ИСПОЛЬЗОВАНИЕ АЛГОРИТМА УСКОРЕННОГО ВЕРОЯТНОСТНОГО МОДЕЛИРОВАНИЯ В СХЕМЕ ТАБУ-ПОИСКА

Леонид Гуляницкий, Александр Турчин

Аннотация: *Рассматривается метаэвристический метод комбинаторной оптимизации, основанный на использовании алгоритмов табу-поиска и ускоренного вероятностного моделирования. Излагается общая вычислительная схема предложенного метода, названного алгоритмом GS-tabu. Приведены результаты серии вычислительных экспериментов по решению известных задач коммивояжера и квадратичных задач о назначении.*

Keywords*: combinatorial optimization, stochastic local search, tabu search.*

ACM Classification Keywords*: G.1.6 Nunerical Analysis; G.2.1 Discrete Mathematics; I.2.8 Artificial Intellicence.*

Conference: *Decision Making*

Формальная постановка задач комбинаторной оптимизации

Задачи комбинаторной оптимизации (ЗКО) возникают при исследовании многих сложных процессов – например таких, как исследование организации сложных систем. Такие задачи – наряду с методами их решения – имеют ключевое значение при решении проблем проектирования и размещения объектов, планирования экспериментов, управления процессами обработки данных, принятия решений в экономике и бизнесе и др. [1-2].

В настоящее время общепринятым является определение ЗКО, предложенное Пападимитриу и Стайглицем [2]: необходимо найти $x_* \in X$ такое, что

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

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

Такое определение однозначно относит к ЗКО все проблемы оптимизации на конечных множествах, однако в случае бесконечных пространств не позволяет четко классифицировать задачи оптимизации по структуре элементов пространства решений. К.Берж [3] 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 [3]. Комбинаторной конфигурацией называется отображение $\varphi : U \to V$, 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.

В работе [4] предлагается следующее обобщение схемы Бержа. Пусть заданы $Y=\{1, \dots, m\}$, Z – дискретное, в частности, конечное пространство (назовем его образующим), φ – гомоморфизм,

$\varphi : Y \to Z$, 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:

$$\kappa = (\varphi, X_{(1)}, \Omega), \quad X_{(1)} \equiv Z .$$

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.

Специфика большинства ЗКО типа (1) состоит в наличии большого количества экстремумов целевой функции. Кроме того, как правило эти задачи относятся к числу *NP*-трудных, так что точное их решение весьма проблематично даже с использованием современных и перспективных компьютеров.

Среди наиболее эффективных подходов к решению ЗКО следует выделить использование *метаэвристических алгоритмов*, или *метаэвристик*. Существует несколько подходов к классификации метаэвристических алгоритмов, среди которых наиболее удачным представляется деление на *метаэвристики 1-го рода* и *метаэвристики 2-го рода*; в свою очередь, метаэвристики 1-го рода представлены классами *траекторных* и *популяционных* метаэвристик. Траекторными метаэвристиками являются методы имитационного отжига, табу-поиск [5], алгоритмы ускоренного вероятностного моделирования [6]. Класс популяционных метаэвристик представляют генетические алгоритмы, алгоритмы оптимизации муравьиными колониями, иммунные алгоритмы [7,8].

Метаэвристический алгоритм (GS-tabu)

В работе [9] предложен подход к построению метаэвристического метода на основе алгоритмов табу-поиска и ускоренного вероятностного моделирования. В качестве базового был выбран метод ускоренного вероятностного моделирования (*GS-method* [10]), который реализует стратегию «золотого сечения». Для построенного алгоритма предложено название GS-tabu.

Алгоритмы ускоренного вероятностного моделирования были разработаны для преодоления таких недостатков методов локального поиска, как преждевременная сходимость алгоритма и большая зависимость конечного результата от выбора начального приближения решения. А среди алгоритмов, которые принадлежат к классу ускоренного вероятностного моделирования, следует выделить G-алгоритмы, которые успешно применялись для решения многих сложных практических задач [6].

Главная идея алгоритма табу-поиска состоит в постоянной локальной модификации текущего решения задачи при запоминании произведенных изменений – чтобы предотвратить алгоритм от повторного построения неэффективных решений и зацикливания процедуры поиска решения [5]. Осуществленные модификации запоминаются в специальных банках памяти, которые называются «табу-списками» (tabu list). Общая схема предлагаемого метаэвристического метода GS-tabu приведена на рис.1.

При практической апробации GS-tabu было установлено, что наилучшие результаты были получены при размере банка памяти около *n*/3, где *n*-размерность задачи. Кроме того, в нашем случае величина μ_t определяется как меньшая (левая) из двух точек, которые реализовывают «золотое сечение» отрезка [μ_t,1]. Таким образом, правило «золотого сечения» задает динамику приближения левой границы этого отрезка к 1 [9].

```
procedure  GS-tabu(x)
    begin
      x^0 := некоторый начальный допустимый вариант решения из X;
      μ_0 := 0;   h := 0;   t := 0;   x_rec:= x^0;   f_rec := f( x^0 );   ТабуСписок=∅;
      while окрестность текущего решения  L(x^h)  не просмотрена полностью do begin
        while не выполнено условие равновесия  do begin
          y := ГенерированиеСледующейТочкиОкрестности L(x^h);
          ПрохождениеПроверкиТабуСписка(y);
          Вычисление Φ(x^h, y);
          p := (1- μ_t) Φ(x^h, y);
          ξ := random[0,1];
          if  p ≥ ξ  then
                         h:=h+1;  x^h := y;
                         if  f_rec > f(x^h)  then  x_rec := x^h;  f_rec := f( x^h ) end if
          end if;
          ОбновлениеТабуСписка;
        end;
        ФормированиеОчередногоЗначения μ_{t+1};        t := t + 1;
      end;
      return   x= x_rec;
    end
```

Рис. 1. Вычислительная схема алгоритма GS-tabu.

Правилом останова может служить: окончание перебора всех точек в окрестности без реализованного перехода в новую точку; ограничение по продолжительности работы алгоритма; достижение требуемой точности – при известной нижней границе целевой функции. Еще одно из распространенных правил – сравнение разности максимального и минимального значений целевой функции с максимальным значением изменения этой функции при данном значении μ: если это соотношение стремится к единице, то вычисления завершаются.

Исследование практической эфективности GS-tabu

Для исследования практической эффективности GS-tabu по сравнению с алгоритмом GS-method были выбраны два наиболее известных класса ЗКО: задачи коммивояжера из библиотеки TSPLIB95 [11] и квадратичные задачи о назначениях из библиотеки QAPLIB [12]. Результаты решения серий реальных и тестовых задач алгоритмами GS-method и GS-tabu приведены в табл.1 (задача коммивояжера) и табл. 2 (квадратичная задача о назначениях). Для каждой задачи приведено ее библиотечное имя [11,12], размерность (*n*) и наилучшее из известных решений.

Задача	n	Наилучшее известное решение	Методы			
			GS-method	Время,с	GS-tabu	Время,с
burmal4	14	**3 323**	3 340	1,2	3 323	1,2
ulyssesl6	16	**6 850**	6 876	2,1	6 891	2,1
grl7	17	**2 085**	2 114	1,9	2 092	1,9
ulysses 22	22	**7 013**	7 116	14,0	7 054	14,0
gr24	24	**1 272**	1 282	14,5	1 286	14,5
fri26	26	**937**	959	19,5	943	20,5
bays29	29	**2 020**	2 052	28,8	2 020	24,8
sui ss42	42	**1 273**	1 278	42,3	1 286	42,3
dantzig42	42	**699**	710	50,0	699	50,0
att48	48	**10 628**	10 930	72,6	10 670	76,6
hk48	48	**11 461**	11 464	78,4	11 603	78,4
eil51	51	**426**	431	106,0	430	106,0
brazil58	58	**25 395**	25 927	123,1	25 460	111,0
st 70	70	**675**	688	112,3	676	114,3
pr76	76	**108 150**	110 571	144,1	109 940	147,1
rat 99	99	**1 211**	1 244	173,0	1 251	153,3
kroAlOO	100	**21 282**	22 243	152,1	21 826	160,1
kroBlOO	100	**22 141**	22 609	154,0	22 540	159,0
kroClOO	100	**20 749**	22 095	183,0	21 148	175,5
prl24	124	**50 030**	52 067	241,3	52 124	252,3
prl52	152	**73 682**	76 533	288,8	76 107	293,8
rati95	195	**2 323**	2 412	320,8	2 380	323,8

Таблица 1. Результаты решения задачи коммивояжера

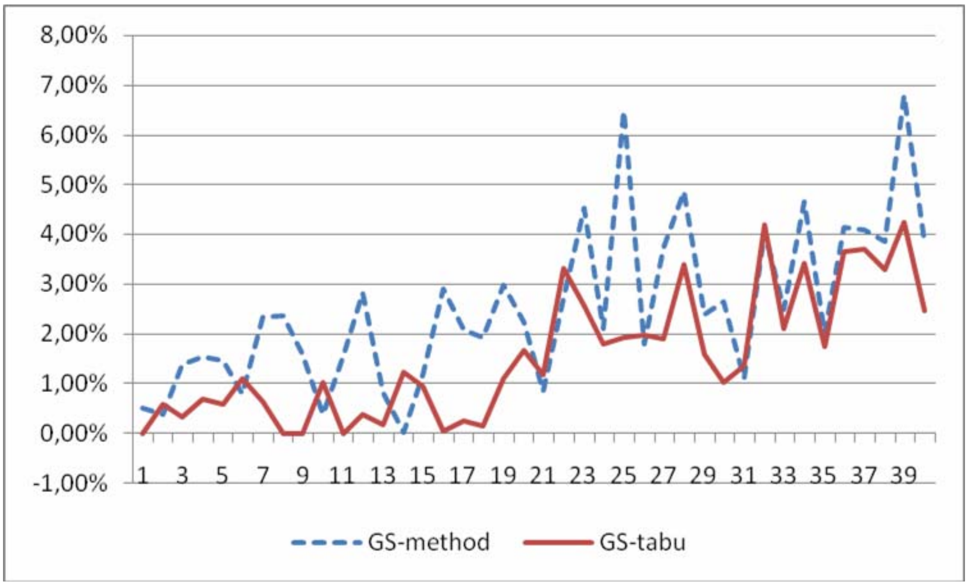

Рис. 2. Относительная точность решений

Задача	n	Наилучшее известное решение	Методы			
			GS-method	Время,с	GS-tabu	Время,с
chr15a	15	**9 896**	10 112	1,8	9 956	1,8
chr15b	15	**7 990**	8 064	3,2	8 011	2,9
chr15c	15	**9 504**	9 611	2,5	9 504	2,8
bur26d	26	**3 821 225**	3 929 672	21,2	3 911 457	23,6
bur26e	26	**5 386 879**	5 478 936	22,8	5 502 452	25,9
bur26f	26	**3 782 044**	3 802 312	28,3	3 788 565	34,2
esc64a	64	**116**	121	89,1	118	92,0
lipa70a	70	**169 755**	175 341	107,4	170 753	112,5
lipa70b	70	**4 603 200**	4 781 819	111,3	4 711 844	132,8
lipa80a	80	**253 195**	255 221	129,5	254 198	144,4
lipa80b	80	**7 763 962**	7 945 390	141,9	7 878 543	150,1
lipa90a	90	**360 630**	361 717	166,5	363 991	182,1
lipa90b	90	**12 490 441**	12 832 091	170,4	12 774 910	188,1
sko100a	100	**152 002**	156 748	179,9	154 340	202,2
sko100b	100	**153 890**	154 120	212,0	157 463	195,5
sko100c	100	**147 862**	155 543	200,4	149 341	225,7
sko100d	100	**149 576**	153 288	192,1	151 922	200,1
sko100e	100	**149 150**	154 012	188,4	149 781	211,4

Таблица 1. Результаты решения квадратичной задачи о назначениях

На рис. 2 приведен график отклонений по точности найденных метаэвристическим методом и GS-алгоритмом вариантов решения от наилучших из известных решений, упорядоченных по возрастанию размерностей задач.

Вычислительный эксперимент проводился на ПК Athlon 64 X2 4Ghz, 2 Gb оперативной памяти. Программный комплекс, реализующий оба метода, написан с помощью среды разработки Delphi.

Заключение

Результаты вычислительного эксперимента свидетельствуют о том, что применение предложенного метаэвристического метода, построенного на базе алгоритмов GS-method и табу-поиска, позволило улучшить точность получаемых решений по сравнению с использованием отдельных алгоритмов, входящих в его схему.

Направление дальнейших исследований будет сосредоточено на исследовании как практической, так и теоретической эффективности нового метода. Особое внимание предполагается уделить возможности построения с помощью GS-tabu новых эффективных метаэвристик 2-го рода, предназначенных для решения широкого круга прикладных ЗКО.

Литература

[1] Сергиенко И.В. Математические модели и методы решения задач дискретной оптимизации. – К.: Наукова думка, 1985. – 384 с.

[2] Пападимитриу Х., Стайглиц К. Комбинаторная оптимизация. Алгоритмы и сложность. – М.: Мир, 1985. – 512 с.

[3] Berge C. Principes de combinatoire. – Paris: Dunod, 1968. – 146 p.

[4] Гуляницкий Л.Ф., Сергиенко И.В. Метаэвристический метод деформируемого многогранника в комбинаторной оптимизации // Кибернетика и системный анализ. – 2007. – № 6 . – С. 70–79.

[5] Glover F. Heuristics for Integer Programming using surrogate constrains // Decision Sciences. – 1977. – N 8. – P.156-166.

[6] Гуляницкий 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.

[7] Blum C., Roli A. Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison // ACM Computing Surveys. – 2003. – 35, No. 3. – P. 268–308.

[8] Hoos H.H., Stützle T. Stochastic Local Search: Foundations and Applications. – San Francisco: Morgan Kaufmann Publ., 2005. – 658 p

[9] Турчин О. Використання алгоритму імовірнісного моделювання у схемі методу табу-пошуку // Пр. Між. симп. «Питання оптимізації обчислень (ПОО-XXXIII)». – К.: ІК ім.В.М.Глушкова НАН України, 2007.– Р. 73–77.

[10] Hulyanitsky L., Turchin O. "Golden section" rule in probabilistic modeling algorithms // Вісник Нац. ун-ту "Львівська політехніка". – 2001. – №415. – С. 50-53.

[11] http://softlib.rice.edu/pub/tsplib

[12] http://www.opt.math.tu-graz.ac.at/~karisch/qaplib

Сведения об авторах

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

Александр Турчин – *Институт кибернетики им. В.М.Глушкова НАН Украины, пр-т Глушкова, 40, Киев, 03680, Украина. e-mail: turchin@ua.fm*