

13-16

组合优化问题

启发式搜索

遗传算法

④

计算机科学 1998 Vol. 25 No. 2

组合优化问题的启发式搜索^{*})

Metaheuristics in Combinatorial Optimization

张鸿宾

(北京工业大学计算机学院 北京 100044)

0224

摘 要 In recent years, Simulated Annealing, Genetic Algorithm and Tabu Search have achieved notable success in combinatorial optimization. In this paper, we review and compare characteristics of these algorithms, and their performance. We emphasize the intensification and diversification problem in metaheuristics, and give some insight in application of these algorithms.

关键词 Combinatorial optimization, Simulated annealing, Genetic algorithm, Tabu search, Meta-heuristics search.

一、引言

组合优化是一个离散最优化问题,在规划,调度,资源分配,决策等问题中有着非常广泛的应用。人们已经认识到,组合优化问题的计算复杂度高,属于 NP 难一类的问题,除了枚举一部分解空间之外,没有更好的解法^[1]。在问题的规模较小时,可以用运筹学的经典算法(如整数规划、动态规划、分支定界或切多面体等)方法求解。当问题的规模增大时,由于解的数目呈指数函数增长,要求准确的最优解实际上已不可能。因此,从实际应用的角度出发,能够得到较好解的近似算法或以一定的概率保证解的质量的随机算法的研究越来越受到重视。除了经典的方法之外,人们利用人工智能、神经网络、模糊系统等领域的一些新工具,或者从生物进化、物理过程中吸取一些启发,研究组合优化问题的新解法。近一些年出现的模拟退火、Tabu 搜索和遗传算法等在组合优化问题中获得了广泛的应用,在一些难解的组合优化问题上取得了显著的成果,超过了过去用其它方法所得到的最好的近似解。国外一些研究人员把这些新的解法称为超启发(metaheuristics)或近代启发(modern heuristics)。所谓“启发式”解法是以求解问题的知识和经验指导求解的过程。称为“超

(meta)”是因为这些解法是从原来的启发式解法中变化而来的,其能力更强。

二、组合优化问题的启发式求解

1. 组合优化问题

组合优化问题是在变量的一些约束条件下,求使某个目标函数最小(或最大)的离散变量的组合。一般可以描述如下。

目标函数: $\min_x f(x)$,

约束条件: $x \in F, F \subseteq X$ 。

式中, X 是基本空间, F 是可行解的集合,它是离散变量的一些组合的集合。

组合优化问题的解法和连续变量的最优化问题不同,一般只能采取枚举的方法。可行解的总数虽然有限,但当问题的规模增大,变量数超过几十个时,组合的可行解的总数变得非常大,要全部枚举这些解而求出最优的,事实上已经不可能。如何削减解的搜索范围,同时又能得到较好的近似最优解,就成了求解组合优化问题时的一个重要问题。下面分析一些常用的求近似解的启发式搜索方法。

2. 局部搜索法(Local Search: LS)

局部搜索是组合优化求解中一种最基本的方法。可以用贪心法、随机法或其它适当的方法产生一

^{*}) 国家和北京市自然科学基金资助课题

个初始解 x 。把 x 作一些变化而得到的解的集合 $N(x)$ 称为解 x 的近邻解。局部搜索法是从某个初始解 x 开始,搜索 $N(x)$ 内更好的解。若有更好的解,把它作为新的暂定解 x ,继续搜索 $N(x)$ 中更好的解。局部搜索法 LS 描述如下:

1. 求一初始解 x , 把 x 作暂定解, 令 $k=1$ 。
2. 若 $N(x)$ 内有比 x 更好的解 y , 令 $x=y, k=k+1$, Goto2; 若不存在更好的解, 输出暂定解, 停止。

在实施局部搜索算法时, 需要考虑以下几个细节问题:

(1) 如何定义 $N(x)$ 。从计算效率考虑, $N(x)$ 应该不太大, 而且含有比 x 更好解的可能性较大。

(2) 在 $N(x)$ 之内如何搜索, 是系统地搜索还是随机地搜索比 x 更好的解 y 。

(3) 暂定解的更新是在找到一个较好的解后立即进行, 还是在搜索完 $N(x)$ 、得到 $N(x)$ 中最好的解后再进行。

这些细节在很大程度上影响了局部搜索法的效率和解的质量。

局部搜索法是求解组合优化问题的最基本方法。但是, 它得到的解是一种局部最优解。在未搜索到的区域内, 有可能存在着更好的解。为了改进局部搜索法的弱点, 人们提出了以下的一些方法:

(1) 从多个不同的初始解出发, 从最终解中选择最好的一个。

(2) 在搜索中加入随机的行为。

(3) 在搜索中允许向较差的解移动。

(4) 同时保留若干个候选解。

(5) 在搜索过程中利用解的履历信息。

在局部搜索法的基础上, 采用上述不同的改进措施, 就形成了以下各种超启发式搜索的方法。

3. 多起点局部搜索法 (Multi-start Local Search; MLS)

多起点局部搜索法是对多个不同的初始解使用局部搜索法, 然后从所得解中选取最优的一种方法。多起点局部搜索法 MLS 如下:

1. 令 $k=1$ 。
2. 若不满足终止条件, 生成新的初始解, 应用 LS 法求出局部最优解, $k=k+1$, Goto2; 若满足终止条件, 输出所得到的最好的近似解, 终止。

算法的终止条件可以使用预先设定的重复次数、运行时间或判断是否得到了满意的解等。初始解的生成可以采用随机的方法, 也可以采用贪心法生成多个初始解, 然后从中随机地选出一些作为初始解。这种类型的算法称为 GRASP (Greedy Random Adaptive Search Procedure)^[2,3]。

初始解的选择对多起点局部搜索法的成功来说是非常关键的。这些初始解应该是一些较好的近似解, 同时应该保持解的广泛性和多样性。

4. 模拟退火法 (Simulated Annealing; SA)

模拟退火法是在局部搜索过程中引入了随机的因素^[4]。在局部搜索中, 当某个解的目标函数即使变坏时, 仍然以一定的概率保留向这个解移动的可能性, 以避免陷入局部最优解。这种算法和物理上的退火过程很相似, 整个过程由一个称为温度的参数 t 来控制。模拟退火法 SA 如下:

1. 初始化: 设定冻结温度 $t_0 (>0)$, 初始温度 $t (> t_0)$, 同一温度下的重复次数 L , 温度下降率 $r (0 < r < 1)$ 。求一初始解 x , 把 x 作为暂定解, $k=1$ 。
2. 随机局部搜索:
 - A. 以下的运算反复进行 L 次:
 - a. 随机选一 $y \in N(x)$;
 - b. 令 $\Delta = f(y) - f(x)$ (f 为目标函数)。若 $\Delta \leq 0$, 令 $x=y$; 若 $\Delta > 0$ 以概率 $\exp(-\Delta/t)$ 令 $x=y$; 当 $\Delta < 0$ 时, 若 $f(y)$ 小于暂定解的值, 令 y 为新的暂定解;
 - B. 令 $t=rt$ 。若 $t < t_0$, 输出暂定解, 终止。否则, $k=k+1$, Goto2。

上述算法中的温度下降方式称为几何冷却法。另外一种温度下降方式是采取对数冷却法, 即 $t_k = d/\log(k+1)$, t_k 是第 k 次反复时的温度。对数冷却法有很好的理论结果。在一定条件下可以渐近地收敛于最优解^[5]。但从实际应用的效果看, 几何冷却法的实用价值更好^[6]。

另外, 有人把模拟退火中的概率机制去掉, 对于预先设置的阈值 c , 若 $\Delta \leq c$, 则 $x=y$, 否则不进行解的更新^[8]。还有人把模拟退火中的温度调节部分去掉, 只进行随机的局部搜索。据称, 这些简单的算法在一些问题上也取得了很好的结果。

5. Tabu 搜索法 (Tabu Search; TS)

把模拟退火法作一些修改, 不是从 $N(x)$ 中随机地选出一个解作为候选解, 而是从 $N(x)$ 中选取除 x 外最好的解 y , 即使解 y 不如 x 好, 也把 y 作为下一个候选解。修改后的算法就成了 Tabu 搜索的基础。然而, 当候选解从 x 移动到 y (属于 $N(x)$) 后, 在求解 $N(y)$ 内的最优解时, 有可能又回到了原来的解 x 。为了防止这种循环情况发生, 可以作一个禁止集合 T , 使 x 只能向 $N(x)-T$ 内最好的解移动。Tabu 搜索法 TS 如下:

1. 初始化: 求一初始解 x , 将 x 作为暂定解, $k=1, T=\Phi$ (空集)。
2. 解的改进:
 - A. 若 $N(x)-T=\Phi$, Goto B; 否则, 求 $N(x)-T$ 中的最优解 y , 令 $x=y$, 若 y 比暂定解好, 把 y 作为新的暂定解。
 - B. 若满足终止条件, 输出暂定解, 终止。否则, 修改 $T, k=k+1$, Goto2。

上述的 Tabu 搜索给出了一个非常灵活的算法框架。其核心思想是利用 Tabu 集合来记录求解过

程的履历,从而能调节以后的搜索过程。禁止集合一般采用以下两种常用的形式。

(1)用一个长度为某个常数 k (控制参数)的队列形成一个先进先出的禁止表(Tabu List; TL),记录刚刚搜索过的 k 个解。在以后的 k 步之内,禁止向这些解移动。

(2)记录最近搜索过的 k 个解中变量的变化方向。在以后的搜索中禁止向相反的方向移动。

实际应用中,禁止集合的作法可以非常灵活。可以把各种先验知识和搜索的履历、解的性质等以禁止集合的形式记录下来,从而控制以后的搜索过程。例如,可以把搜索中解的变量出现的频率和变化的方向存储起来,用这些信息来调节搜索过程,使得在候选解出现较多、从而是有希望的区域内进行更详细的搜索,同时有意识地向还未搜索或搜索不充分的区域移动,增加搜索的广泛性。

自从 F. Glover 提出 Tabu 搜索方法以来,在许多问题上取得了优于其它方法的结果^[9,10]。同时,由于它的灵活性,使 Tabu 搜索很容易和其它方法及特定问题的具体知识结合起来,组成面向具体对象的实用算法。Tabu 搜索的这些长处正在引起人们越来越大的重视。

6. 遗传算法(Genetic Algorithm:GA)

遗传算法在最优化,特别是组合优化问题中的应用是近些年的一个热点课题^[11,12]。遗传算法模仿生物遗传和进化的过程。它把解编码为字符串(一般是 0 和 1),把类似于遗传基因的一些行为,例如交叉重组、变异、选择和淘汰等机制引入到求解过程。和基于解的近邻 $N(x)$ 内的搜索不同,遗传算法同时保留着若干个局部最优解,通过两个或多个解的交叉重组和单个解的变异来寻找整体最优解。遗传算法 GA 如下:

1. 初始化:确定候选解的个数 p (正整数);求 p 个起始解,将其中最好的解作为暂定解; $k=1$
2. 若满足终止条件,输出暂定解,终止;否则,进行如下的运算:
 - a. 解的交叉:从 p 个候选解中选出两个(或多个),通过交叉运算产生新的解。重复该运算适当次。
 - b. 解的变异:对若干候选解进行一些随机的变形,产生新的解。
 - c. 解的改进:应用 LS 法对新生成的解进行改进,若得到了比暂定解更好的解,把它作为暂定解。
 - d. 解的淘汰:从所得的全部解中,按一定的准则(如目标函数值的大小、优度等),选出 p 个作为下一步的候选解。
 然后, $k=k+1$; Goto2.

和 Tabu 搜索一样,遗传算法也是一个框架,具体的内容可根据需要进行调整。上述算法中加入了局部搜索的运算,是为了改善 GA 的集中搜索能力较弱的缺点。省略这一步也可以。另外,在解的淘汰

和选择中,应该注意在选择好的解的同时,还应该保持解的多样性。这些问题将在下一节进行分析。

在遗传算法解的交叉重组中,要由两个(或多个)候选解生成一个新的解。交叉重组方法的适当与否直接影响遗传算法的性能。解的交叉常采用类似于遗传基因重组的方法。例如,对于用 0,1 字符串表示的解,可以采用一点交叉,两点交叉或多点交叉等方法。在组合优化的实际问题中,可以不必拘泥于上述类似生物基因重组的方法。关键的问题是如何把解 x 和 y 的好的部分(形态)传给新生成的解 z ,同时要保持新生成解的多样性。

三、算法性能的分析 and 比较

上节简单概括了局部搜索、模拟退火、Tabu 搜索和遗传算法。人们很自然地会提出一个问题,这些算法有效吗?哪一个算法最好?由于前一节给出的只是各个算法的框架,算法实施的细节留有很大的灵活余地,所以很难严格地比较各个算法的性能。草率地作出某个算法优于另一个算法的结论是不可取的。这一节我们从不同的侧面对这些算法的性能作一些分析和比较,分析这类算法成功的关键是集中搜索和广泛搜索的问题。

在启发式搜索中,为了得到好的近似解,应该在尽可能广的区域内进行搜索。在遇到暂时变坏解时,也不应该停止搜索,应该强制地向迄今尚未搜索过的区域移动。另一方面,为了缩短搜索时间,提高算法的效率,应该充分利用各种先验知识、解的履历等信息,在有希望得到最优解的附近区域进行集中地、详细地搜索。上述集中搜索和广泛搜索是矛盾的两个要求,是解的质量与搜索时间的矛盾。问题是如何找到一个好的折衷解决方法。

局部搜索的基础是人们认为在较好解的附近区域可能仍然存在着更好的解。为了增加搜索的广泛性,可以从多个初始解开始。在实际应用中,多起点的局部搜索法是一个非常有力的工具。程序设计的工作量也较少。只要确定好解 x 的近邻 $N(x)$ 就行了。在巡回售货员(TSP)问题中, Lin 和 Kernighan 的算法就是一个非常成功的例子^[13]。他们的算法在非常大规模的 TSP 中仍然得到了相当好的近似解。LK 算法实际上是一个精心设计的多起点局部搜索法。其成功的关键是他们巧妙地设计了 $N(x)$ 。如果在初始解的生成上再下一些工夫,用贪心一类的算法生成好的初始解,如 GRASP,一般还会有更好的结果。

在模拟退火中,为了避免局部最优解,增加搜索的广泛性,引入了解的移动的随机性。Johnson 等人曾对图的二分割问题,用实验详细地比较了 SA 法和有名的 KL 法^[6]。对于一定类型的图,得出了 SA 法优于 KL 法的结果。SA 法存在的弱点是,当目标函数有较深的局部最优解时,仅仅利用随机性来跳出局部最优解就无能为力了。改进的方法是采用再加热的方法^[15]。SA 法的另一个问题是,由于没有记录解的履历,有可能多次搜索到同一个解。因此,有人称它为盲目的搜索^[14]。

遗传算法的应用近年来引起了人们的关注,在一些问题中取得了很好的结果。遗传算法同时保持着若干个解,通过交叉重组或变异生成新的解。遗传算法的运算一定程度上保证了解的多样性,起到了广泛搜索的作用。但单纯的 GA 算法缺乏集中搜索的机制。为了改进 GA 算法的这个不足,在第二节的 GA 算法中,我们已经加入了局部搜索。另外,通过同时保留若干解实现的广泛搜索有时不够充分,有可能过早地收敛到同一个解。

Tabu 搜索的优点是它提供了一个非常灵活的框架,可以方便地把一些知识和规则通过禁止集合体现出来,实现搜索的集中性和广泛性。Aarts 等人对于作业调度(job shop)问题,比较了 MLS、SA、GA 和 TS 等算法的性能,得出的结果是 TS 算法在最短的时间内给出了最好的结果^[15]。在其它一些问题上,Tabu 搜索也取得了很好的结果。Tabu 搜索的另一个长处是它容易把其它的解法纳入到它的框架之内,组成更有效的算法。Tabu 搜索的应用应该引起更大的重视。我们把 Tabu 搜索应用到模式识别中的特征选择、树分类器的设计、聚类分析和平面曲线特征点的抽取等问题中,均取得了较好的效果。

小结 本文分析、比较了求解组合优化问题的局部搜索、模拟退火、Tabu 搜索和遗传算法等启发式搜索方法。分析了这类算法成功的关键——搜索的广泛性和集中性问题。这些方法为组合优化问题的近似求解提供了有力的工具。但是,它们也不是万能的。不能仅凭一些个别的实验,得出以偏概全的结论。在实际应用时,我们应该对要解决的问题有充分的了解,然后从这些算法中选择适当的算法作为工具。还必需综合考虑计算机的速度和存储能力,待解决问题的规模,所允许的计算时间,程序设计的难易程度和所需的时间等问题。本文所作的分析可作为

算法选择时的一些参考。

参考文献

- [1] C. H. Papadimitriou et al., Combinatorial optimization: algorithms and complexity. Prentice-Hall, 1982
- [2] M. Laguna et al., A greedy randomized adaptive search procedure for the two-partition problem, Operations Research, Vol. 42, 1994
- [3] T. A. Feo et al., A greedy randomized adaptive search procedure for maximum independent set, Same to [2]
- [4] S. Kirkpatrick et al., Optimization by simulated annealing, Science, Vol. 220, 1983
- [5] B. Hajek, Cooling schedules for optimal annealing, Mathematics of Operations Research. Vol. 13, 1988
- [6] D. S. Johnson et al., Optimization by simulated annealing: an experimental evaluation, Part I, Graph partitioning. Operations Research. Vol. 37, 1989
- [7] D. S. Johnson et al., Optimization by simulated annealing: an experimental evaluation, Part I, Graph coloring and number partitioning, Operations Research, Vol. 39, 1991
- [8] G. Dueck and T. Scheuer, Threshold accepting: a general purpose optimization algorithm. J. of Computational Physics. Vol. 90, 1990
- [9] F. Glover, Tabu search I, ORSA J. on Computing, Vol. 1 1989
- [10] F. Glover, Tabu search II, ORSA J. on Computing, Vol. 2, 1989
- [11] J. Holland, Adaption in natural and artificial systems, The Univ. of Michigan Press, 1975
- [12] D. E. Goldberg, Genetic algorithms in search, optimization and machine learning, Addison-Wesley, 1989
- [13] S. Lin and B. W. Kernighan, An efficient heuristic algorithm for the traveling salesman problem, Operations Research. Vol. 21, 1973
- [14] B. L. Fox, Integrating and accelerating tabu search, simulated annealing and genetic algorithm, Annals of Operations Research, Vol. 41, 1993
- [15] L. Ingber, Simulated annealing: Practice vs theory, Math. Computer modeling, Vol. 19, 1993
- [16] E. H. L. Aarts et al., A computational study of local search algorithms for job shop scheduling, ORSA J. on Computing. Vol. 6, 1994