文章编号: 1000−8152(2011)08−1113−08

求解资源受限项目调度问题的约束规划/数学规划混合算法

刘士新[1], 宋健海[2]

(1. 东北大学 信息科学与工程学院 流程工业综合自动化国家重点实验室, 辽宁 沈阳 110819;

2. 上海宝信软件股份有限公司, 上海 201900)

摘要: 利用约束规划(constraint programming, CP)与数学规划(mathematical programming, MP)结合的方法求解调度问题已经获得了一些较好的研究成果, 正成为调度问题研究领域的一个新的热点研究方向. 本文针对求解资源受限项目调度问题(RCPSP)的整数规划模型, 设计了基于CP技术的问题和模型预处理方法, 证明了整数规划模型的有效不等式定理, 提出了通过将项目子网络图转化为加权最大团问题求解后获得有效不等式的方法. 引用标准问题库PSPLIB中的一组典型问题进行求解实验, 结果表明本文提出的有效不等式可以明显改进模型的求解质量和时间性能. 论文最后对实验结果进行了深入讨论, 讨论了未来的研究方向.

关键词: 项目调度; 资源受限; 整数规划; 约束规划; 有效不等式; 最大团问题

中图分类号: C934 **文献标识码**: A

Combination of constraint programming and mathematical programming for solving resources-constrained project-scheduling problems

LIU Shi-xin[1], SONG Jian-hai[2]

(1. School of Information Sciences & Engineering, Northeastern University,
State Key Laboratory of Synthetical Automation for Process Industries, Shenyang Liaoning 110819, China;
2. Shanghai Baosight Software Limited Company, Shanghai 201900, China)

Abstract: Combining constraint programming(CP) and mathematical programming(MP) to solve scheduling problems has been an interesting topic for researchers, and promising results are obtained. We propose a preprocessing approach for solving resource-constrained project-scheduling problems(RCPSP) with integer programming(IP) model, and prove an effective inequality theory for the IP model. The effective inequality can be obtained by solving a maximum clique problem which is built on a sub-network of the original project. A detailed computational experiment is performed using the well-known standard instances in PSPLIB. Computational results show that the proposed effective inequality remarkably improves the performances of the IP model. Finally, the computational results are analyzed and future research directions are discussed.

Key words: project scheduling; resource-constrained; integer programming; constraint programming; effective inequality; maximum clique problem

1 引言(Introduction)

资源受限项目调度问题(resource-constrained project scheduling problem, RCPSP)在实践中具有广泛的应用. 在理论上, RCPSP模型丰富, 而且多属NP-hard问题, 求解困难, 几十年来一直受到国内外众多学者的研究与关注[1]. 早期的研究方法主要是采用整数规划(integer programming, IP)和分枝定界等精确算法[2～11]. 根据作者掌握的资料, 文献[6～10]中介绍的分枝定界算法是目前最好的精确算法, 可以在不设求解时间限制的条件下求出RCPSP标准问题库PSPLIB中J30一组测试问题中全部480个实例的最优解. 精确算法的缺点是只能求解小规模问题, 对实际应用中的大规模问题无能为力. 近年来随着遗传算法、禁忌搜索等元启发式算法的兴起, 元启发式算法被越来越多地应用在RCPSP的求解中[12], 具有代表性的算法有Hartmann提出的基于工作链表编码方式的自适应遗传算法[13]、Merkle等人提出的蚁群算法[14]、Debels等人提出的分散搜索和电磁学算

收稿日期: 2010−02−05; 收修改稿日期: 2010−10−09.
基金项目: 国家自然科学基金资助项目(71021061, 70771020); “863”计划/先进制造技术领域专题资助项目(2007AA04Z194); 中央高校基础科研业务费资助项目(N100504001).

法混合算法[15]、Debels和Vanhoucke提出的基于分解技术的遗传算法[16]、Valls等人提出的双向求解的模拟退火算法[17]、Gonçalves等人提出的求解多项目调度问题的遗传算法[18]、Ballestín等人提出的求解允许工作占先的RCPSP的启发式算法[19]、Ranjbar等人提出的分散搜索和路径重链混合算法[20].

近年来, 起源于人工智能研究领域的约束规划(constraint programming, CP)技术在调度问题研究领域获得了越来越多的研究和关注[21~27], 在RCPSP研究领域也获得了比较好的研究成果[25~27]. 虽然目前基于约束规划的算法的求解效果还不如最好的元启发式算法的求解效果, 但作为一类较新的研究方法仍值得学者的研究和关注. 特别是在CP与数学规划结合的研究方面, 已经在网络设计、车辆路径优化、生产计划/调度等多种组合优化问题中获得了很好的应用效果[28~31]. 2002年, 运筹学和管理学研究协会(INFORMS)旗下的著名学术期刊《INFORMS Journal on Computing》出版了专辑"Special Issue on the Merging of Mathematical Programming and Constraint Programming"介绍数学规划和CP结合的研究成果[31].

本文基于Alvarez-Valdés和Tamarit[5]针对RCPSP建立的IP模型, 结合CP中的约束传播技术, 设计了问题和模型的预处理方法, 证明了IP模型的有效不等式定理, 提出了通过将项目子网络图转化为加权最大团问题求解后获得有效不等式的方法. 实验结果表明经过预处理并添加了有效不等式后的模型求解效果明显优于原始模型, 在设定求解时间为30分钟的情况下, 可以求得J30一组测试问题中480个实例中的462个实例的最优解, 与最优解的标准差平均值只有0.071%.

2 问题描述及数学模型(Problem description and mathematical model)

RCPSP可以由一张有向图$G=(V,H)$表示, 其中$V=\{1,2,\cdots,J\}$为节点(工作)的集合, H为工作间优先关系集合, 有向弧$(i,j)\in H$代表工作i直接优先于工作j执行. 工作1是唯一最早开始的工作, 工作J是唯一最晚完成的工作, 分别代表整个项目的开始和结束. 工作$i(i=1,\cdots,J)$的最早开始时间为r_i, 执行时间为p_i, 最迟完成时间为d_i, 在执行的每个时间段内需要第$k(k=1,\cdots,K)$种资源量为r_{ik}. 第k种资源在项目工期内每个阶段的可用量为R_k. 问题是在满足各工作间优先关系和资源限制的条件下如何安排各项工作的开始时间, 使得完成整个项目的时间最小.

定义 1[32] 禁止集是一个工作的集合Ω, 满足对于$\forall i,j\in\Omega$, 工作i,j之间不存在优先关系(或传递优先关系)约束, 但由于资源约束, Ω中的工作不能同时进行. 如果Ω不存在严格子集为禁止集, 则称Ω为最小禁止集(minimal forbidden set, MFS).

根据定义1有, 对于任意最小禁止集Ω, $\exists k\in\{1,\cdots,K\}$满足$\sum\limits_{j\in\Omega} r_{jk} > R_k$, 同时$\forall k\in\{1,\cdots,K\}$, $\forall i\in\Omega$满足$\sum\limits_{j\in\Omega\backslash\{i\}} r_{jk}\leqslant R_k$. 显然, 对于任何最小禁止集, 只要在$\Omega$中的任何两个工作之间添加一个优先关系约束, 就可以化解Ω中工作间的资源冲突. 如果能够化解一个项目中的所有最小禁止集, 则可以应用时间约束网络计算方法获得一个RCPSP可行调度计划. 文献[33]在理论上证明了这一结论.

引入以下符号定义:

$$x_{ij}=\begin{cases}1, \text{如果工作}i\text{优先(传递优先)于工作}j,\\ 0, \text{其他}.\end{cases}$$

t_i为工作i的开始时间.

H^*为工作间优先(包括传递优先)关系集合, 即如果工作i优先(或传递优先)于工作j, 则$(i,j)\in H^*$; Ω_p为第$p(p=1,\cdots,P)$个仅包含2项工作的最小禁止集; Ω_n为第$n(n=1,\cdots,N)$个包含3项或3项以上工作的最小禁止集; M为足够大的自然数. Alvarez-Valdés和Tamarit[5]针对RCPSP建立了如下数学模型:

$$\min t_J, \tag{1}$$
$$\text{s.t.}: \ x_{ij}=1, x_{ji}=0, \forall (i,j)\in H^*, \tag{2}$$
$$0\leqslant x_{ij}+x_{ji}\leqslant 1, \forall i,j\in V, \tag{3}$$
$$x_{ik}\geqslant x_{ij}+x_{jk}-1, \forall i,j,k\in V, \tag{4}$$
$$\mathbf{P}: x_{ij}+x_{ji}=1, i,j\in\Omega_p, p=1,\cdots,P, \tag{5}$$
$$\sum_{i,j\in\Omega_n} x_{ij}\geqslant 1, n=1,2,\cdots,N, \tag{6}$$
$$t_j\geqslant t_i+(p_i+M)x_{ij}-M, \forall i,j\in V, \tag{7}$$
$$x_{ij}\in\{0,1\}, \forall i,j\in V, \tag{8}$$
$$r_i\leqslant t_i\leqslant d_i-p_i, \forall i\in V. \tag{9}$$

其中: 目标函数式(1)代表项目完成时间最小化. 约束式(2)代表工作间优先关系约束; 约束式(3)保证任意2项工作i和j间最多有1种优先关系; 约束式(4)为传递优先关系约束; 约束式(3)和(4)联合保证有向图G中无环; 约束式(5)保证仅包含2项工作的最小禁止集中的工作间一定存在优先关系; 约束式(6)保证包含3项或3项以上工作的最小禁止集中的工作间至少在2项工作间存在优先关系; 约束式(7)保证变量t_i, t_j和x_{ij}间的正确关系; 约束式(8)和(9)代表变量的取值范围.

在文献[5]中, Alvarez-Valdés和Tamarit针对问题$\mathbf{P}$的维数、多面体特性等问题进行了理论分析, 提出了

一种生成模型有效不等式的方法: 设$(i,j)\in H$, 定义$S(i)=\{k\in V|(i,k)\in H^*,(j,k)\notin H^*, P(j)=\{k\in V|(k,j)\in H^*,(k,i)\notin H^*, P(i,j)=\{k\in V|(k,i)$或$(k,j)$或两者均为最小禁止集, $k\notin P(j)\cup S(i)\}$.

定理 1[5]　记$L=S(i)\cup P(j)\cup P(i,j), L'\subseteq L$满足$\forall i,j\in L'$不能够同时执行, 则不等式

$$t_j\geqslant t_i+p_i+\sum_{k\in L'\cap S(i)}p_kx_{kj}+\sum_{k\in L'\cap P(j)}p_kx_{ik}+\sum_{k\in L'\cap P(i,j)}p_k(x_{ik}+x_{kj}-1)\tag{10}$$

为问题$\mathbf{P}$的有效不等式.

文献[5]举例说明了有效不等式(10)的使用方法, 但没有给出该有效不等式在求解RCPSP时对模型的改进效果.

3　约束规划与有效不等式(Constraint programming and valid inequation)

3.1　基于约束传播的问题及模型预处理(Constraint propagation based preprocessing of problem and model)

约束传播是CP的关键技术之一. 文献[34]比较了累积资源调度问题研究领域常用的约束传播算法并提出了一种新的约束传播算法. 这些算法均可以在RCPSP的求解过程中获得应用. 记UB为RCPSP问题实例可行解目标函数值的上界, r'_i和d'_i分别为应用约束传播算法调整后工作i的最早开始时间和最晚完成时间, 则RCPSP存在以下性质:

性质 1　如果工作对$(i,j)\notin H^*$满足$r'_i\geqslant d'_j$, 则在任何目标函数值小于等于UB的可行解中工作j优先于工作i执行.

证　由假设条件可知工作j的最晚完成时间d'_j小于等于工作i的最早开始时间. 因此, 在任何可行解中必有工作j优先于工作i执行.　　证毕.

性质 2　如果工作i和j不能并行且$r'_i+p_i+p_j>d'_j$, 则在任何目标函数值小于等于UB的可行解中工作j优先于工作i执行.

证　如果工作i和j不能并行, 则在问题$\mathbf{P}$的可行解中, 或者$x_{ij}=1$, 或者$x_{ji}=1$. 假设在某个解中工作i优先于工作j执行, 即决策变量$x_{ij}=1$, 由性质2的假设条件可知$r'_j+p_j\geqslant r'_i+p_i+p_j>d'_j$, 即工作$j$的完成时间大于其最晚完成时间$d'_j$. 因此, 该解不可行, 只能有决策变量$x_{ji}=1$, 即在任何目标函数值小于等于UB的可行解中工作$j$优先于工作$i$执行.　　证毕.

性质 3　如果工作i和j不能并行且在项目网络图中添加优先关系弧(i,j)后网络图的关键路径长度大于UB, 则在任何目标函数值小于等于UB的可行解中工作j优先于工作i执行.

证　如果工作i和j不能并行, 则在问题$\mathbf{P}$的可行解中或者$x_{ij}=1$, 或者$x_{ji}=1$. 假设在某个解中工作i优先于工作j执行, 即决策变量$x_{ij}=1$, 由性质3的假设条件可知网络图的关键路径长度大于UB, 因为关键路径长度是可行解的下界. 因此, $x_{ij}=1$不满足上界条件, 只能有决策变量$x_{ji}=1$, 即在任何目标函数值小于等于UB的可行解中工作j优先于工作i执行.　　证毕.

性质 4　如果工作i和j不能并行且在项目网络图中添加优先关系弧(i,j)后问题不存在能量可行解[34], 则在任何目标函数值小于等于UB的可行解中工作j优先于工作i执行.

证　如果工作i和j不能并行, 则在问题$\mathbf{P}$的可行解中或者$x_{ij}=1$, 或者$x_{ji}=1$. 假设在某个解中工作i优先于工作j执行, 即决策变量$x_{ij}=1$, 由性质4的假设条件可知在目标函数值小于等于UB的条件限制下该解不可行. 因此, 只能有决策变量$x_{ji}=1$, 即在任何目标函数值小于等于UB的可行解中工作j优先于工作i执行.　　证毕.

记文献[34]中的混合约束传播算法为HCPA, 本文基于性质1～4设计了以下问题预处理过程:

过程1: 基于约束传播的问题预处理.

1) 应用HCPA算法调整各项工作的r_i和d_i值, 记调整后的值为r'_i和d'_i.

2) 对$\forall i,j\in V$满足性质1的假设条件, 在网络图中添加优先关系弧(j,i).

3) 生成问题实例中仅包含2项工作的全部最小禁止集, 记为$\Omega_p, p=1,\cdots,P$.

4) 对于每一个最小禁止集$\Omega_p, p=1,\cdots,P$, 如果工作$i,j\in\Omega_p$满足性质2～4中的任何假设条件, 在网络图中添加优先关系弧(j,i).

5) 如果在2)～4)执行过程中在网络图中添加了新的优先关系弧(j,i), 转到1).

6) 输出新的项目网络图和各工作的r'_i, d'_i值.

经过过程1对问题进行预处理后, 可以建立问题$\mathbf{P}$模型. 在求解模型前, 应用以下过程对模型进行预处理:

过程2: 基于约束传播的模型预处理.

1) 根据过程1的输出确定变量$t_1,t_2,\cdots,t_n$的上下界, $r'_i\leqslant t_i\leqslant d'_i-p_i, i=1,2,\cdots,n$.

2) 预处理变量x_{ij}. 对$\forall(i,j)\notin H^*$.

① 如果$r'_i+p_i+p_j>d'_j$, 则可直接确定变量值$x_{ij}=0$;

② 如果在项目网络图中添加优先关系弧(i,j)后

网络图的关键路径长度大于UB, 则可直接确定变量值$x_{ij}=0$;

③ 如果在项目网络图中添加优先关系弧(i,j)后问题不存在能量可行解, 则可直接确定变量值$x_{ij}=0$.

3) 输出新的模型.

分别应用过程1和2对问题及模型进行预处理, 可以起到两方面的作用: ① 直接确定一些最小禁止集中工作的优先顺序, 从而减少约束式(5)对应的约束数量; ② 确定一些变量的取值, 降低模型的规模.

3.2 基于约束传播的有效不等式(Constraint propagation based valid inequation)

Alvarez-Valdés和Tamarit提出的有效不等式定理仅可以应用在$(i,j)\in H$的情况. 以下约定$(i,i)\in H^*$, 本文基于约束传播技术提出如下定理:

定理 2 对$\forall(i,j)\in H^*$, 记$\Omega_{ij}=\{k|(i,k)\in H^*,(k,j)\in H^*,k\neq j\}$, $\Omega_{ij}^p\subseteq\Omega_{ij}$满足$\Omega_{ij}^p$中的任何2项工作都不能同时执行, 则

$$t_j-t_i\geqslant\max\{\sum_{l\in\Omega_{ij}^p}p_l,\max_{k\in K}(\sum_{l\in\Omega_{ij}}\frac{p_lr_{lk}}{R_k})\}.\tag{11}$$

证 分两阶段证明.

1) $t_j-t_i\geqslant\sum\limits_{l\in\Omega_{ij}^p}p_l$. 由于$\Omega_{ij}^p$中的任何2项工作都不能同时执行, 因此, 完成$\Omega_{ij}^p$中全部工作的时间为$\sum\limits_{l\in\Omega_{ij}^p}p_l$. 又$\Omega_{ij}^p\subseteq\Omega_{ij}$, 因此, 在任何可行计划中完成$\Omega_{ij}$中全部工作的时间大于等于$\sum\limits_{l\in\Omega_{ij}^p}p_l$. 于是$t_j-t_i\geqslant\sum\limits_{l\in\Omega_{ij}^p}p_l$.

2) $t_j-t_i\geqslant\max\limits_{k\in K}(\sum\limits_{l\in\Omega_{ij}}\frac{p_lr_{lk}}{R_k})$. 完成$\Omega_{ij}$中全部工作需要第$k$种资源的总消耗量为$\sum\limits_{l\in\Omega_{ij}}p_lr_{lk}$. 而第$k$种资源单位时间供给量为$R_k$, 因此, 完成$\Omega_{ij}$中全部工作需要的时间大于等于$\sum\limits_{l\in\Omega_{ij}}\frac{p_lr_{lk}}{R_k}$. 考虑全部资源种类则有$t_j-t_i\geqslant\max\limits_{k\in K}(\sum\limits_{l\in\Omega_{ij}}\frac{p_lr_{lk}}{R_k})$. 证毕.

综合考虑1)和2)有

$$t_j-t_i\geqslant\max\{\sum_{l\in\Omega_{ij}^p}p_l,\max_{k\in K}(\sum_{l\in\Omega_{ij}}\frac{p_lr_{lk}}{R_k})\}.$$

每个Ω_{ij}可以有多个Ω_{ij}^p, 应用定理2时需要计算具有最大$\sum\limits_{l\in\Omega_{ij}^p}p_l$值的一个, 该集合可以通过求解$\Omega_{ij}$的最大团问题(maximum clique problem, MCP)得到. 虽然MCP已经被证明属于NP-hard问题, 但由于在RCPSP中Ω_{ij}的规模往往很小, 因此, 仍可以采用数学规划的方法求解. 记$G_{ij}=(\Omega_{ij},H_{ij})$为$G=(V,H)$的子图, $G'_{ij}=(\Omega_{ij},E)$为基于G_{ij}生成的无向图, $E=\{(k,l)|k,l\in\Omega_{ij},(k,l)\in H^*$或$(l,k)\in H^*$或$\{l,k\}\in\{\Omega_p|p=1,\cdots,P\}\}$. 引入如下决策变量:

$$y_k=\begin{cases}1,\ k\in\Omega_{ij}^p,\\0,\ 否则,\end{cases}$$

则本文应用文献[35]中的MCP数学模型求解Ω_{ij}^p, 模型如下:

$$\max\sum_{k\in\Omega_{ij}}p_ky_k,\tag{12}$$

$$\text{MCP: s.t. } y_k+y_l\leqslant1,\forall(k,l)\in\bar{E},\tag{13}$$

$$y_k\in\{0,1\},\forall k\in\Omega_{ij}.\tag{14}$$

其中: p_k为工作k的执行时间, $\bar{E}$为E的补集. 由MCP的定义可知, 式(12)中的数据项$\sum\limits_{l\in\Omega_{ij}^p}p_l$等于问题MCP的最优目标函数值.

3.3 算法总体流程(General flow of the algorithm)

综合第3.1节和3.2节的预处理过程及有效不等式定理, 本文设计了求解RCPSP的算法, 算法主流程如过程3所示.

过程3: 算法总体流程.

1) 应用过程1对问题实例预处理, 得到新的项目网络图和各工作最早开始/最晚完成时间值.

2) 根据1)输出的网络图和各工作最早开始/最晚完成时间建立模型**P**.

3) 应用过程2对模型**P**的变量进行预处理, 约减问题解空间.

4) 向3)输出的模型中添加有效不等式,

For $i=1$ to J

For $j=1$ to J

If $(i,j)\in H^*$

Then 生成有效不等式加到模型**P**中.

5) 求解添加了有效不等式的模型**P**.

6) 输出模型解.

在实现步骤4)向模型添加有效不等式时, 可以根据实验需求分别应用定理1和定理2添加不同的不等式, 也可以两种不等式都添加或者都不添加. 具体实验结果参见第4节.

4 实验结果(Experimental results)

4.1 实验设计(Experiment design)

为了测试本文提出的问题和模型预处理过程以及有效不等式在求解RCPSP中的作用, 引用标准问题库PSPLIB[36]中J30一组问题进行实验. 测试实例根据项目网络复杂性系数(network complexity, NC)、资源因数系数(resource factor, RF)、资源

强度系数(resource strength, RS)的不同组合随机生成. 其中NC反应项目网络图中各工作间优先关系约束的松紧; RF反映工作对资源需要的耦合程度, RF= 1代表每项工作需要全部种类的资源; RS反映资源的限制强度, RS值越小表示资源限制越紧.

每个问题实例包含32项工作, 需要4种可更新资源($K = 4$), 主要参数水平如表1所示. 对参数NC, RF和RS的48种不同组合, 每种组合随机生成10个问题实例, 因此, 共有$3 \times 4 \times 4 \times 10 = 480$个问题实例.

表 1 测试问题各种参数的组合

Table 1 Parameter settings of the benchmark problems

NC	RF	RS
{1.5, 1.8, 2.1}	{0.25, 0.5, 0.75, 1.0}	{0.2, 0.5, 0.7, 1.0}

实验中首先应用文献[37]描述的遗传局域搜索算法求解每个测试问题得到初始解, 将问题**P**的目标函数值上界UB设定为初始解的目标函数值. 记问题及模型预处理过程为CPP, 定理1提出的有效不等式为AT-Cuts, 定理2提出的有效不等式为New-Cuts. 在实验中对原模型**P**进行以下处理: I) 添加AT-Cuts; II) 应用CPP预处理后添加AT-Cuts; III) 应用CPP预处理后添加New-Cuts; IV) 应用CPP预处理后添加AT-Cuts和New-Cuts. 将经过以上4种方式处理后的模型应用CPLEX 11.2分别进行求解, 每次求解时间限制设定为1800 s.

4.2 计算结果与分析(Computational results and analysis)

实验统计结果总体情况如表2所示. 表2中数据项Δopt(%)代表目标函数值与最优解目标函数值的平均标准差; avg_CPU代表求解每个问题实例的平均CPU时间(s). 由表2可见: 1) CPP预处理过程可以改进解的质量和算法效率; 2) 采用第III种方式处理模型时算法在求解质量和时间性能方面均表现出较大优势; 3) 采用第IV种方式处理模型时算法性能不如采用第III种方式, 但稍好于采用第II种方式. 笔者无法提供第3)种实验统计结果存在的理论依据. 推测原因是因为同时添加两种有效不等式会导致有些测试问题出现过多的退化解, 使得问题变得更加困难.

表 2 总体实验结果

Table 2 The summary of experimental results

AT-Cuts		CPP+AT-Cuts		CPP+New-Cuts		CPP+AT-Cuts+New-Cuts	
Δopt/%	avg_CPU/s	Δopt/%	avg_CPU/s	Δopt/%	avg_CPU/s	Δopt/%	avg_CPU/s
0.135	141.63	0.119	128.60	0.071	108.43	0.113	121.99

为了研究问题参数NC, RF和RS对各种处理方式下算法性能的影响, 作者对实验结果按照NC, RF和RS的不同设置进行统计, 统计结果如表3所示. 由表3可见: 问题参数NC, RF和RS对算法性能影响显著, 且对4种处理方式下算法性能的影响体现出相同的规律, 即算法性能随着网络复杂性系数NC的增大而改善, 随着资源因数系数RF的增大而变坏, 随着资源强度系数RS的增大而改善.

表 3 问题参数NC, RF和RS对算法性能的影响

Table 3 The influence of NC, RF and RS on the performance of algorithms

参数		AT-Cuts		CPP+AT-Cuts		CPP+New-Cuts		CPP+AT-Cuts+New-Cuts	
		Δopt/%	avg_CPU/s	Δopt/%	avg_CPU/s	Δopt/%	avg_CPU/s	Δopt/%	avg_CPU/s
NC	1.5	0.255	268.77	0.221	213.71	0.145	208.02	0.218	209.83
	1.8	0.151	139.24	0.135	160.46	0.069	113.49	0.121	138.83
	2.1	0.000	16.88	0.000	11.62	0.000	3.78	0.000	17.32
RF	0.25	0.000	0.07	0.000	0.070	0.000	0.07	0.000	0.08
	0.50	0.000	22.59	0.000	17.76	0.000	25.45	0.000	15.17
	0.75	0.195	213.56	0.207	195.58	0.058	147.13	0.135	176.35
	1.00	0.345	330.30	0.268	300.99	0.227	261.05	0.317	296.36
RS	0.2	0.454	441.69	0.384	393.26	0.241	353.15	0.394	391.45
	0.5	0.086	111.53	0.091	106.56	0.045	75.82	0.058	91.39
	0.7	0.000	13.29	0.000	14.58	0.000	4.73	0.000	5.12
	1.0	0.000	0.01	0.000	0.07	0.000	0.01	0.000	0.01

表3统计结果表明, 4种处理方式下的算法针对问题参数值NC = 2.1, RF = $\{0.25, 0.5\}$以及RS = {0.7, 1.0}的测试实例均可以在1800 s内全部获得最优解. 统计全部480个测试实例的结果隐藏了各种算法的性能差异. 因此, 作者进一步统计4种处理方式下算法求解480个测试实例中的30个困难问题的情况(困难问题的选择标准是存在某种处理方式的算法1800 s内没求得最优解), 进而比较不同处理方式下算法的性能差异. 统计结果如表4所示.

表4中Para和Inst为测试实例编号, 例如Para = 9, Inst = 4代表测试实例J309_4.SM; Δopt(%)代表相应算法得到的目标函数值与最优解目标函数值的标准差; CPU代表求解测试问题使用的CPU时间(s); Nodes代表算法终止时CPLEX搜索过的节点数量; opt行代表每种处理方式下算法求得的最优解的数量; average行代表每种处理方式下算法输出解与最优解的标准差平均值; sum行代表每种处理方式下的算法求解30个难题的总运行时间以及CPLEX搜索的节点总数; nodes/cpu行代表每种处理方式下算法在每秒钟内CPLEX搜索的节点数.

表 4 算法求解困难问题时的性能对比

Table 4 The comparison of algorithms performance for solving hard problems

		CPP+AT-Cuts			CPP+New-Cuts			CPP+AT-Cuts+New-Cuts		
Para	Inst	Δopt/%	CPU/s	Nodes/个	Δopt/%	CPU/s	Nodes/个	Δopt/%	CPU/s	Nodes/个
9	4	1.408	1800	3842	1.408	1800	7266	1.408	1800	3087
9	6	5.084	1800	4842	0.000	1485	8225	0.000	1122	3281
9	7	3.174	1800	3607	3.174	1800	11610	3.174	1800	4408
9	10	1.136	1800	3751	0.000	621	5459	1.136	1800	3071
10	3	1.612	1800	31749	0.000	824	19154	1.612	1800	33177
10	10	2.439	1800	10563	0.000	411	2481	0.000	616	3521
13	1	5.172	1800	2981	3.448	1800	6161	5.172	1800	3492
13	3	1.316	1800	2859	2.631	1800	10235	3.947	1800	2786
13	4	2.777	1800	6266	1.388	1800	20660	1.388	1800	5323
13	5	1.492	1800	2517	1.492	1800	7157	2.985	1800	3805
13	6	1.562	1800	2688	1.562	1800	4172	3.125	1800	5436
13	7	1.298	1800	3728	1.298	1800	16521	1.298	1800	3580
13	8	0.000	1198	2828	0.000	855	27855	2.830	1800	4193
13	9	1.408	1800	9708	1.408	1800	16636	1.408	1800	9011
13	10	1.562	1800	2652	1.562	1800	7597	1.562	1800	2964
14	3	1.724	1800	30924	1.724	1800	44036	1.724	1800	25893
14	5	0.000	749	5974	0.000	791	10155	0.000	524	4263
14	9	2.173	1800	6269	2.173	1800	8170	2.173	1800	4897
25	1	2.150	1800	8061	1.075	1800	77765	2.150	1800	12343
25	3	2.631	1800	4958	1.315	1800	18010	2.631	1800	5190
25	4	1.234	1800	10453	0.000	1168	176180	1.234	1800	8913
25	7	1.052	1800	16807	0.000	124	25461	0.000	968	8901
25	8	2.898	1800	5437	0.000	1163	22275	2.898	1800	4996
29	1	1.176	1800	6191	3.529	1800	41154	2.352	1800	5976
29	2	2.222	1800	5371	1.111	1800	20527	2.222	1800	5277
29	5	2.040	1800	7248	0.000	1635	55496	1.020	1800	6600
29	8	1.250	1800	5140	2.500	1800	10371	1.250	968	7558
29	9	2.061	1800	7692	0.000	1337	51641	2.061	1800	7022
30	2	2.941	1800	40796	1.470	1800	36089	1.470	1800	46243
45	9	0.000	1233	21161	0.000	113	23954	0.000	1070	17730
opt		3			12			5		
average		1.900			1.142			1.808		
sum		51780	277063		42927	792473		48468	262937	
nodes/cpu		5.351			18.461			5.425		

表4中的统计数据更能体现不同处理方式下算法的性能差异. 表4中的数据表明: 1) 采用第III种处理方式时算法针对30个困难问题获得解的质量最好, 在1800 s内有12个问题求得了最优解, 与最优解的平均标准差只有1.142%; 采用第II、IV种处理方式时算法对应的统计值分别为3和1.900%, 5和1.808%. 2) 3种处理方式下CPLEX优化软件在单位时间内搜索的节点数量有很大的差异, 其中采用第III种处理方式时算法在单位时间内搜索的节点数量最大, 为每秒平均18.461个节点, 这也可以解释该处理方式下算法求解效果最好的实验结果. 3) 虽然针对测试实例应用有效不等式(12)较应用有效不等式(10)或同时应用式(10)和(12)获得了更好的总体求解效果, 但存在某些测试问题应用有效不等式(10)求解效果更好, 例如表4中的问题实例J3013_3.SM, J3029_1.SM和J3029_8.SM.

5 结论(Conclusions)

本文基于CP技术针对求解RCPSP的IP模型提出了问题和模型预处理方法以及有效不等式定理. 通过对典型问题的求解实验得出以下结论: 1) 问题和模型预处理过程对问题求解具有有效作用; 2) 本文提出的有效不等式定理对于应用本文模型求解RCPSP具有明显的改进作用, 可以同时提高算法的求解质量和性能; 3) 在模型中添加有效不等式可以在很大程度上提高原模型线性松弛问题的求解速度.

参考文献(References):

[1] BRUCKER P, DREXL A, MOHRING R, et al. Resource-constrained project scheduling: notation, classification, models, and methods[J]. *European Journal of Operational Research*, 1999, 112(1): 3 – 41.

[2] PRITSKER A, WATTERS L, WOLFE P. Multi-project scheduling with limited resources: a zero-one programming approach[J]. *Management Science*, 1969, 16(1): 93 – 108.

[3] CHRISTOFIDES N, ALVAREZ-VALDES R, TAMARIT J M. Project scheduling with resource constraints: a branch and bound approach[J]. *European Journal of Operational Research*, 1987, 29(3): 262 – 273.

[4] DEMEULEMEESTER E L, HERROELEN W S. A branch-and-bound procedure for the multiple resource-constrained project scheduling problem[J]. *Management Science*, 1992, 38(12): 1803 – 1818.

[5] ALVAREZ-VALDÉ R, TAMARIT J M. The project scheduling polyhedron: Dimension, facets and lifting theorems[J]. *European Journal of Operational Research*, 1993, 67(2): 204 – 220.

[6] DEMEULEMEESTER E L, HERROELEN W S. New benchmark results for the resource-constrained project scheduling problem[J]. *Management Science*, 1997, 43(11): 1485 – 1492.

[7] BRUCKER P, KNUST S, SCHOO A, et al. A branch and bound algorithm for the resource-constrained project scheduling problem[J]. *European Journal of Operational Research*, 1998, 107(2): 272 – 288.

[8] MINGOZZI A, MANIEZZO V, RICCIARDELLI S, et al. An exact algorithm for the resource-constrained project scheduling problem based on a new mathematical formulation[J]. *Management Science*, 1998, 44(5): 714 – 729.

[9] MÖHRING R H, SCHULZ A S, STORK F, et al. Solving project scheduling problems by minimum cut computations[J]. *Management Science*, 2003, 49(3): 330 – 350.

[10] DORNDORF U, PESCH E, PHAN-HUY T. A time-oriented branch-and-bound algorithm for resource-constrained project scheduling with generalized precedence constraints[J]. *Management Science*, 2000, 46(10): 1365 – 1384.

[11] HARDIN J R, NEMHAUSER G L, SAVELSBERGH M W P. Strong valid inequalities for the resource-constrained scheduling problem with uniform resource requirements[J]. *Discrete Optimization*, 2008, 5(1): 19 – 35.

[12] 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.

[13] HARTMANN S. A self-adapting genetic algorithm for project scheduling under resource constraints[J]. *Naval Research Logistics*, 2002, 49(5): 433 – 448.

[14] MERKLE D, MIDDENDORF M, SCHMECK H. Ant colony optimization for resource-constrained project scheduling[J]. *IEEE Transactions on Evolutionary Computation*, 2002, 6(4): 333 – 346.

[15] DEBELS D, DE REYCK B, LEUS R, et al. A hybrid scatter sear-ch/electromagnetism meta-heuristic for the resource-constrained project scheduling problem[J]. *European Journal of Operational Research*, 2006, 169(3): 638 – 653.

[16] DEBELS D, VANHOUCKE M. A decomposition-based genetic algorithm for the resource-constrained project-scheduling problem[J]. *Operations Research*, 2007, 55(3): 457 – 469.

[17] VALLS V, BALLESTÍN F, QUINTANILLA S. Justification and RCPSP: A technique that pays[J]. *European Journal of Operational Research*, 2005, 165(2): 375 – 386.

[18] GONÇALVES J F, MENDES J J M, RESENDE M G C. A genetic algorithm for the resource constrained multi-project scheduling problem[J]. *European Journal of Operational Research*, 2008, 189(3): 1171 – 1190.

[19] BALLESTÍN F, VALLS V, QUINTANILLA S. Pre-emption in resource-constrained project scheduling[J]. *European Journal of Operational Research*, 2008, 189(3): 1136 – 1152.

[20] RANJBAR M, DE REYCK B, KIANFAR F. A hybrid scatter search for the discrete time/resource trade-off problem in project scheduling[J]. *European Journal of Operational Research*, 2009, 193(1): 35 – 48.

[21] BAPTISTE P, LE PAPE C, NUIJTEN W. *Constraint Based Scheduling*[M]. Kluwer, Amsterdam: Kluwer Academic Publishers, 2001.

[22] MERCIER L, HENTENRYCK P V. Strong polynomiality of resource constraint propagation[J]. *Discrete Optimization*, 2007, 4(3/4): 288 – 314.

[23] MERCIER L, HENTENRYCK P V. Edge finding for cumulative sche-duling[J]. *INFORMS Journal on Computing*, 2008, 20(1): 143 – 153.

[24] TERCINET F, NÉRON E, LENTÉ C. Energetic reasoning and bin-packing problem, for bounding a parallel machine scheduling problem[J]. *4OR*, 2006, 4(4): 297 – 317.

[25] DEMASSEY S, ARTIGUES C, MICHELON P. Constraint-propagation-based cutting planes: an application to the resource-constrained project scheduling problem[J]. *INFORMS Journal on Computing*, 2005, 17(1): 52 – 65.

[26] BAPTISTE P, DEMASSEY S. Tight LP bounds for resource constrained project scheduling[J]. *OR Spectrum*, 2004, 26(2): 251 – 262.

[27] CARLIER J, NÉRON E. Computing redundant resources for the resource constrained project scheduling problem[J]. *European Journal of Operational Research*, 2007, 176(3): 1452 – 1463.

[28] HOOKER J N. Logic, optimization, and constraint programming[J]. *Informs Journal on Computing*, 2002, 14(4): 295 – 321.

[29] HENTENRYCK P V. Constraint and integer programming in OPL[J]. *Informs Journal on Computing*, 2002, 14(4): 345 – 372.

[30] MARAVELIAS C T, GROSSMANN I E. A hybrid MILP/CP decomposition approach for the continuous time scheduling of multipurpose batch plants[J]. *Computers & Chemical Engineering*, 2004, 28(10): 1921 – 1949.

[31] CHINNECK J W. Special issue on the merging of mathematical programming and constraint programming[J]. *Informs Journal on Computing*, 2002, 14(4): 293 – 294.

[32] RADERMACHER F J. Scheduling of project networks[J]. *Annuals of Operation Research*, 1985, 4(2): 227 – 252.

[33] BARTUSCH M, MOHRING R H, RADERMACHER F J. Scheduling project networks with resource constraints and time windows[J]. *Annals of Operations Research*, 1988, 16(2): 201 – 240.

[34] 刘士新, 郭哲, 唐加福. 具有优先关系的累积调度问题的约束传播算法[J]. 自动化学报, 2010, 36(4): 586 – 592.
(LIU Shixin, GUO Zhe, TANG Jiafu. Constraint propagation for cumulative scheduling problems with precedences[J]. *Acta Automatica Sinica*, 2010, 36(4): 586 – 592.)

[35] BOMZE I M, BUDINICH M, PARDALOS P M, et al. The maximum clique problem[A]. *Handbook of Combinatorial Optimization*[M]. Du D Z, Pardalos P M, Eds. Dordrecht, Netherlands: Kluwer Academic Publishers, 1999.

[36] KOLISCH R, SPRECHER A. PSPLIB—a project scheduling problem library[J]. *European Journal of the Operational Research*, 1997, 96(1): 205 – 216.

[37] LIU S X, YUNG K L, IP W H. Genetic local search for resource-constrained project scheduling under uncertainty[J]. *International Journal of Information and Management Sciences*, 2007, 18(4): 347 – 363.

作者简介:

刘士新 (1968—), 男, 教授, 博士生导师, 从事生产计划与调度、物流系统建模与优化、组合最优化等的研究, E-mail: sxliu@mail.neu.edu.cn;

宋健海 (1965—), 男, 博士, 教授级高级工程师, 从事ERP, MES及企业信息系统整体解决方案等的研究, E-mail: songjianhai@baosight.com.