов свидетельствует, что для большинства задач алгоритмом *MH*-метода позволил повысить точность решения, для остальных – показал такой же результат, что и *H-VNSG* алгоритм. Также следует отметить, что предложенная схема алгоритма, включая и использованный встроенный алгоритм локального поиска, является универсальной для всех задач комбинаторной оптимизации, что значительно расширяет область ее применения.

Таблица 1. Результаты решения ЗК

Название задачи	N	f^*	*H-VNSG* алгоритм			*MH*-алгоритм		
			ε	ε_{min}	t	ε	ε_{min}	t
Els19	19	17212548	0,00	0,00	16,88	0,00	0,00	18,75
chr20a	20	2192	0,20	0,00	20,66	0,00	0,00	19,39
Chr20b	20	2298	1,38	0,00	25,20	0,2	0,00	12,86
Chr20c	20	14142	0,00	0,00	2,82	0,00	0,00	5,95
Nug21	21	2438	0,00	0,00	11,35	0,00	0,00	12,59
Lip40a	40	31538	0,50	0,00	23,79	0,00	0,00	20,20
Lip40b	40	476581	0,00	0,00	5,63	0,00	0,00	5,65
Lipa50a	50	62093	0,27	0,00	32,70	0,23	0,00	39,34
Lipa50b	50	1210244	0,00	0,00	16,06	0,00	0,00	16,06
Sco81	81	90998	0,32	0,19	61,94	0,24	0,17	84,87
Will100	100	273038	0,23	0,18	91,92	0,17	0,14	199,54
Tai100b	100	1185996137	0,34	0,21	153,84	0,20	0,08	221,39

Таблица 2. Результаты решения КЗН

Название задачи	N	f^*	*H-VNSG* алгоритм			*MH*-алгоритм		
			ε	ε_{min}	t	ε	ε_{min}	t
Ftv33	34	1286	0,00	0,00	7,94	0,00	0,00	10,57
Ftv44	45	1613	1,43	0,00	27,69	1,36	0,00	47,99
Ry48p	48	14422	0,61	0,17	50,83	0,55	0,00	51,41
Eil51	51	426	0,70	0,00	55,31	0,47	0,00	50,1
Berlin52	52	7542	0,58	0,58	38,52	0,58	0,58	40,70
Ftv55	56	1608	3,65	1,74	58,33	3,26	1,37	58,68
Ftv64	65	1839	3,81	1,30	100,87	3,24	0,38	115,42
Sto70	70	675	1,41	0,00	235,94	1,28	0,00	212,33
KroA100	100	21282	1,30	0,79	983,13	0,86	0,00	789,30

Заключение

Предложенный метод, который базируется на мультиагентном подходе и идеях *H*-метода, показал свою высокую эффективность при экспериментальном исследовании. Комбинирование механизма сканирования пространства и процедуры обучения позволило повысить точность решения ряда известных задач коммивояжера и квадратичных задач о назначениях в сравнении с *H*- методом.

Целью дальнейших исследований можешь стать исследование эффективности алгоритма в зависимости от изменения процедуры обновления значении матрицы «желательности», количества агентов и других параметров *MH*-метода. Представляет интерес получение оценок сходимости и трудоемкости.

Благодарности

Статья частично финансированна из проекта **ITHEA XXI** Института Информационных теории и Приложений FOI ITHEA и Консорциума FOI Bulgaria (www.ithea.org, www.foibg.com).

Список литературы

[1] Gendreau M., Potvin J-Y. Metaheuristics in Combinatorial Optimization // Annals of Operations Research. – 2005. – 140. – P. 189–213.

[2] Hoos H.H., Stützle T. Stochastic Local Search: Foundations and Applications. – San Francisco: Morgan Kaufmann Publ., 2005.

[3] Гуляницкий Л.Ф., Гобов Д.А. Применение *H*-метода для решения задач комбинаторной оптимизации на перестановках// Системные исследования и информационные технологии. –2007. –№2. –С. 74–86.

[4] Dorigo M., Stützle T. Ant Colony Optimization. – Cambridge: MIT Press, MA, 2004.

[5] Chong C.S., Low M.Y.H. A bee colony optimization algorithm to job shop scheduling // Winter Simulation Conference: Proceedings of the 38th conference on Winter Simulation. – Monterey: Monterey Press, 2006. – P. 1954-1961.

[6] Химмельблау Д. Прикладное нелинейное программирование. – М.: Мир, 1974.

[7] Сергиенко И.В., Гуляницкий Л.Ф. Фронтальные алгоритмы для многопроцессорных ЭВМ // Кибернетика. – 1981. – №6. – С.1–4.

[8] Larrañga P., Lozano J. A. (Eds.) Estimation of distribution algorithms: A new tool for evolutionary computation. – Kluwer Academic Publishers, 2001.

[9] QAPLIB // http://www.opt.math.tu-graz.ac.at/qaplib/

[10] http://www.akira.ruc.dk/~keld/research/LKH/LKH-1.2/TSPLIB/

[11] Hansen P., Mladenovic N. Variable neighborhood search: Principles and applications // European Journal of Operational Research. – 2001. – №3. – P. 449–467.

Информация об авторах

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

***Денис Гобов** – аспирант, Институт кибернетики им. В.М.Глушкова НАН Украины, пр. Глушкова, 40, Киев, 03680, Украина. e-mail: gda1980@ukr.net*