

拟牛顿粒子群优化算法求解调度问题

丁知平

(清远职业技术学院, 广东 清远 511510)

摘 要: 针对资源受限的项目调度问题,将粒子群优化算法与拟牛顿优化算法相结合,提出了一种混合粒子群算法。本算法利用粒子群算法求得优化解,然后利用拟牛顿方法对所得到的解进行局部优化,以尽量达到或接近全局最优点。结果表明,本算法能够有效地求解大规模项目调度问题,具有较好的应用价值。

关键词: 拟牛顿方法; 粒子群优化算法; 项目调度问题

中图分类号: TP18; TP393

文献标志码: A

文章编号: 1001-3695(2012)01-0140-02

doi:10.3969/j.issn.1001-3695.2012.01.039

Quasi-Newton method particle swarm optimization algorithm for solving scheduling problem

DING Zhi-ping

(Qingyuan Polytechnics, Qingyuan Guangdong 511510, China)

Abstract: To solve the resource-constrained project scheduling problem, this paper proposed a novel hybrid particle swarm optimization algorithm based on quasi-Newton method. In the proposed algorithm, firstly, using particle swarm optimization algorithm to obtain the better solution, and then to obtain the optimal solution using quasi-Newton method, to try to achieve the global optimal solution or the approximate global optimal solution. The results of experiment verify the effectiveness of the proposed method. It is effective for solving the resource-constrained project scheduling problem, and has good application value.

Key words: quasi-Newton method; particle swarm optimization algorithm; project scheduling problem

在满足活动时序、资源约束等条件下安排各活动开始时间和结束时间,从而使工期最小化,被称为资源受限的项目调度问题(resource constrained project scheduling problem, RCPSP)。该问题被证明是 NP 难问题^[1]。对于小规模 RCPSP,可以采用整数规划、动态规划、分支定界法来求解。对于大型复杂的 RCPSP,常采用启发式算法求解,如遗传、蚁群、模拟退火、粒子群优化等算法^[2-5]。粒子群优化算法的优点是搜索速度快、效率高、算法简单,适合于实值型处理。但是,基本粒子群算法在求解该问题时,对于离散的优化问题处理不佳,容易陷入局部最优,导致优化结果与理想结果之间有一定差距。

拟牛顿法是求解非线性优化问题最有效的方法之一,于 20 世纪 50 年代由美国 Argonne 国家实验室的物理学家 Davidson^[6] 提出。拟牛顿法只要求每一步迭代时知道目标函数的梯度,通过测量梯度的变化,构造一个目标函数的模型,使之足以产生超线性收敛性。因为拟牛顿法不需要二阶导数的信息,所以有时比牛顿法(Newton's method)更为有效。但拟牛顿法对初始点比较敏感,而且求得的大多是局部最优解。鉴于此,本文把拟牛顿法和粒子群优化算法融合在一起,提出一种混合型粒子群优化算法。该混合算法先利用粒子群优化算法的群体搜索性在全变量空间内搜索问题的解,等进化到后期时,再利用拟牛顿法在收敛点附近细致地搜索最优解。

1 项目调度问题的模型

对于 RCPSP,有三个假设:

a) 只考虑 FS = 0 的逻辑关系,即所有紧前任务结束后,后续任务可以立刻开始,不存在延迟、反馈和循环;

b) 只考虑可更新资源,即资源在每个时刻的供应量是有限的,但并不随着项目进展而逐步消耗;

c) 每一项任务均不可中途暂停。

基于这三个假设,RCPSP 可描述为:一个项目包含 J 项任务,每项任务 $j(j=0,1,2,\dots,J)$ 可拆分,总耗时为 d_j 个时间单位,项目整个工期为 T ; P_j 表示任务 j 的紧前任务集合,任务 j 在其任一紧前任务 $p(p \in P_j)$ 完成前不能开始, S_j 表示任务 j 的紧后任务集合;项目共有 K 种可更新资源,第 k 种资源在任一时刻 t 的总量为 $R_k^t(k=1,2,\dots,K;t=0,1,2,\dots,T)$, r_{jk} 表示任务 j 所需的资源 k 的数量,在任何时刻,项目使用的各种资源不能超过其总量。RCPSP 的目标是使项目在资源均衡使用的前提下工期最短。引入两个决策变量:

$$x_{jt} = \begin{cases} 1 & \text{任务 } j \text{ 时刻 } t \text{ 正在进行} \\ 0 & \text{其他} \end{cases} \quad (1)$$

任务 j 的开始时间:

$$ST_j = \min_{1 \leq t \leq T} \{tx_{jt} | x_{jt} = 1\} \quad (2)$$

任务 j 的完成时间:

$$FT_j = \max_{1 \leq t \leq T} \{tx_{jt}\} \quad (3)$$

RCPSP 可以描述为

$$\min FT_j \quad (4)$$

$$\text{s. t. } \sum_{t=1}^T x_{jt} = d_j; j=0,1,2,\dots,J \quad (5)$$

$$FT_j - FT_p \geq d_j, \forall p \in P_j; j=0,1,2,\dots,J \quad (6)$$

$$\sum_{j=0}^J x_{jt} r_{jk} \leq R_k^t; k=1,2,\dots,K; t=0,1,2,\dots,T \quad (7)$$

其中:式(4)表示目标函数为项目总工期最小化;式(5)表示每

个任务拆分方式不影响总工期;式(6)表示任务紧前关系;式(7)表示资源约束。

2 求解项目调度问题的混合粒子群优化算法

2.1 拟牛顿算法的步骤

- a) 给出 $x_0 \in R^n, H_0 \in R^{n \times n}, 0 \leq \varepsilon < 1, k:=0$ 。
- b) 如果 $g_k \leq \varepsilon$, 则停止; 否则, 计算 $d_k = -H_k g_k$ 。
- c) 沿方向 d_k 作线性搜索求 $\alpha_k > 0$, 令 $x_{k+1} = x_k + \alpha_k d_k$ 。
- d) 校正 H_k 产生 H_{k+1} , 使得拟牛顿条件 $H_{k+1} y_k = s_k$ 成立。
- e) $k:=k+1$, 转 b)。

2.2 混合粒子群算法的步骤

- a) 初始化一群粒子(种群规模为 N), 包括随机位置和速度, 置 $k=0$ 。
- b) 计算每个粒子的适应度值。
- c) 对每个粒子将其适应度值与其历史最好位置 $pbest_i$ 的适应度值比较, 若较好, 则将其作为当前的最好位置 $pbest_i$ 。
- d) 对每个粒子将其适应度值与全局所经历的最好位置 $gbest$ 的适应度值比较, 若较好, 则将其作为当前的全局最好位置 $gbest$ 。
- e) 根据式(2)和(3)对粒子的速度和位置进行更新。
- f) 若达到终止条件(达到预设最大代数 M), 返回当前全局最优个体 $gbest$, 转向 g); 否则, $k=k+1$, 转 b)。
- g) 拟牛顿算法。以 f) 返回的当前全局最优个体 $gbest$ 为拟牛顿法的初始点, 进行迭代。

h) 若达到终止条件(达到最大代数或预设精度 ε), 则结束, 输出当前结果作为所求问题的最优解; 否则转 g) 继续拟牛顿算法。

2.3 粒子的表示

在 RCPSP 模型中, 将一个粒子代表一个可行解。每个粒子作为 J 维空间的一个点, 其 J 个元素代表所研究项目的 J 项任务。 $X_i = (x_{i1}, x_{i2}, \dots, x_{ij})$ 表示粒子 i 的 J 维位置, 它代表了 J 项任务的调度次序。任务 x_{ij} 在粒子转换为调度方案后, 其调度次序为 j 。如可行粒子 $X_1 = (0, 1, 2, 4, 3, 6, 5, 7, 8, 10, 9, 11, 12)$, 其中的每个数字的先后顺序就代表该项任务的调度次序。

2.4 调度方案的生成

项目调度生成方案有两种, 即串行和并行调度方案。串行调度方案产生积极计划, 并行调度方案产生非延迟计划^[8]。由于非延迟计划有可能错过最优解, 所以在采用启发式算法求解 RCPSP 时多采用串行调度方案。在 PSODE 算法中, 按照串行调度方案将粒子的可行解转换为可行的调度方案。定义如下符号来描述调度方案生成过程: S 为等待处理的任务集合; P 为已处理完的任务集合; W_t 为在时刻 t 正在处理的任务集合; ΔR_{kt} 为第 k 种资源在时间段 t 的剩余量; EST_j 为任务 j 的最早开始时间; EFT_j 为任务 j 的最早完成时间。

- 调度方案生成方法描述如下:
- a) 选择第 $i=1$ ($i=1, 2, \dots, N$) 个粒子初始化: $P=\emptyset, S=\emptyset, t=0$ 。
 - b) 更新 $S, S=\{j|j \notin P, P_j \subseteq P, j=0, 1, 2, \dots, J\}$ 。
 - c) 根据 P 中已调度任务完成时间 FT_m 和资源剩余量 $\Delta R_{kt} = R_k - \sum_{m \in X_i} \sum_{m=0}^J x_{mi} r_{mk} \ (m=j-1)$, 要求 $r_{jk} \leq \Delta R_{kt}$, 确定任务 j 的最早开始时间 EST_j 及最早结束时间 EFT_j 。
 - d) 更新 $P, P=P \cup \{j\}$, 判断 j 是否等于 J , 如果等于, 则转

- e); 否则, 令 $j=j+1$, 转 b)。
- e) 判断 i 是否等于种群规模数, 如不是, $i=i+1$, 转 a); 否则, 转 f)。
- f) 结束。

3 实验结果与分析

采用某型号工程机械装配项目为例, 其逻辑关系网络如图 1 所示, 各项任务参数如表 1 所示。

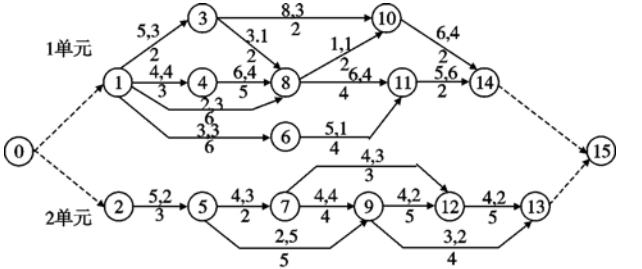


图1 算例逻辑关系网络

为了测试本文算法, 对本文算法及基本 PSO 算法进行了对比实验。两种算法的参数设置如下: 最大速度 v_{max} 和最小速度 v_{min} 分别为设置的上限和下限的一半, $w_{max}=0.9, w_{min}=0.4, c_1=c_2=2.0$ 。此外, 设定控制因子 $k=3$, 最大允许停滞次数为 $P=20$, 设定最大迭代次数为 1 500, 三种算法的种群规模均为 50, 每个实验设置运行 50 次。

表 1 各项任务参数

任务 编号	所需 资源 R_1	所需 资源 R_2	任务持 续时间	所属 单元	任务 编号	所需 资源 R_1	所需 资源 R_2	任务持 续时间	所属 单元
0-1	0	0	0	1	0-2	0	0	0	2
1-3	5	3	2	1	2-5	5	2	3	2
1-4	4	4	3	1	5-7	4	3	2	2
1-8	2	3	6	1	5-9	2	5	5	2
1-6	3	3	6	1	7-12	4	3	3	2
3-8	3	1	2	1	9-12	4	2	5	2
4-8	6	4	5	1	9-13	3	2	4	2
3-10	8	3	2	1	12-13	4	2	5	2
6-11	5	1	4	1					
8-10	1	1	2	1					
8-11	6	4	4	1					
10-14	6	4	2	1					
11-14	5	6	2	1					

优化后各项任务的开始时间如表 2 所示。

表 2 优化后各项目任务的开始时间

任务 编号	开始 时间	结束 时间	所属 单元	任务 编号	开始 时间	结束 时间	所属 单元
0-1	0	0	1	0-2	0	0	2
1-3	1	2	1	2-5	1	3	2
1-4	1	3	1	5-7	4	5	2
1-8	3	8	1	5-9	11	15	2
4-8	4	8	1	7-9	12	15	2
1-6	6	11	1	7-12	16	18	2
3-8	9	10	1	9-12	16	20	2
3-10	9	10	1	9-13	19	22	2
8-10	11	12	1	12-13	21	25	2
6-11	13	16	1	11-14	21	22	1
8-11	17	20	1	10-14	23	24	1

由于优化目标的差异, 本文对算法的评价方式作了符合本文模型的修改。算法所得解与最优解差距的最大值 d_{max} 、最小值 d_{min} 、平均值 d_{mean} , 以及达到最优解次数的百分比 w , 求解算法的平均计算时间 t_{mean} , 所得到的优化结果对比如表 3 所示。

2000000]之间均匀分布的随机数, S 取[2000,4500]间均匀分布的随机数。

免疫粒子群算法的参数设置为:最大迭代次数 $k_{\max}=1\,000$,种群大小 $N=100$,学习因子 $c_1=c_2=2$,惯性权重 $\omega=1$,惩罚因子参数 $a_1、a_2、a_3$ 分别为 2、1、0.5。

经过运算,算法得到的最优解,即每个广告在各类网页上印象次数如表 1 所示,表明了算法的有效性。

表 1 算法最优解

j	i																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	30	41	12	76	79	75	7	126	103	52	104	49	95	112	78	8	135	11	59	90
2	96	40	56	48	159	140	23	69	18	128	25	75	94	3	58	119	27	116	48	40
3	83	9	116	136	7	60	0	69	143	112	45	34	3	98	39	39	109	79	78	6
4	39	38	133	7	0	87	52	152	69	33	10	85	110	82	70	0	87	75	47	13
5	52	114	158	16	0	113	173	6	129	8	12	96	23	37	49	120	77	63	106	10
6	87	17	79	40	76	108	76	82	37	66	66	14	10	91	171	47	77	40	107	46
7	137	46	34	171	168	87	156	103	13	3	69	119	101	118	89	98	93	8	163	88
8	58	60	71	89	0	48	8	59	109	8	59	66	73	55	13	132	2	39	5	119
9	90	93	76	6	82	85	184	0	49	62	75	69	34	135	76	39	78	3	99	89
10	8	53	108	26	28	79	39	78	13	47	48	109	49	124	0	67	97	29	55	12

从表 1 中可以看到,有的印象次数比较少,说明广告和此类网页的相关性很小,所以点击率偏低。因此,广告在这类网页上投放的次数相对就要少些,否则无法得到很好的广告效果。反之,如果相关性很高,则点击率也就很高,那么在这类网页上可以相对多投放一些广告。

为了说明本文算法的效果,将本文算法与文献[8]算法进行了对比分析,得到的最终优化结果(网站最大化收益)如图 1 所示。从图 1 的结果可知,本文算法由于设计了适合问题表示的相关算子,加快了收敛速度,求解效率较高;本文算法得到的最优结果优于文献[8],说明本文算法采取的各种策略是有效的,可以得到更好的优化结果。算法运行结束,本文算法得到的网站最大收入为 438.8 万元,高于文献[8]算法的 422.5 万元。

4 结束语

本文提出了一种基于混合定价模型的网络广告资源优化模型,平衡了网站广告客户和网站运营商的利益,更适合实际应用情况。此模型是一个约束优化问题,通过罚函数将其转换

为无约束问题,并通过改进粒子群算法进行求解。仿真实验结果表明了算法的有效性。

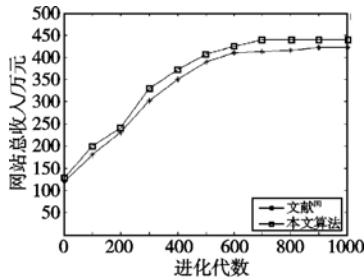


图1 相关算法性能比较

参考文献:

[1] LANGHEINRICH M, NAKAMURA A, ABE N, et al. Unintrusive customization techniques for web advertising[J]. Computer Networks, 2003, 31(11-16): 1259-1272.

[2] NAKAMURA A, ABE N. Improvements to the linear programming based scheduling of Web advertisements[J]. Electronic Commerce Research, 2008, 5(1): 75-98.

[3] BOSKAMP V, KNOOPS A, FRASINCAR F, et al. Maximizing revenue with allocation of multiple advertisements on a Web banner[J]. Computers and Operations Research, 2011, 38(10): 1412-1424.

[4] 齐洁, 汪定伟. 求解网络广告资源优化模型的改进微粒群算法[J]. 控制与决策, 2004, 19(8): 881-884.

[5] 李震. 网络广告效果的综合评价研究[J]. 科技管理研究, 2010, 12(8): 321-325.

[6] 王建冬. 网络广告界面评价中的人机交互理论——网络广告交互界面与广告效果关系模型[J]. 现代图书情报技术, 2009, 3(10): 301-304.

[7] 张志华, 王莉. 网络环境下广告资源优化决策模型[J]. 鞍山科技大学学报, 2006, 31(2): 321-315.

[8] ZHAO Wen-dan, WANG Ding-wei. Modeling and algorithm of Web advertising resources optimization[C]//Proc of the 29th Chinese Control Conference. 2010: 5590-5594.

[9] 刘朝华, 张英杰, 章兢. 一种双态免疫微粒群算法[J]. 控制理论与应用, 2011, 28(1): 65-72.

[10] 刘朝华, 张英杰, 章兢, 等. 基于免疫双态微粒群的混沌系统自抗扰控制[J]. 物理学报, 2011, 60(1): 019501.

(上接第 141 页) 表 3 两种算法计算结果比较

算法	$d_{\min}$	$d_{\max}$	d_{mean}	$w/\%$	t_{mean}
本文算法	0	1	1.00	96	3.87
基本 PSO	0	5	2.25	84	3.63

从表 3 可以看出,本文算法所得到调度方案的精确度比基本 PSO 算法方案高。由于本文算法在利用粒子群算法求得优化解后,还要利用拟牛顿方法进行局部优化,以尽量达到或接近全局最优点,所以其运算时间比基本 PSO 算法要稍长。总体来说,本文算法在缩短项目工期、合理调配项目资源上有比较好的效果,其求解质量要优于基本 PSO 算法。

4 结束语

本文针对以项目工期最小为目标的资源受限项目调度问题,设计了一种混合粒子群算法,并与基本 PSO 算法在解决资源受限项目调度问题时的精确度和效率进行了比较。计算测试表明,本文算法解决资源约束项目调度问题相对于基本 PSO 算法有较大优势,这说明本文方案有效。

参考文献:

[1] KOLISCH R, HARTMANN S. Experimental investigation of heuristics for resource-constrained project scheduling: an update [J]. European Journal of Operational Research, 2006, 174(1): 23-37.

[2] 杜焱, 彭武良. 一种求解资源受限项目调度问题的遗传算法[J]. 沈阳理工大学学报, 2009, 28(1): 70-74.

[3] 郑超, 高连生. 蚁群算法在资源受限项目调度问题中的应用[J]. 计算机工程与应用, 2005, 41(27): 205-208.

[4] 喻小光, 战德臣, 聂兰顺. 应用遗传模拟退火算法实现资源受限项目调度[J]. 计算机工程与应用, 2009, 45(24): 17-20.

[5] 王巍, 赵国杰. 粒子群优化在资源受限工程调度问题中的应用[J]. 哈尔滨工业大学学报, 2007, 39(4): 669-772.

[6] 袁亚湘, 孙文瑜. 最优化理论与方法[M]. 北京: 科学出版社, 2009.

[7] KOLISCH R. Serial and parallel resource-constrained project scheduling methods revisited: theory and computation [J]. European Journal of Operational Research, 2009, 99(2): 320-333.