

基于过滤束的转移瓶颈算法在JSSP中的应用

黄肖玲¹ 杨焕熙² 魏俊秀¹

(辽宁大学轻型产业学院 沈阳110036)¹ (辽宁大学信息学院 沈阳110036)²

摘要 举例证明了传统转移瓶颈算法在求解瓶颈机时并不能得到局部最优解。提出一种新的确定瓶颈机的模型,在综合考虑时间和求解精度的情况下,采用分支定界方法的改进方法——过滤束搜索来求解此模型。在过滤束中引用了不同工件到达时间和处理时间的约束关系来解决成环问题。通过对OR-Library中的33个标准job shop问题的实验结果看,此算法得到了比较满意的效果。

关键词 作业车间调度,转移瓶颈,过滤束搜索,调度模型

Shifting Bottleneck Algorithm Based on Filtered Beam Search for JSSP

HUANG Xiao-ling¹ YANG Huan-xi² WEI Jun-xiu¹

(College of Light Industry, Liaoning University, Shenyang 110036, China)¹

(College of Information Industry, Liaoning University, Shenyang 110036, China)²

Abstract It was proved by example that the traditional shifting bottleneck cannot obtain a local optimal solutions in solving bottleneck machine, a new model of bottlenecks was presented and it was solved by Filtered beam search—the improving algorithm of branch and bound method in the comprehensive consideration of time and accuracy. The relation of starting time and processing time of different work was considered to avoid cycle in the filtered beam search. We obtained a satisfactory result by the experiments of 33 benchmark problem of OR-Library.

Keywords Job shop scheduling, Shifting bottleneck, Filtered beam search, Scheduling mode

1 引言

作业车间调度问题(JSSP)通常也称为排序问题或资源分配问题,是对具体生产环境中复杂的、动态的、多目标的排程问题的一种抽象和简化,属NP-难问题^[1],也是最难的组合优化问题之一。在对排程问题的研究方法上,最初是集中在整数规划、仿真和简单的规则上,这些方法不是排程结果不理想就是难以解决复杂的问题。随着各种新的相关学科和优化技术的发展,在排程领域也出现了许多新的优化方法,比如神经网络算法、模拟退火、蚁群算法、遗传算法、转移瓶颈法(SB)和禁忌搜索法等,使得该问题的研究方法向多元化方向发展。其中SB算法是目前求解JSSP的较好方法之一。

经检索国内外期刊和IEEE数据库,目前关于SB用于解决调度问题的研究主要集中在模型的评价^[2]和与其它算法的结合上^[3],而对算法本身的研究比较少。传统的SB算法是由一系列的单机调度问题用Calier算法求解组成。但是Calier算法并未考虑约束条件,很多学者针对这个问题做了一些改进:文献[13]举例证明环的存在性,并且用开工时间小于另一个未加工工件的可完成时间来约束工件执行顺序,以消除环;文献[9]在 r (到达时间)中引入 q (剩余加工时间),提高了 q 的重要性,从而降低形成环的概率;文献[11]通过矩阵的直观性,来消除没考虑约束产生的环。由此可见,目前所做的工作

都是针对Calier算法中约束产生的环来做的改进。本文发现Calier算法不仅存在环的问题,而且即使在无环的情况下也并非单机调度问题的最优解。因此,本文证明了Calier算法本身是得不到局部最优解的,并提出了一种新的求解瓶颈机的模型。

束搜索(Beam Search)是在分支定界方法的基础上发展起来的一种启发式算法,它使用启发式方法估计 k 个最好的路径,仅从这 k 条路径出发向下搜索,即每一层只有满意的节点会被保留,其它的节点则被永久抛弃,因此比分支定界法大大节省了运行时间^[7]。它的改进算法——过滤束搜索^[9](filtered beam search, FBS)于1999年由Sabuncuoglu与Bayiz应用到job shop排序问题,其最大的优点是减少了不必要的分支,因此时间上要优于分支定界算法;其缺点是在问题比较复杂时求解时间仍然太长,导致不能在合理的时间内求得最优解。

本文综合考虑以上两种算法中的优缺点,在举例证明Calier算法存在缺陷的前提下,用 $1|r_j|L_{max}$ 代替 $1|r_j|C_{max}$ 建立模型来确定瓶颈机,用过滤束算法来求解此模型,并且采用公式 $r_{jk} < \min_{l \in J} (t, r_l) + p_l$ 来克服同一机器上不同工件的约束问题和减少不必要的解。在此方法中,过滤束方法只解单机问题,这样可以避免算法的复杂度,同时因为不采用Calier算法解单机问题,可以提高局部解的精确度,从而得到更好的全

到稿日期:2008-04-01 本文受辽宁省教育厅高等学校科目(20060335),国家自然科学基金重点项目(60534010)资助。

黄肖玲(1962—),女,教授,博士,研究方向为流程工业制造执行系统、生产计划与调度、智能优化方法等;杨焕熙(1982—),男,硕士研究生,主要研究方向为调度、智能优化方法;魏俊秀(1968—),男,副教授,主要研究方向为智能检测与运动控制。

局解。

2 问题描述

JSSP 可以描述为:设有 N 个工件在 M 台机器上加工,由于工件加工工艺的要求,每个工件使用机器的顺序以及机器每道工序所花时间已给定,调度问题就是如何安排在每台机器上工件的加工顺序,使得某种指标最优。具体满足下面约束条件:(1) 每道工序必须在指定的机器上加工;(2) 每道工序必须在前一道工序加工完成后才能开始加工;(3) 某一时刻一台机器只能加工一个工件;(4) 各工件的工艺路线和工件的每道工序的持续时间已知,且不随加工排序的改变而改变。令

$J = \{1, \dots, n\}$ 代表工件的集合;

$M = \{1, \dots, m\}$ 代表机器的集合;

$N = \{(i, j) | i = 1, \dots, n, j = 1, \dots, m\}$ 代表所有工序对集合;

合;

$A = \{(i, j) \rightarrow (k, j) | (i, j) \in N, (k, j) \in N\}$ 代表已排好序的集合;

$B = \{(i, j) \rightarrow (i, k) | (i, j) \in N, (i, k) \in N\}$ 代表未确定的排序;

p_{ij} : 工件 j 在机器 i 上的加工时间, $i = 1, \dots, m, j = 1, \dots, n$;

r_{ij} : 工件 j 在机器 i 上的到达时间;

y_{ij} : 工序 (i, j) 的开始加工时间, $i = 1, \dots, m, j = 1, \dots, n$;

则问题可以形式化的描述为:

Min C_{\max}

其中,

$$y_{ij} - y_{ik} \geq p_{ij} \quad (i, j) \rightarrow (i, k) \in A \quad (1)$$

$$C_{\max} - y_{ij} \geq p_{ij} \quad (i, j) \in N \quad (2)$$

$$y_{ij} - y_{ik} \geq p_{ij} \text{ or } y_{ik} - y_{ij} \geq p_{ij} \quad i \in M \quad (3)$$

$$y_{ij} \geq 0 \quad (i, j) \in N \quad (4)$$

约束(1)确保操作 (i, j) 在操作 (i, k) 之前进行,约束(2)确保完工时间在最后一个工件完工后,约束(3)确保分离图中相关两个操作的优先顺序,约束(4)确保开工时间大于 0。图 1 给出了 3 个工件在 4 个机器上加工的 (N, A, B) 分离图,其中 N 为工序对集合, A 为连接边的集合, B 为分离边的集合,结点 (i, j) 代表工件 i 在机器 j 上加工,边代表加工时间,问题就是根据目标函数求解使得图中最长路径最小的完全图。

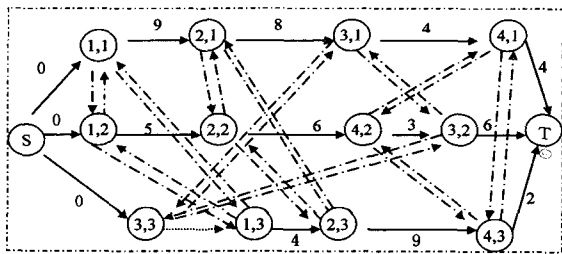


图 1 作业车间调度问题的分离图

3 过滤束搜索方法

FBS 算法是束搜索方法的一种改进,其基本思想是在搜索树的第一层,对所有的节点进行全局评价以选取 β 个满意解,作为 β 个并行束的起始节点,称其为 beam 结点。之后,对

于每一束,由每个 beam 结点产生的子结点都要进行两次评价,首先进行局部评价,根据评价结果在由 beam 结点产生的所有子结点中选择 α 个满意的结点;然后用全局评价函数对这 α 个结点进行全局评价,从中选出最好的一个作为下一层的 beam 结点。称 β 为束宽 (beam width), α 称为过滤宽度 (filter width)。FBS 算法过程如图 2 所示。

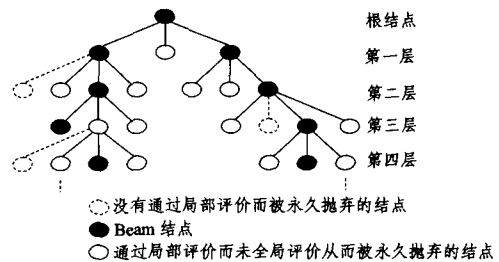


图 2 过滤束分支图

本文的局部评价为 r_{ij} , 选取 r_{ij} 最小的 α 个节点, 全局评价为 L_{\max} 。在求解 $1 | r_{ij} | L_{\max}$ 单机模型时, 采用分支定界法可以求得最优解。但是考虑到分支定界的分支太多, 计算复杂等原因, 我们采用它的改进方法——过滤束算法。此算法比分支定界算法在时间上有大大改进, 并且在解的精确度上与分支定界算法的误差不大。

4 转移瓶颈方法及其改进

4.1 原始 SB 算法及其缺陷证明

Calier 算法中用到了 Schrage 算法, 描述如下。令

I_i : 机器 i 上的工序集合;

U_i : 机器 i 上已经调度的工序的集合;

\bar{U}_i : 机器 i 所有未调度的集合;

r_{ij} : 工件 j 到达机器 i 的时间(从 S 到结点 (i, j) 的最远距离);

q_{ij} : 工序 (i, j) 到终点 V 的最远距离减去 p_{ij} 。

对每一台机器 i ,

$$(1) t = \min r_{ij}; U_i = \Phi;$$

(2) 在 t 时刻, 从“就绪”工序 $(i, j) (r_{ij} \leq t)$ 中选择 q_{ij} 值最大的工序 (i, j) 加工;

(3) $U_i = U_i \cup \{(i, j)\}; t_j = t; t = \max(t_j + p_{ij}, \min r_{ij})$ 。如果 $U_i = I_i$, 则算法结束; 否则跳到 (2)。

虽然在文献[13]中考虑了工序之间的依赖约束关系, 并且针对依赖关系提出了一种改进, 但是它的本身算法仍然是 Schrage 算法, 而此算法很多情况下并不能得到最优解。我们可以用反例证明一下, 如图 3 所示。

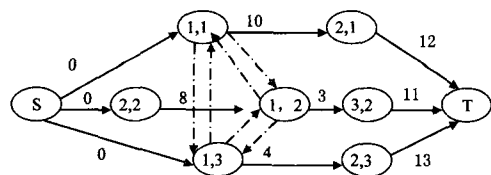


图 3 作业车间调度问题的部分分离图

图 3 的机器 1, $q_{11} = 12, q_{12} = 11, q_{13} = 13$ 。根据上面的 Schrage 算法, 机器 1 的排序为 $r_{13} \rightarrow r_{11} \rightarrow r_{12}$, 它的 C_{\max} 为 28, 如果我们按照 $r_{11} \rightarrow r_{12} \rightarrow r_{13}$ 的顺序的话, 它的 C_{\max} 是 27。由

此可以看出采用 Schrage 算法未必能得到最优解。

4.2 改进的转移瓶颈算法

上面的证明可看到 Schrage 算法存在很大缺陷,未必能得到最优解,故本文采用一种新的确定瓶颈机的方法。

$$\text{令 } d_{ij} : C_{\max} - q_{ij}$$

把每一单机问题转化为 $1 | r_j | L_{\max}$, 而此问题也是 NP-hard 问题,通常的解法是分支定界法,而分支定界法往往比较复杂,求解时间长,因此我们采用其改进的方法——过滤束方法来求解,求得的最大 L_{\max} 就是瓶颈机。总的算法流程为:

(1) 令 $M_0 = \phi$ (M_0 是为安排调度的机器), 图 G 包含所有的链接边,并且去掉所有的分离边,令 $C_{\max}(M_0)$ 等于图 G 中的最长路径。

(2) 对每一个 $M-M_0$ 的机器 i 做如下操作: 产生模型 $1 | r_{ij} | L_{\max}$ 的实例(其中 r_{ij} 是从结点 S 到结点 (i, j) 的最长路径)利用过滤束方法优化此模型,令 $L_{\max}(i)$ 代表机器 i 中最小的 L_{\max} 。

(3) 令 $L_{\max}(k) = \max_{i \in (M-M_0)} (L_{\max}(i))$, 根据步骤(2)中求得循序安排机器 k , 把相关的分离边插入到图 G , 并且把 k 插入到 M_0 , 并且令 $C_{\max}(M_0) += L_{\max}(k)$ 。

(4) 对每一个 $i \in (M_0 - k)$ 的机器删除其在图 G 中加入的相关边,重新利用过滤束方法求其 $L_{\max}(i)$, 并插入相关边。

(5) 如果 $M_0 = M$, 则结束, 否则跳转到(2)。

此方法的主要思路是不断确定瓶颈机的过程,不过它的瓶颈机的确立及求解过程都已发生了很大变化。对瓶颈机的求解更加精确,从而提高了全局解的质量。

5 算法仿真及结果研究

为了分析 FSB 的性能,我们首先与采用分支定界法的瓶颈转移算法(BBSB)在误差和计算时间上做比较,如表 1 所列,然后与两种主流的运用瓶颈转移的算法 TSSB, SB-GLS1 分别做比较,如表 2 所列。实验采用 OR-Library 中的 33 个标准 job shop 问题,其中包括 5 个 ORB 问题、3 个 FT 问题、5 个 ABZ 问题、20 个 TA 问题,并且令过滤束中的参数 α 为 4, β 为 3, Opt 为最优值, LB, UB 分别为上下界。

表 1 BBSB 和 FSB 的时间和误差比较

问题	n	m	Opt(LB UB)	BBSB	RE _{BBSB}	TAV	FSB	RE _{FSB}	TAV
ORB1	10	10	1059	1064	0.47	80	1066	0.66	69
ORB2	10	10	888	890	0.23	75	891	0.34	60
ORB3	10	10	1005	1013	0.80	87	1010	0.50	68
ORB4	10	10	1005	1013	0.80	75	1013	0.80	70
ORB5	10	10	887	887	0.00	81	890	0.34	67
FT6	6	6	55	55	0.00	50	55	0.00	49
FT10	10	10	930	930	0.00	80	930	0.00	70
FT20	20	5	1165	1165	0.00	115	1170	0.43	102
ABZ5	10	10	1034	1234	0.00	75	1234	0.00	70
ABZ6	10	10	943	943	0.00	80	943	0.00	71
ABZ7	20	15	656	666	1.52	200	626	0.00	150
ABZ8	20	15	(645 669)	678	5.12	205	680	5.36	165
ABZ9	20	15	(661 679)	693	4.84	195	697	5.45	163

从表 1 中的 13 个标准模型比较结果看,在误差上,两者有 5 个相同、6 个 BBSB 优、2 个 FSB 优,但是两者误差都在 1% 以内。但是从时间上看,所有的 FSB 计算时间都比 BBSB 少,所以综合考虑 FSB 要明显好于 BBSB。

表 2 FSB 和 TSSB, SB-GLS1 在误差上的比较

问题	n	m	Opt(LB UB)	TSSB	RE _{BBSB}	SB-GLS1	RE _{SB-GLS1}	FSB	RE _{FSB}
TA1	15	15	1231	1241	0.812	1244	1.056	1231	0.000
TA2	15	15	1244	1244	0.000	1255	0.884	1422	0.000
TA3	15	15	(1206 1218)	1222	1.327	1225	1.575	1218	0.995
TA4	15	15	(1170 1175)	1175	0.427	1191	1.795	1181	0.995
TA5	15	15	(1210 1228)	1229	1.570	1256	3.802	1233	1.901
TA6	15	15	(1210 1239)	1245	2.893	1247	3.058	1243	2.727
TA7	15	15	(1223 1228)	1228	0.490	1244	1.717	1228	0.409
TA8	15	15	(1187 1217)	1220	2.780	1222	2.949	1217	2.527
TA9	15	15	(1247 1274)	1291	3.528	1291	3.528	1274	2.165
TA10	15	15	1241	1250	0.725	1266	2.015	1241	0.000
TA11	20	15	(1321 1364)	1371	3.785	1402	6.132	1409	6.662
TA12	20	15	(1321 1367)	1379	4.391	1416	7.192	1367	3.482
TA13	20	15	(1271 1350)	1362	7.160	1377	8.340	1350	6.216
TA14	20	15	1345	1345	0.000	1361	1.190	1345	0.000
TA15	20	15	(1293 1342)	1382	5.182	1383	6.961	1353	4.640
TA16	20	15	(1300 1368)	1370	5.385	1418	9.077	1369	5.308
TA17	20	15	(1458 1464)	1481	1.578	1519	4.184	1478	1.372
TA18	20	15	(1369 1396)	1426	4.64	1433	4.675	1396	1.972
TA19	20	15	(1276 1341)	1351	5.878	1376	7.837	1341	5.094
TA20	20	15	(1316 1353)	1366	3.799	1398	6.231	1359	3.267

从表 2 计算结果我们可以看到,其中有 4 个问题得到了最优解。FSB 在和 TSSB 比较中,2 个比 TSSB 差、2 个与 TSSB 相同,剩下都比 TSSB 优。而在与 SB-GLS1 的比较中,仅 1 个比 SB-GLS1 差,其余的全优于 SB-GLS1。因此我们可以看出,采用过滤束的瓶颈转移算法可以大大提高算法性能。

结束语 本文举例证明了采用 Calier 算法的转移瓶颈算法在求解局部解的时候存在着不能得到单机最优解的缺陷,提出了把 $1 | r_j | C_{\max}$ 模型转换成 $1 | r_j | L_{\max}$ 模型来求解单机局部解,采用分支定界的改进方法——过滤束方法来求解。通过与 BBSB, TSBS, 以及 SB-GLS1 问题的不同标准 job shop 问题的比较,可以看出 FSB 无论从时间还是误差上都取得了理想的效果。

参考文献

- [1] Garey M R, Johnson D S, Sethy R. The complexity of flow shop and job shop scheduling [J]. Mathematics of Operations Research, 1976, 1(2): 117-29
- [2] Pezzella F, Merelli I. A tabu search method guided by shifting bottleneck for the job shop scheduling problem [J]. European Journal of Operational Research, 2000(120): 297-310
- [3] Adams J, Balas E, Zawack D. The Shifting Bottleneck Procedure for Job Shop Scheduling [J]. Management Science, 1988, 34(3): 391-401
- [4] Carlier J. The One-machine Sequencing Problem [J]. Journal of Operational Research, 1982, 11: 42-47
- [5] Loerre B T. The Harpy Speech Recognition System [d]. Pittsburgh: Carnegie Mellon University, 1976
- [6] Ow P S, Morton T E. Filtered beam search in scheduling [J]. International journal of Production Research, 1988, 26(1): 35-62
- [7] Pinedo M. Scheduling, Theory, Algorithms, and Systems [M]. New Jersey: Prentice Hall, 2005
- [8] Morton T E, Pentico D W. Heuristic Scheduling Systems [M]. New York: Wiley, 1993
- [9] Zhang Defu, Li Tangqiu, Li Shaozi. An Improved Shifting Bottleneck Algorithm for Job Shop Scheduling Problem // 9th International Conference on Computer Supported Cooperative Work in Design Proceedings. 2005: 1112-1116

应考虑的问题。到目前为止,已经有不计其数研发完成的新技术因其成本太高而被扼杀在摇篮里。这也是数据云所面临的现实问题。通俗地说,未来的数据云就是要取代如今计算机中的内存和硬盘间相互传输数据的功能,通过网络实现计算机与数据云之间的交换,完成数据存储和计算,并将结果返回给计算机用户。正如目前呼声越来越高的网格技术一样,在单个或几个企业内部使用网格计算会达到优化系统、简化管理、降低成本的目标,但是一旦其应用区域超过了企业范围甚至变为服务于全球性的技术,则对服务器的空间大小和响应速度提出了完全不同的要求,同时需要高速网络传输和数据安全保证,因此大范围应用该技术的边际收益很可能成为负数,整体成本也直冲而上。数据云技术也面临着相同的问题。这种规模效应的递减在经济学中是一个非常简单的道理,同时也应该引起数据云技术研究者的高度重视。否则,数据云将来也会面临有技术支持无市场需求的窘境。

3.2 异构性

据初步估算,仅中国国内就有近 60 家大大小小的硬件生产厂家,全球范围内的厂家数量更是大得惊人。目前,业内对于同类硬件商品并无统一生产标准,众多厂家生产的计算机硬件必然具有异构性。退一步讲,即使全球只有两家公司生产硬件,只要不能形成统一的制造标准,就会对数据云技术的应用产生巨大阻力。未来的数据云作为一个全球性的统一服务器和网络硬盘,很难有效分辨所有硬件的异构部分并加以区别处理,这是数据云在技术方面遇到的最大难题。例如,如果每一台计算机的声卡不同,所需驱动程序就会不一样,因而面对通过数据云传输而来的相同声音数据,必定会出现一部分“哑机”的情况。

3.3 安全问题

数据云的安全威胁主要来自两个方面:第一,互联网故障或中断造成的用户数据损坏或丢失;第二,由于计算机病毒、黑客软件等造成用户的秘密文件被窃取。

互联网发生故障的概率会随着互联网服务器的不断升级以及互联网传输技术的不断改进而降低。但是故障和中断不仅仅来源于技术方面,更多的还来自于自然灾害和人为破坏。谁又能保证将来为我们服务的数据云会像硬盘一样忠诚地始终随着计算机的启动而启动呢?

从网联网开始应用的第一天到现在,虽然各种防火墙、密码认证等安全保障技术层出不穷,但是“道高一尺,魔高一丈”,计算机病毒和黑客软件对个人数据的威胁有增无减,如果数据云的管理者不能向用户保证数据的完整性和安全性,用户肯定不能产生对它的完全信赖;反过来,即使有人敢保证用户把自己的东西存在网络上是完全安全可靠的,聪明或谨慎的用户就会相信吗?

3.4 时间考验

目前各大研究机构所憧憬的数据云是一种能够取代现有 PC 机硬盘、通过服务器与 PC 机之间的数据交换实现数据处理和存储的技术,这就意味着计算机硬盘将会完全被排斥于 PC 机之外。

目前全球平均互联网普及率仅为 19.1%,全球未连接互联网的 PC 机数量众多,要让互联网的光芒洒到地球的每个角落,尚需假以时日。因此,数据云技术目前只能在小范围内发展,而在全球范围内其发展所需要的网络环境暂时还不具备。数据云的研究者们必须耐得住寂寞,经得起时间的考验,等待网络环境成熟之时,再将数据云推出市场。

另一方面,数据云完全代替硬盘和 U 盘的道路也很艰难,硬盘、U 盘存储上的文件资料取用方便且更让用户放心,在这点上目前技术上定义的数据云还远远无法和其竞争。数据云技术必须有一种包容力和毅力,也许数据云与硬盘、U 盘共同掌管计算机存储世界的时间将会维持很长。

结束语 数据云技术正在成为存储领域的又一次巨大创新和革命。它在现有的限量网络存储基础上,将令每个用户都能够通过互联网迅速链接资源并且及时得到响应而且同步共享海量数据,而不受任何用户自身设备的制约。从这个角度来说,数据云要承担现有 PC 内存和硬盘的双重任务,既要作为无限网络内存,通过服务器实现用户数据的计算和存取;又要作为无限网络保险柜,存储用户需要长时间保存的数据并保障其安全。可以构想,尽管还面临着不少亟待攻关的难题,但数据云的未来大范围应用过程与方式,会通过现有的网络存储、加密保护、网络计算、直接内存存取等以及将来可能出现的网络交换技术的大整合,而将整个世界虚拟化并重构成一个超级巨型计算机群世界。这个超级巨型计算机群,拥有分布在全球各地的、高度灵活的前台操作和显示工具,也具有数据云这个无限量内存和硬盘作为强力支撑与高效处理的强大后台数据基地。

总之,随着数据云技术的日渐进步与日益成熟,它终将为 PC 机“减肥”和网络“健身”,使它们的内存和硬盘无限扩充,从而为人们带来无限量的共享资源,并将促使全球更快地迈向更高阶段的网络化世界。

参 考 文 献

(上接第 256 页)

[10] Kasin O, Scott J M. Scheduling Batch Processing Machines in Complex Job Shops[J]//Proceedings of the 2001 Winter Simulation Conference. 2001:1200-1207
[11] Oey k, Mason S J. Scheduling batch processing machines in complex job shops[J]//Proceeding of the 2001 Winter Simulation

[1] Roush W. The Internet Is Your Next Hard Drive. Technology Review [J], July 2006
[2] DAS, NAS, SAN 存储系统完全分析. <http://biz.chinabyte.com/128/2300128.shtml>
[3] 九陆. 中容数据保驾“在线存储”. 通信产业报, 2006-03-27
[4] Xdrive 网络硬盘[J]. 计算机采购, 2000(43)
Conference. 2001:1200-1207
[12] Jens Z, Lars M. Simulation-based selection of machine criticality measures for a shifting bottleneck heuristic[J]//Proceedings of the 2006 Winter Simulation Conference. 2006:1848-1854
[13] 黄文奇, 黄志. 作业车间调度问题的一种改进的转移瓶颈算法[J]. 计算机工程与应用, 2005(2):59-62