多机多窗口优化调度问题的模型与算法

刘 伟 孟 新 郑建华

(中国科学院空间科学与应用研究中心，北京 100190)

摘 要：最小化完工费用且具有多个时间窗口的多机调度问题，一直是组合优化领域的一个研究难点.首先给出描述问题的2种模型：整数规划IP(Integer Programming)模型，约束规划CP(Constraint Programming)模型.通过对IP模型和CP模型各自缺点的讨论，引出一个新的模型——混合IP-CP模型，重点讨论了该混合模型的求解方法，给出一个模型求解的启发式算法，经测试表明新模型和算法能极大地提高问题求解效率，为解决此类大规模优化调度问题提供了方法.

关 键 词：多机多窗口优化调度；混合IP-CP模型；启发式算法

中图分类号：TP 39

文献标识码：A **文 章 编 号**：1001-5965(2009)03-0330-04

Multi-machine scheduling problem with multi-time windows: model and algorithm

Liu Wei Meng Xin Zheng Jianhua

(Center for Space Science and Applied Research, Chinese Academy of Science, Beijing 100190, China)

Abstract: With regard to the multi-machine scheduling problem with multi-time windows for minimizing the cost, how to solve the question quickly and effectively remains to be a hard problem in combination optimization research field. Firstly two models were established to describe the simplified questions during the study of optimization, including an integer programming (IP) model and a constraint programming (CP) model. By discussing the defects of the IP model and CP model, a new hybrid IP-CP model was constructed. In addition, a heuristic algorithm was applied to solve the hybrid IP-CP model. Result of the tests indicate that the hybrid IP-CP model and heuristic algorithm developed are proved to be feasible and effective for the multi-machine scheduling problem with multi-time windows, especially for the large-scale scheduling problem.

Key words: multi-machine scheduling; hybrid IP-CP model; heuristic algorithm

并行多机优化调度问题是一类重要的调度问题，很多实际问题如车间调度、多星任务调度都可转化成多机调度问题来处理，文献[1]总结了1990年以前多机调度领域的主要研究工作，文献[2－6]介绍了解决多机调度问题的一些算法(拉格朗日松弛算法、遗传算法等)，这些次优化算法一是以牺牲问题最优解为代价，同时随着问题规模的增加其求解效率仍然不能令人满意，本文提出一种混合IP(Integer Programming)-CP(Constraint Programming)模型及其求解算法，从而很好地解决此类大规模问题.

本文介绍了多机多窗口优化调度问题的定义，给出了解决问题的整数规划模型和基于ILOG规划语言的约束规划模型，通过对上述两个模型的分析导出求解效率高许多的混合模型及求解算法，最后给出相应的测试结果及结论.

1 多机调度问题的两种模型

收稿日期：2008-03-11

作者简介：刘 伟(1981－)，男，安徽合肥人，博士生，wei_liu@cssar.ac.cn.

1.1　问题定义

现有 n 个任务,m 台机器,每个任务可以在任意机器上加工完成,加工时间和加工费用已知,并规定任务执行时不能中断且各任务有且执行一次,不同任务在不同机器上的时间窗口(即任务的可执行时间段,一个或多个)已知,现在要求在完成所有任务的前提下使得花费总费用最小.

根据上述描述下面分别给出问题的 IP 模型和 CP 模型.

1.2　IP 模 型

下面定义 IP 模型中的相关符号和变量.

集合和参数:

E 为任务集合;R 为机器集合;$N_{i,m}$ 为任务 i 对应于机器 m 的时间窗口个数;$W_{i,m}$ 为任务 i 对应于机器 m 的时间窗口集合;$W_{i,m}=\{w_{i,m}^1,\cdots,w_{i,m}^{N_{i,m}}\}$,$w_{i,m}^n=[s_{i,m}^n,e_{i,m}^n]$,$n\in\{1,\cdots,N_{i,m}\}$.

其中,$s_{i,m}^n$,$e_{i,m}^n$ 分别代表在任务 i 占用机器 m 时所允许的第 n 个时间窗口开始时间和结束时间.$\alpha_{i,m}$ 为机器 m 执行任务 i 的花费;$p_{i,m}$ 为机器 m 执行任务 i 的持续时间;M 为很大的整数.

变量:

$$x_{i,m,t}=\begin{cases}1 & \text{任务 } i \text{ 在机器 } m \text{ 的第 } t \text{ 个窗口内执行}\\ 0 & \text{其它}\end{cases}$$

t_i 为任务 i 的开始执行时间.

$$y_{i,j}=\begin{cases}1 & \text{任务 } i \text{ 和任务 } j \text{ 在同一}\\ & \text{机器上执行,且 } i \text{ 在 } j \text{ 前}\\ 0 & \text{其它}\end{cases}$$

下面给出多机多窗口调度问题的 IP 模型:

$$\min\sum_{i\in E}\sum_{m\in R}\alpha_{i,m}\Big(\sum_{t=1}^{N_{i,m}}x_{i,m,t}\Big) \tag{1}$$

约束条件:

$$\sum_{m\in R}\sum_{t=1}^{N_{i,m}}x_{i,m,t}=1\quad\forall i\in E \tag{2}$$

$$\sum_{t=1}^{N_{i,m}}x_{i,m,t}+\sum_{t=1}^{N_{j,m}}x_{j,m,t}\leqslant y_{i,j}+y_{j,i}+1$$
$$\forall i\in E\quad j\in E\quad i\neq j\quad\forall m\in R \tag{3}$$

$$y_{i,j}+y_{j,i}\leqslant 1\quad\forall i\in E\quad\forall j\in E\quad i\neq j \tag{4}$$

$$t_i\geqslant\sum_{m\in R}\sum_{t=1}^{N_{i,m}}x_{i,m,t}s_{i,m}^t\qquad\forall i\in E\quad\forall m\in R \tag{5}$$

$$t_i\leqslant\sum_{m\in R}\sum_{t=1}^{N_{i,m}}x_{i,m,t}(e_{i,m}^t-p_{i,m})$$
$$\forall i\in E\quad\forall m\in R \tag{6}$$

$$t_i-t_j+\sum_{t=1}^{N_{i,m}}x_{i,m,t}p_{i,m}\leqslant M(1-y_{i,j})$$
$$\forall i\in E\quad j\in E\quad i\neq j\quad\forall m\in R \tag{7}$$

$$y_{i,j}+y_{j,i}+\sum_{t=1}^{N_{i,m}}x_{i,m,t}+\sum_{t=1}^{N_{j,l}}x_{j,l,t}\leqslant 2$$
$$\forall i\in E\quad j\in E\quad i\neq j$$
$$\forall m\in R\quad l\in R\quad m\neq l \tag{8}$$

$$y_{i,j}+y_{j,i}+\sum_{t=1}^{N_{i,m}}x_{i,m,t}+\sum_{t=1}^{N_{j,m}}x_{j,m,t}\leqslant 3$$
$$\forall i\in E\quad j\in E\quad i\neq j\quad\forall m\in R \tag{9}$$

$$\left.\begin{aligned}&x_{i,m,t}\in\{0,1\}\quad\forall i\in E\\&\forall m\in R\quad t=1,2,\cdots,N_{i,m}\\&y_{i,j}\in\{0,1\}\quad\forall i\in E\quad\forall j\in E\quad i\neq j\\&t_i\geqslant 0,\text{且 } t_i \text{ 为整数}\end{aligned}\right\} \tag{10}$$

其中,目标函数式(1)保证了所有执行的任务总费用最小.

约束条件式(2)保证了每个任务有且仅执行一次.

约束条件式(3)、式(4)从逻辑关系上保证了任务执行的先后顺序.任务 i 在任务 j 之前执行或者任务 j 在任务 i 之前执行或者二者之间无先后关系(即在不同机器上执行).

约束条件式(5)、式(6)规定了每个任务的执行时间范围.

约束条件式(7)保证了任务执行时不能出现交叉,如果任务 i 在任务 j 之前执行($y_{i,j}=1$),则任务 j 至少要等任务 i 执行完才能开始执行.

约束条件式(8)表明如果任务 i 和任务 j 被分配给不同的机器,则它们之间不存在执行的先后次序,即 $y_{i,j}$ 与 $y_{j,i}$ 均为 0.约束条件式(9)则说明了任务 i 和任务 j 被分配给相同机器时必存在的执行上的先后次序.式(8)、式(9)为冗余约束,不过经测试增加它们可以加快模型的求解速度.

约束条件式(10)对各变量的取值进行了限制.

说明:还可以根据不同的目标准则(如最快完成所有任务等)对模型进行相应地修改,本文仅对最小化开销进行讨论.

1.3　CP 模 型

约束规划方法是用于处理复杂问题的常用方法,尤其是在处理包含多个复杂约束的组合优化问题(如资源配置问题、调度问题等)时,能够比传统数学规划模型更为简洁且易于求解.约束规划模型的建立通常是和规划描述语言、规划软件相关的,如 ILOG 软件,COPS 系统,CHIP 语言,选用较为常用的 ILOG-Solver(ILOG-Scheduler)建立 CP 模型.

相关说明：

i：IloActivity，$\forall i \in E$. 即定义 i 为任务集中的一个任务.

U_m：IloUnaryResuorce，$\forall m \in R$. 定义 U_m 为一元资源(IloUnaryResuorce)，一元资源表示一旦该资源被占用，则在其被释放前其它任务不能占用它.

C_i：决策变量，$\forall i \in E$. 即任务 i 由机器 C_i 执行. 从而有问题的 CP 模型如下：

$$\min \sum_{i \in E} \alpha_{i,C_i} \tag{11}$$

约束条件：

$$\text{IloActivity } i(\text{env}, p_{C_i}, "i") \quad \forall i \in E \tag{12}$$

$$\text{i.requires}(U) \quad \forall i \in E \tag{13}$$

$$\text{IloIfThen}(\text{env}, \text{i.select}(U_m) \quad C_i = m) \quad \forall i \in E \quad \forall m \in R \tag{14}$$

$$\left.\begin{array}{l} \text{IloIfThen}(\text{env}, C_i = m, \\ (\text{i.startsAfter}(s^1_{i,m}) \,\&\&\, \\ \text{i.endsBefore}(e^1_{i,m} - P_{i,m})) \\ \| \ldots \| (\text{i.startsAfter}(s^{N_{i,m}}_{i,m}) \\ \&\& \text{i.endsBefore}(e^{N_{i,m}}_{i,m} - p_{i,m})) \\ \forall i \in E \quad \forall m \in R \end{array}\right\} \tag{15}$$

$$i \in E \quad m \in R \quad C_i \in R \tag{16}$$

式中，目标函数式(11)与 IP 模型中的式(1)对应，即使总开销最小.

约束条件式(12)创建了任务 i 并规定了任务 i 的执行时间.

约束条件式(13)对应于式(2)，即任务需要机器来完成，且默认仅要一台机器.

约束条件式(14)将任务与机器对应起来，即如果任务 i 选择机器 U_m 来执行(i.select(U_m))，则决策变量 $C_i = m$.

约束条件式(15)表示：如果任务 i 选择机器 m 执行，则任务的执行时间必须在其对应的多个时间窗口中的某一个内来执行.

约束条件式(16)对各参变量的取值进行了限制.

本文分别采用 ILOG-CPLEX10.2 和 ILOG-Solver6.4 对 IP 和 CP 模型分别求解，问题规模(使用相同的测试数据)和求解结果见表 1、表 2.

表 1 IP 模型的计算结果

问题编号	任务数/机器数	时间/s	目标值
1	5/2	0.02	15
2	10/4	0.86	55
3	25/6	216.53	325
4	50/8	1 995.04	1 275
5	100/16	10 000	5 231

注：使用 CPLEX 10.2，Pentium D 2.8GHz(1G 内存)台式机.

表 2 CP 模型的计算结果

问题编号	任务数/机器数	时间/s	目标值
1	5/2	0.01	15
2	10/4	0.21	55
3	25/6	19.03	325
4	50/8	431.41	1 275
5	100/16	10 000	5 211

注：使用 Solver 6.4，Pentium D 2.8GHz(1G 内存)台式机.

2 混合 IP-CP 模型

2.1 IP-CP 模型

多机多窗口调度问题的混合 IP-CP 模型如下：

$$\min \sum_{i \in E} \sum_{m \in R} \alpha_{i,m} \left(\sum_{t=1}^{N_{i,m}} x_{i,m,t} \right) \tag{17}$$

约束条件：

$$\sum_{m \in R} \sum_{t=1}^{N_{i,m}} x_{i,m,t} = 1 \quad \forall i \in E \tag{18}$$

$$t_i \geqslant \sum_{m \in R} \sum_{t=1}^{N_{i,m}} x_{i,m,t} s^t_{i,m} \quad \forall i \in E \quad \forall m \in R \tag{19}$$

$$t_i \leqslant \sum_{m \in R} \sum_{t=1}^{N_{i,m}} x_{i,m,t} (e^t_{i,m} - p_{i,m}) \quad \forall i \in E \quad \forall m \in R \tag{20}$$

$$f\left(\sum_{t=1}^{N_{i,m}} x_{i,m,t} = 1 \right) \text{then}(C_i = m) \quad \forall i \in E \quad \forall m \in R \tag{21}$$

$$\text{IloActivity } i(\text{env}, p_{C_i}, "i") \quad \forall i \in E \tag{22}$$

$$\text{i.requires}(U_{C_i}) \quad \forall i \in E \tag{23}$$

$$\left.\begin{array}{l} \text{IloIfThen}(\text{env}, C_i = m, \\ (\text{i.startsAfter}(s^1_{i,m}) \,\&\&\, \\ \text{i.endsBefore}(e^1_{i,m} - p_{i,m})) \\ \| \ldots \| (\text{i.startsAfter}(s^{N_{i,m}}_{i,m}) \,\&\&\, \\ \text{i.endsBefore}(e^{N_{i,m}}_{i,m} - p_{i,m}))) \\ \forall i \in E \quad \forall m \in R \end{array}\right\} \tag{24}$$

$$\left.\begin{array}{l} x_{i,m,t} \in \{0,1\} \quad \forall i \in E \\ \forall m \in R, t = 1,2,\cdots,N_{i,m} \\ t_i \geqslant 0, \text{且 } t_i \text{ 为整数} \\ C_i \in R \end{array}\right\} \tag{25}$$

其中，目标函数式(17)同式(1).

约束条件式(18)～式(20)分别来自 IP 模型中的式(2)、式(5)、式(6).

式(21)将 IP 模型和 CP 模型联系起来，即将变量 $x_{i,m,t}$ 与 C_i 联系起来.

式(22)～式(24)对应 CP 模型中的式(12)、式(13)、式(15).注意式(23)与式(13)的区别,即已经由式(21)指定了 C_i 的值.

式(25)则是对各变量取值的限定.

上述混合模型成功地处理了 IP 模型中的复杂约束,并且消除了变量 $y_{i,j}$,这得益于 CP 模型的帮助.即只要对式(17)～式(20)进行求解,得到 $x_{i,m,t}$ 和 t_i 的值(实际上有用的仅是 $x_{i,m,t}$ 的值),交由式(21)得到 C_i 的值,再去解式(22)～式(24)即能得到原问题的最优解(当然可能需要多次迭代求解).同样地,CP 模型由于不要考虑目标函数(最优解),同时借助于 IP 模型得到的确定的 C_i 值,仅需要求解各任务的执行时间(而这是 CP 的优势)即可.下面给出求解上述混合模型的启发式算法.

2.2　启发式算法

1) 令 $k=0$,解式(17)～式(20),得到 $x^k_{i,m,t}$ 值,交由式(21)得到 C^k_i 的值,转 2).

2)由确定的 C^k_i 的值,解式(22)～式(24),如果有可行解,转 5);否则,$k=k+1$,转 3).

3)令 $d^k_m=\dfrac{\sum\limits_{i\in E}N_{i,m}}{\sum\limits_{i\in E}\sum\limits_{t=1}^{N_{i,m}}x^{k-1}_{i,m,t}}$,$\forall m\in R$,记 $n^k=\{m\mid \min\limits_{m\in R}d^k_m\}$,并增加新约束式:

$$\sum_{i\in E}\sum_{t=1}^{N_{i,n^k}}x^k_{i,n^k,t}\leqslant\sum_{i\in E}\sum_{t=1}^{N_{i,n^k}}x^{k-1}_{i,n^k,t}-1 \qquad (26)$$

转 4).

4) 解式(17)～式(20)、式(26),得到 $x^k_{i,m,t}$ 值,交由式(21)得到 C^k_i 的值,转 2).

5) 输出最优解.

上述算法在遇到不可行情况时(即由松弛后的 IP 模型得到的 $x_{i,m,t}$ 值无法从 CP 模型中得到可行解),就会对 IP 模型得到的解进行必要的改进:尽可能的平衡各机器的负载(使用率),对于那些平均时间窗口数(机器总窗口数/机器分到的任务数)较少的机器进行调整,使得下次迭代得到的解中该机器的平均窗口数有所增加,因为往往就是由于那些平均时间窗口数少的机器上的任务无法全部安排造成了不可行.

表 3 给出了基于混合 IP-CP 模型下的计算结果(测试数据同表 1、表 2 中的数据).从中可以看出:采用混合 IP-CP 模型及启发式算法,可以极大的提高问题求解速度,解的精确度也有所提高(更容易获得最优解),随着问题规模的加大体现得更为明显,同时比起其它的优化算法(拉格朗日松弛算法、遗传算法等)也具有上述优势.

表 3　混合 IP-CP 模型的计算结果

问题编号	任务数/机器数	IP 用时/s	CP 用时/s	迭代次数	目标值
1	5/2	0.02	0.00	1	15
2	10/4	0.39	0.11	3	55
3	25/6	6.02	1.34	11	325
4	50/8	14.23	9.08	30	1 275
5	100/16	25.15	13.67	46	5 050

注:使用 CPLEX 10.2,Solver 6.4,2.8GHz(1G 内存)台式机.

3　结 束 语

本文针对多机多窗口优化调度问题进行研究.在给出 IP 模型和 CP 模型的基础上对两种模型的不足进行分析,采用了一种混合 IP-CP 的新模型和启发式算法,通过实验表明新模型和算法能够很好的适用于此类大规模问题.

参考文献(References)

[1] Cheng T. A state-of-the-art review of parallel machine scheduling research [J]. European Journal of Operation Research, 1990,47:271－292

[2] Fisher M. The Lagrangian relaxation method for solving integer programming problems [J]. Mgmt Sci,1981,27:1－18

[3] Muckstadt J A,Koening S A. An application of Lagrangian relaxation to scheduling in power-generation systems [J]. Oper Res,1977,25: 387－403

[4] Dincbas M. The constraint logic programming language CHIP [J]. Proceedings of International Conference on Fifth Generation Computer Systems, 1988,1: 693－702

[5] Akker V. Time-indexed formulations for machine-scheduling problems: column generation [J]. Informs J Comput,2000, 12(2): 111－124

[6] Hoitomt D. Scheduling job with simple precedence constraints on parallel machines [J]. Control Syst Mag,1990, 10: 34－40

(责任编辑:张　嵘)