

文章编号:1671-6833(2017)05-0086-05

带批处理机的多阶段柔性流水车间调度优化

王君妍,王薛苑,轩 华

(郑州大学 管理工程学院,河南 郑州 450001)

摘 要:从钢铁行业的炼钢—连铸—热轧过程提炼出中间阶段有多台批处理机,其它阶段为离散机的多阶段柔性流水车间调度问题.首先,结合工件动态到达、各阶段间的运输时间以及机器的调整时间等生产特征,对问题进行描述,建立以最小化总加权完成时间为目标的数学模型.然后,针对该问题提出了改进的自适应遗传算法,使遗传参数随其迭代及适应函数值调节.对 150 个工件的大量随机数据进行测试,结果表明,与常规遗传算法相比,所提出的自适应遗传算法能在较短的计算时间内得到更好的解;与拉格朗日松弛算法相比,求解大规模问题时,所提算法在解的质量方面优势较为明显.

关键词:柔性流水车间调度;批处理机;总加权完成时间;自适应遗传算法;自适应调节

中图分类号:TB49 文献标志码:A doi:10.13705/j.issn.1671-6833.2017.05.012

0 引言

在钢铁行业,炼钢、连铸、热轧作为炼钢的主要工序,在整个流程中起着重要作用.在炼钢—连铸—热轧过程中,转炉生产出来的钢水经过精炼炉精炼后,要铸造成不同类型、不同规格的板坯;将装有钢水的钢包运至回转台,回转台转动到浇铸位置后,钢水被注入中间包,钢水分配到各个结晶器中结晶成各种铸件,之后铸件经过切割机的连续切割成为板坯^[1].使用同一中间包的多个炉次(称为一个浇次)要按一定顺序连续生产.进而,板坯由保温车运送到加热炉进行加热,直至达到热轧加工的温度要求,然后,先由高压水枪祛除板坯表面的氧化鳞片,再经定宽压力机对板坯进行侧压以调整其宽度,并依次通过粗轧机进行多道次可逆轧制^[2-3].

上述生产过程中,各阶段由多台同构并行机组成,并且在连铸阶段一个浇次内的所有炉次都要在同一台连铸机上进行无间断的加工,因此该结构可看作是中间阶段具有串行批处理机的柔性流水车间.考虑到钢铁生产中各个加工阶段的运输时间、机器的调整时间以及工件的释放时间,提出笔者所研究的多阶段柔性流水车间问题(flexi-

ble flowshop scheduling problem, FFSP).

对于两机流水车间批调度,文献[4]提出了一种分解启发式算法用来求解工件的总加权拖期费用;文献[5]提出以最小化加工时间跨度为目标的蚁群优化算法;文献[6]以最小化完工期 makespan 为目标,考虑工件动态到达,构建了一种最优化算法以求解各批次加工时间为定值的流水车间批调度问题;文献[7]针对差异工件批处理机调度,设计了一种基于 LPT 规则和批调度近似规则的算法用以最小化制造跨度和总完工时间.

就 FFSP 的串行批问题,文献[8]研究了任意一个阶段有串行批处理机且其它阶段为离散机的多阶段 FFSP,提出了改进的拉格朗日松弛(Lagrangian relaxation, LR)算法,目标为总加权完成时间最小.就 FFSP 的并行批问题,文献[9]提出了一个新的遗传算法(genetic algorithm, GA)最小化 makespan;文献[10]利用启发式搜索 GA 求解一个阶段含并行批处理机的情况,目标分别为最小化 makespan 及最小化总加权拖期费用.

从上述研究现状可知,对于两机流水车间问题,多采用一些启发式算法或最优化算法来求解最大完工时间最小化问题;针对含串行批的 FFSP,多采用 LR 算法最小化总加权完成时间;针对含并行

收稿日期:2017-01-16;修订日期:2017-04-30
基金项目:国家自然科学基金资助项目(U1604150);教育部人文社会科学研究基金项目(15YJC630148);郑州大学优秀青年教师发展基金项目(1421326092)
通讯作者:王薛苑(1985—),男,河南郑州人,郑州大学讲师,博士,主要从事生产调度等方面的研究,E-mail:wangxueyuan@zzu.edu.cn.

批的 FFSP,则多采用启发式、GA 等最小化 makespan. 针对具有串行批处理机的 FFSP,GA 求解总加权完成时间问题的研究较少.

在 GA 中, p_c 和 p_m 等遗传参数常固定不变,因此, Srinivas 等^[11] 首先提出了用 AGA 来求解多目标函数优化问题. 后来也有一些学者^[12-15] 对 AGA 在车间调度中的应用进行了研究,文献[12]对流水车间问题进行建模,采用 AGA 仿真求解出最优调度方案;文献[13]为了解决置换流水车间调度问题,提出了基于模拟退火算法的多种群和自适应遗传算法;文献[14-15]均利用 AGA 求解了 FFSP,目标为最小化 makespan.

综上所述,针对参数自适应的研究多集中在流水车间和 FFSP 上,目标多为最小化 makespan. 由于笔者所研究的多阶段 FFSP 考虑了批生产方式和批生产阶段机器的调整时间、工件动态到达以及阶段间运输时间,是一个较为复杂的 NP-hard 问题,因此,需要合理地设计 GA 以提高其求解能力. 故而,笔者提出遗传参数自适应调节策略改进遗传算法(improved adaptive genetic algorithm, IAGA),旨在扩展 GA 对该类问题的求解能力.

1 问题描述与建模

1.1 问题描述

中间阶段为批处理机的多阶段 FFSP 可描述为: n 个工件依次经过 s 个加工阶段进行加工,当经过某阶段 $k(1 < k < s)$ 时,同一批内的工件按照已知的优先级顺序在一台机器上进行不间断的加工. 至少一个阶段有两台或两台以上的同构并行机. 第 l 批内包含的工件数 B_l 已知,工件只有结束前一阶段的加工才能进入下一阶段. 假设工件动态到达第一阶段,阶段间的运输时间和批生产阶段机器的调整时间不可忽略,目标是使整个流程的总加权完成时间最小化.

1.2 符号

定义一些基本符号及决策变量,如表 1 所示. 定义参数如表 2 所示.

1.3 数学模型

根据以上描述及符号定义,将所研究的多阶段 FFSP 建立为整数线性规划模型. 由于实际生产中需要节省整个流程的时间,提高加工效率,所以将目标设为:

$$\min Z = \underset{\{X_{ki}\}}{\text{Minimize}} \sum_{i=1}^n w_i X_{si} . \tag{1}$$

满足以下约束:

表 1 基本符号及决策变量

Tab.1 The basic notation and decision variables

i	工件, $i \in I = \{1, 2, \dots, n\}$
k	阶段, $k \in K = \{1, 2, \dots, s\}$
j	时间单位, $j \in J = \{1, 2, \dots, T\}$, T 为完成时间的上界
b	批, $b \in l = \{1, 2, \dots, B\}$, B 为总批数
f	批内工件, $f \in I_b$ (第 b 批内的工件集)
X_{ki}	工件 i 在阶段 k 的完成时间
U_{kij}	0~1 变量,表示在时刻 j 工件 i 是否在阶段 k 上进行加工

表 2 参数

Tab.2 The parameters

w_i	工件 i 的权重
P_{ki}	工件 i 在阶段 k 的加工时间
r_i	工件 i 的释放时间
L_b	第 b 批工件的数量
N_{bf}	第 b 批的第 f 个工件
M_k	阶段 k 可用于加工的机器数
$T_{k,k+1}$	工件由阶段 k 到 $k+1$ 的运输时间
x	有批生产要求的阶段, $1 < x < s$
ST_{ki}	工件 i 在阶段 k 上的调整时间,当 $k = x$ 且 $i = N_{b_1}$ 时, $ST_{ki} > 0$, 否则为 0

$$X_{ki} + T_{k,k+1} + P_{k+1,i} \leq X_{k+1,i}, \quad \forall i,k; \tag{2}$$

$$X_{1i-P_1i} + 1 \geq r_i, \quad \forall i; \tag{3}$$

$$X_{x,N_{bf}} + P_{x,N_{b,f+1}} = X_{x,N_{b,f+1}}, \quad \forall b,f; \tag{4}$$

$$\sum_{i \in I} U_{kij} \leq M_k, \quad \forall k,j; \tag{5}$$

$$\sum_{j \in J} U_{kij} = P_{ki} + ST_{ki}, \quad \forall i,k. \tag{6}$$

约束(2)表示工件只有在完成前一阶段的加工且被运送到下一阶段,才能开始它的加工;约束(3)定义了工件动态到达第一阶段;约束(4)表示在批生产阶段同一批内的工件在一台机器上连续地进行加工;约束(5)确保在同一时刻进行加工的工件数不能超过该时刻可用的机器数;约束(6)表示工件占用机器的时间区间.

2 IAGA 求解算法设计

步骤 1 种群初始化. 对种群进行初始化,设置种群规模 $Popsiz e = 100$,遗传代数 $G = 200$. 采用实数编码方式,用随机产生的正整数对工件依次进行编号,初始阶段每个工件都有一个释放时间,先到达先加工,同时到达按照编号顺序

加工.

步骤 2 选择. 采用轮盘赌选择方式, 首先对种群内个体的适应度函数值进行评价, 令适应值函数 $f(i)$ 为总加权完成时间的倒数. $f(i)$ 越大, 说明经函数优化后的个体对应的解越好, 从而保留此个体.

步骤 3 交叉. 采用两点交叉方式. 因种群的更新效率是由交叉算子 p_c 决定的, 若 p_c 过大, 则优秀个体被破坏的可能性会增加; 反之, 则会使算法搜索能力变差. 所以动态设置 p_c , 即可以持续优化种群, 又能保留优秀个体. 在 Srinivas 等^[11]的研究基础上, 令

$$p_{cmax} = \begin{cases} 0.99, & g \leq G/2; \\ 0.6, & g > G/2, \end{cases} \quad (7)$$

式中: g 为当前迭代数; G 为总迭代数. 设置 p_c 如下:

$$p_c = \begin{cases} p_{cmax} - (p_{cmax} - p_{cmin}) \left(\frac{g}{G} + \frac{f_i}{f_{max} - \bar{f}} \right), & f_i \geq \bar{f}; \\ p_{cmax}, & f_i < \bar{f}. \end{cases} \quad (8)$$

式中: p_{cmin} 、 p_{cmax} 分别为进行交叉操作的最小和最大概率, 令 $p_{cmin} = 0.5$; f_{max} 为当前种群内所有个体最大的适应值; f_i 为个体的适应值; \bar{f} 为当前种群内所有个体的平均适应值.

步骤 4 变异. 采用单点变异方式. 虽然 GA 具有全局搜索能力, 但也可能陷入局部收敛. 有效设计遗传算子 p_m , 可以增加算法搜索的广度和深度. 将个体进行变异操作的实际概率 p_m 设置如下^[11]:

$$p_{mmin} = \begin{cases} 0.03, & g \leq G/2, \\ 0.01, & g > G/2, \end{cases} \quad (9)$$

$$p_m = \begin{cases} p_{mmin} + (p_{mmax} - p_{mmin}) \left(\frac{g}{G} + \frac{f_i}{f_{max} - \bar{f}} \right), & f_i \geq \bar{f}, \\ p_{mmin}, & f_i < \bar{f}, \end{cases} \quad (10)$$

式中: p_{mmin} 和 p_{mmax} 分别为进行变异操作的最小和最大概率, 令 $p_{mmax} = 0.05$.

3 试验测试

3.1 试验参数设置

本节分别对 GA 和 IAGA 进行相应设计, 并与 LR 算法进行对比, 在 Matlab 上测试其性能, 这 3 种算法均在 Celeron (R) Dual-Core CPU 2. 10

GHz 微机上运行. 公平起见, 将 IAGA 进行 200 次迭代所需时间作为停止条件, 设置最大运行时间为 1 000 s. 当工件规模 $n = \{ 30, 60, 90 \}$ 时, 对 3 种算法的目标值 (min Z) 和迭代次数 (ItN) 进行比较; 而当 $n = \{ 120, 150 \}$ 时, 由于 LR 运行一次迭代所需的计算时间较长致使在较短的计算时间内运行的迭代次数较少, 解的质量也较差, 随着问题规模增大这种情况也更明显, 因此未再测试 LR 求解这 2 种工件规模的情况, 仅对 IAGA 和 GA 的结果进行了比较.

令 GA 的交叉概率为 0. 8, 变异概率为 0. 05. 针对每种工件规模的 $\{ n, s, M_k \}$ 组合随机产生 10 个实例, 因此本次试验共测试了 300 组实例, 测试数据随机产生.

(1) 工件数 $n \in \{ 30, 60, 90, 120, 150 \}$; 阶段数 $s \in \{ 3, 4, 5 \}$; 机器数 $M_k \in \{ 3, 4 \}$;

(2) $x = 2, L_b = 5$;

(3) 工件的权重以及加工时间是满足 $[1, 10]$ 之间均匀分布的随机数;

(4) 工件动态到达第一阶段的时间、各阶段间的运输时间以及批生产阶段所需要的调整时间都是 $[3, 5]$ 之间均匀分布的随机数.

3.2 结果分析

表 3、表 4 为测试结果.

(1) 当工件规模和阶段数不变时, 随着可用机器数的增加, 3 种算法的目标函数值都会相应降低, 求解时间也会有所减少.

(2) 当 $n = 30$ 时, LR 得到的解的质量最好, 目标值比 IAGA 和 GA 分别改进了 10. 7% 和 13. 8%. 当 $n = 60$ 时, IAGA 的优势逐渐显现, 目标值相对 LR 改进了 8. 3%. 当 $n = 90$ 时, IAGA 求得的目标值比 LR 改进了 24. 0%. 由此可得, LR 求解小规模问题的效果最好, 而 IAGA 更适合求解贴近实际生产的中大规模问题.

(3) 当 $n = \{ 30, 60, 90, 120, 150 \}$ 时, IAGA 所求的目标值相比 GA 分别改进了 2. 8%、3. 3%、10. 4%、12. 5%、13. 1%. 即对于不同规模问题, IAGA 求解该问题的能力一致强于 GA, 且在大规模问题时这种优势更加明显.

(4) 在相同的时间内, IAGA 的迭代次数多于 GA. 当 $n = \{ 30, 60, 90, 120, 150 \}$ 时, IAGA 的迭代次数比 GA 分别多了 5、6、8、10、15. 因此, 随着问题规模增大, 两者迭代数的差额也越大, IAGA 的效率更高.

表 3 规模 $n = \{30, 60, 90\}$ 的测试结果对比
Tab.3 Testing results of $n = \{30, 60, 90\}$

n	M_i	s	min Z			ItN			CPU/s
			IAGA	GA	LR	IAGA	GA	LR	
30	3	3	9 886	10 244	8 351	200	196	10 895	210
		4	11 262	11 487	10 020	200	194	9 983	230
		5	12 283	12 471	11 529	200	194	9 985	264
	4	3	8 833	9 296	7 291	200	196	10 765	207
		4	9 859	10 373	8 918	200	193	9 768	224
		5	10 343	10 350	10 303	200	194	9 832	260
平均			10 411	10 703	9 402	200	195	10 204	232
60	3	3	23 536	24 438	27 771	200	196	90	390
		4	27 255	27 814	33 624	200	192	77	458
		5	30 937	31 803	33 980	200	193	74	543
	4	3	23 056	23 964	22 828	200	195	89	385
		4	26 496	27 401	27 196	200	193	78	455
		5	29 753	30 907	28 992	200	193	73	534
平均			26 838	27 721	29 065	200	194	80	460
90	3	3	54 108	63 072	73 964	200	192	4	590
		4	68 920	75 371	80 734	200	193	3	712
		5	75 781	79 915	86 177	200	192	3	841
	4	3	41 128	50 774	59 075	200	193	4	589
		4	53 443	58 495	61 023	200	191	3	707
		5	58 364	60 814	75 234	200	193	3	839
平均			58 624	64 740	72 701	200	192	3	713

表 4 规模 $n = \{120, 150\}$ 的测试结果对比
Tab.4 Testing results of $n = \{120, 150\}$

n	M_i	s	min Z		ItN		CPU/s
			IAGA	GA	IAGA	GA	
120	3	3	69 862	79 045	200	189	806
		4	80 014	89 136	200	188	985
		5	89 543	101 897	185	175	1 000
	4	3	66 108	75 714	200	190	785
		4	78 214	83 357	200	188	979
		5	82 176	95 065	186	177	1 000
平均		77 652	87 369	195	185	925	
150	3	3	109 798	116 102	196	182	1 000
		4	126 411	140 462	160	141	1 000
		5	145 357	160 047	135	120	1 000
	4	3	99 726	110 596	197	182	1 000
		4	114 261	136 408	164	150	1 000
		5	130 271	157 061	139	122	1 000
平均		120 970	136 779	165	150	1 000	

4 结论

针对钢铁行业的炼钢—连铸—热轧工艺流程,考虑工件的释放时间、运输时间以及机器的调整时间,建立了中间阶段有批处理特征的整

数线性规划模型,目标为所有工件的总加权完成时间最小. 针对该多阶段 FFSP,设计了遗传参数随迭代次数和适应值的改变而进行调节的自适应遗传算法. 对各种规模的问题进行仿真试验结果显示,相同时间内,无论何种工件规模,

IAGA 解的质量均明显优于 GA,且 IAGA 的迭代次数多于 GA;与 LR 算法相比,在求解较大规模工件时,IAGA 的性能更好,更符合实际生产对时间的要求。

参考文献:

- [1] 汪恭书,唐立新. 连铸—轧制生产中带有批决策的排序问题的建模与优化方法[J]. 自动化学报, 2012, 38(10): 1713–1720.
- [2] 洪悦,唐立新,张颜颜. 基于数据子空间 PLS 建模技术的热轧轧制力优化设定[J]. 控制与决策, 2014, 29(7): 1199–1204.
- [3] YU Y S, JIE J C, ZHANG S. Three-layer Al/Al-B₄C composite material prepared by casting and hot rolling[J]. Powder metallurgy and metal ceramics, 2015, 54(7/8): 390–396.
- [4] TAN Y, MONCH L, FOWLER J W. Proceedings of winter simulation conference[C]. 2015.
- [5] 陈成栋,陈华平,朱颀,等. 两阶段流水车间批调度问题的蚁群优化算法[J]. 计算机工程, 2012, 38(19): 137–141.
- [6] 黄锦钊,刘建军,陈庆新,等. 两机 flow-shop 类型模具热处理车间批调度算法[J]. 计算机集成制造系统, 2014, 20(7): 1665–1674.
- [7] 程八一,胡笑旋. 差异作业批调度的流水车间问题及近似算法[J]. 系统工程学报, 2011, 26(3): 393–399.
- [8] XUAN H, LI B. Scheduling dynamic hybrid flowshop with serial batching machines[J]. Journal of the operational research society, 2013(64): 825–832.

- [9] COSTA A, CAPPADONNA F A, FICHERA S. A novel genetic algorithm for the hybrid flow shop scheduling with parallel batching and eligibility constraints[J]. International journal of advanced manufacturing technology, 2014, 75(5/8): 833–847.
- [10] LI D, MENG X, LIANG Q. A heuristic-search genetic algorithm for multi-stage hybrid flowshop scheduling with single processing machines and batch processing machines[J]. Journal of intelligent manufacturing, 2015(26): 873–890.
- [11] SRINIVAS M, PATNAIL K L M. Adaptive probabilities of crossover and mutation in genetic algorithms[J]. IEEE transactions on system, man and cybernetics, 1994, 24(4): 656–667.
- [12] XU Y C, TAN W N. The modeling and simulation of flow shop scheduling problem based on adaptive genetic algorithm[J]. RISTI-Revista iberica de sistemas e tecnologias de informacao, 2016(17): 25–40.
- [13] SUN H M, YU J W, WANG H L. Proceedings of Chinese intelligent automation conference-intelligent technology and systems[C], 2015.
- [14] ZHU H M, ZHAO J F, WANG M L. Adaptive genetic algorithm for Hybrid Flow-Shop scheduling[C]. 3rd international conference on advanced engineering materials and technology, 2013.
- [15] ZHAO J F, ZHU X C, WANG B S. Hybrid flow-shop scheduling method and simulation based on adaptive genetic algorithm[C]. Applied mechanics and materials, 2014.

Multi-stage Flexible Flowshop Scheduling with Batching Machines

WANG Junyan, WANG Xueyuan, XUAN Hua

(School of Management Engineering, Zhengzhou University, Zhengzhou 450001, China)

Abstract: Based on steel making-continuous casting - hot rolling production process in iron and steel industry, the problem of scheduling n jobs in a multi-stage flexible flowshop with batching machines at some middle stage was studied. The batching production stage consisted of multiple serial batching machines in parallel, and the other stages contained discrete machines. Firstly, a mathematical model was formulated to minimize the total weighted completion time with the consideration of job dynamic arrival, transportation time between the adjacent stages and machine setup time. Then, an improved adaptive genetic algorithm was developed for this NP-hard problem where the genetic parameters were associated with the iteration number and the fitness function values. Computational experiments tested a large number of random data for up to 150 jobs. The results show that the proposed algorithm could find the better solutions within a shorter period of time, as compared with the general genetic algorithm. The comparison with Lagrangian relaxation showed that the improved genetic algorithm performed better on solution quality for medium and large sized problems.

Key words: flexible flowshop scheduling; batch machines; total weighted completion time; self-adaptive genetic algorithm; self-adaptive adjustment