Best Known Solutions for Taillard's Instances Adapted to the Distributed Blocking Flow Shop Scheduling Problem

Imma Ribas, Ramon Companys
Departament d'Organització d'Empreses. Universitat Politècnica de Catalunya, BarcelonaTech
Barcelona, Spain
imma.ribas@upc.edu, ramon.companys@upc.edu

Abstract— The distributed blocking flow shop scheduling problem (DBFSP) allows modeling the scheduling process in companies with more than one factory. It configures production systems as flow shop lines where the blocking constraint must be considered. To the best of our knowledge, this variant of the distributed permutation flow shop scheduling problem has not been studied. In this working paper, we show the best solutions found during our research for the Taillard's instances adapted to this problem.

Keywords—scheduling; heuristics; distributed blocking flow shop; blocking flow shop; distributed permutation flow shop

Introduction

Distributed manufacturing is a common situation for large enterprises that compete in a globalized market. Because of current globalization trends, production has shifted from single factory production to a multi-factory production network [1]. In this environment, the scheduling problems deal with the allocation of jobs to factories and the scheduling of jobs in each plant. Since the flow shop configuration is the most common processing layout, the flow shop scheduling problem has been studied greatly since the seminal paper of Johnson [2]. However, its extension to a multi-plant environment was first presented by Naderi and Ruiz [3], who referred to it as the Distributed Permutation Flow Shop Scheduling Problem (DPFSP). After the publication of [3], several authors proposed various heuristics to solve this problem ([4]–[13]), but the blocking constraint has been considered in none of them. The blocking flow shop scheduling problem allows many production systems to be modeled when there are no buffers between consecutive machines. In general, it is useful for those systems that have a production line without a drag system that forces a job to be transferred between two consecutive stations at pre-established times. Some industrial examples can be found in the iron and steel industry [14]; in the treatment of industrial waste and the manufacture of metallic parts [15]; or in a robotic cell, where a job may block a machine while waiting for the robot to pick it up and move it to the next stage [16]. The blocking constraint tends to increase the completion time of jobs, because the processed job cannot leave the machine if the next machine is busy. Therefore, the heuristics designed to schedule jobs in this environment have to consider this fact in order to minimize the idle time of machines due to possible blockage. Therefore, the distributed blocking flow shop scheduling problem (DBFSSP) deals with the allocation and scheduling of jobs in a multi-factory production network with the blocking constraint present in the manufacturing system. It is interesting to study this problem in order to design specific procedures, since the adaptation of those designed for the DPFSP probably perform worse than procedures which consider its characteristics.

In this working paper we show the best values found for the Taillard's benchmark [17] applied to the DBFSP.

Problem definition

The problem is defined as follows: n jobs have to be scheduled in one of the F identical factories. The production configuration of each factory is a flow shop consisting of m machines. Each factory is able to process all jobs. The jobs assigned to a factory have to be processed by all machines in the same order, from machine 1 to machine m. Each job i, $i \in \{1,2,...,n\}$ requires a fixed non-negative processing time $p_{j,i}$ on every machine j, $j \in \{1,2,...,m\}$, which does not change from factory to factory. Setup times are considered to be included in the processing time. The objective is to schedule the jobs to the different factories such that the maximum makespan (C_{max}) among factories is minimized. Hence, a plant f has a set of n_f, jobs to be sequenced in order to minimize the C_{max} of the plant. We denote σ_f as the sequence of the n_f jobs assigned to plant f. Therefore, a solution Π is formed by the sequence of jobs in each factory (Π=(σ_1, σ_2, …, σ_f)).

We denote [k,f] as the job which occupies position k in the sequence of σ_f, and f_{max} as the factory with the maximum makespan. Let $e_{j,k,f}$ be the time in which the job [k,f] starts to be processed in machine j, and $c_{j,k,f}$ be the departure time of job [k,f] in machine j. $C_{max} = C_{max}(\Pi)$ denotes the global makespan, i.e., the maximum completion time of the last job processed in any of the factory.

Therefore, according to this notation the problem can be formalized as follows:

$$Min\ max\{ c_{m,n_f,f} \} \qquad \forall j,k,f$$

$$e_{j,k,f} = Max\{c_{j,k-1,f}; c_{j-1,k,f}\} \qquad \forall j,k,f$$
$$c_{j,k,f} = Max\{c_{j,k-1,f}; e_{j,k,f} + p_{j,[k,f]}\} \qquad \forall j,k,f$$

with $c_{j,0} = 0$, $c_{0,k,f} = 0$, $c_{m+1,k,f} = 0$ $\quad \forall j,k,f$ being the initial conditions.

Best solutions for Taillard's instances

The Taillard's benchmark was generated to test algorithms for the permutation flow shop problem with makespan criterion, although they have also been used under other criteria and conditions. In particular, these instances were adapted to the DPFSP in [3] and used later in [18] and [9] to test their algorithms for the same problem. The benchmark is composed of 12 sets of 10 instances, ranging from 20 jobs and 5 machines to 500 jobs and 20 machines, where $n \in \{20, 50, 100, 200, 500\}$ and $m \in \{5, 10, 20\}$, although not all combinations of n and m are available. In particular, sets 200x5, 500x5 and 500x10 are missing. These 120 instances were augmented with six values of $F \in \{2, 3, 4, 5, 6, 7\}$.

The next table shows the best makespan value known for each instance.

num	*n*	*m*	F					
			2	3	4	5	6	7
1	20	5	771	583	493	440	409	384
2	20	5	793	589	496	438	404	381
3	20	5	704	522	446	398	373	360
4	20	5	805	616	521	469	432	413
5	20	5	758	573	489	438	406	385
6	20	5	745	565	483	436	404	383
7	20	5	757	568	478	439	430	430
8	20	5	758	568	489	440	412	386
9	20	5	767	577	478	427	396	376
10	20	5	699	513	434	392	365	347
11	20	10	1085	888	789	732	693	670
12	20	10	1155	941	840	781	735	707
13	20	10	1044	845	754	700	673	650
14	20	10	969	778	689	635	605	586
15	20	10	1005	817	727	671	650	628
16	20	10	978	786	695	648	613	591
17	20	10	1019	832	743	696	671	671
18	20	10	1075	875	773	720	692	692
19	20	10	1070	860	769	713	702	702
20	20	10	1127	915	808	760	723	707
21	20	20	1715	1482	1363	1305	1257	1237
22	20	20	1602	1391	1297	1235	1191	1191
23	20	20	1772	1528	1405	1347	1320	1320
24	20	20	1665	1443	1354	1297	1259	1239
25	20	20	1726	1490	1377	1309	1280	1253
26	20	20	1687	1468	1356	1286	1256	1256
27	20	20	1701	1474	1362	1297	1251	1232
28	20	20	1648	1422	1321	1258	1232	1227
29	20	20	1699	1483	1369	1302	1257	1240
30	20	20	1638	1416	1302	1240	1201	1166
31	50	5	1563	1076	828	688	603	536
32	50	5	1657	1143	887	738	648	582
33	50	5	1565	1072	837	699	609	543
34	50	5	1633	1135	882	736	642	578
35	50	5	1648	1129	882	736	636	572
36	50	5	1649	1145	897	749	657	591
37	50	5	1589	1109	872	730	640	575
38	50	5	1582	1092	856	715	622	560
39	50	5	1511	1052	825	688	602	540
40	50	5	1616	1116	873	727	631	565
41	50	10	1985	1452	1190	1041	940	870
42	50	10	1910	1394	1143	1003	904	844
43	50	10	1919	1399	1146	1009	921	856
44	50	10	1989	1450	1188	1042	948	882
45	50	10	1976	1437	1190	1042	947	879
46	50	10	1978	1432	1181	1036	935	869
47	50	10	2038	1488	1227	1069	973	901
48	50	10	1981	1443	1195	1045	948	875
49	50	10	1931	1404	1148	1008	911	844
50	50	10	2005	1474	1220	1067	971	902

num	*n*	*m*	F					
			2	**3**	**4**	**5**	**6**	**7**
51	50	20	2736	2147	1864	1692	1582	1502
52	50	20	2597	2043	1761	1610	1501	1423
53	50	20	2598	2045	1765	1603	1494	1419
54	50	20	2641	2073	1797	1638	1531	1452
55	50	20	2589	2042	1765	1609	1504	1428
56	50	20	2586	2043	1769	1602	1495	1426
57	50	20	2608	2047	1766	1603	1497	1417
58	50	20	2613	2063	1791	1624	1515	1436
59	50	20	2653	2094	1820	1652	1539	1468
60	50	20	2667	2097	1808	1644	1531	1453
61	100	5	3138	2135	1628	1331	1127	982
62	100	5	3088	2091	1594	1302	1101	963
63	100	5	3018	2057	1568	1273	1082	942
64	100	5	2941	1993	1519	1236	1049	912
65	100	5	3054	2067	1579	1278	1093	954
66	100	5	2969	2021	1533	1248	1059	924
67	100	5	3070	2074	1586	1287	1095	957
68	100	5	2985	2025	1541	1250	1056	922
69	100	5	3119	2125	1621	1328	1125	985
70	100	5	3122	2113	1611	1309	1109	967
71	100	10	3677	2549	2004	1687	1469	1323
72	100	10	3558	2449	1911	1595	1384	1239
73	100	10	3614	2496	1952	1638	1422	1277
74	100	10	3775	2616	2051	1718	1502	1353
75	100	10	3584	2481	1953	1632	1427	1280
76	100	10	3502	2420	1877	1568	1364	1222
77	100	10	3581	2476	1934	1608	1407	1263
78	100	10	3594	2514	1967	1645	1437	1292
79	100	10	3712	2579	2022	1692	1475	1319
80	100	10	3656	2543	1991	1666	1461	1314
81	100	20	4350	3183	2606	2278	2043	1891
82	100	20	4351	3167	2589	2250	2037	1881
83	100	20	4342	3163	2608	2272	2051	1897
84	100	20	4351	3173	2605	2261	2043	1891
85	100	20	4355	3175	2605	2269	2051	1900
86	100	20	4407	3236	2654	2319	2094	1934
87	100	20	4410	3197	2636	2294	2070	1908
88	100	20	4465	3264	2695	2353	2125	1963
89	100	20	4398	3230	2650	2304	2084	1931
90	100	20	4421	3243	2661	2317	2097	1934
91	200	10	6902	4719	3605	2956	2499	2197
92	200	10	6831	4639	3576	2907	2483	2170
93	200	10	6848	4674	3593	2929	2496	2185
94	200	10	6830	4675	3559	2915	2480	2170
95	200	10	6830	4674	3585	2921	2481	2169
96	200	10	6739	4585	3514	2863	2444	2137
97	200	10	6961	4759	3644	2987	2534	2220
98	200	10	6909	4698	3603	2946	2504	2190
99	200	10	6842	4657	3560	2915	2479	2169
100	200	10	6867	4676	3586	2917	2492	2175

num	n	m	F					
			2	3	4	5	6	7
101	200	20	7820	5487	4314	3608	3151	2824
102	200	20	7929	5533	4336	3644	3183	2855
103	200	20	8000	5610	4402	3682	3212	2871
104	200	20	7948	5556	4373	3645	3183	2854
105	200	20	7859	5484	4297	3599	3139	2809
106	200	20	7940	5519	4321	3611	3136	2819
107	200	20	7944	5554	4370	3659	3203	2869
108	200	20	7965	5574	4377	3660	3202	2864
109	200	20	7908	5540	4349	3635	3172	2845
110	200	20	7951	5538	4367	3651	3202	2863
111	500	20	18399	12525	9619	7869	6704	5849
112	500	20	18541	12670	9724	7959	6766	5919
113	500	20	18344	12517	9609	7863	6701	5860
114	500	20	18469	12610	9688	7908	6742	5888
115	500	20	18374	12490	9575	7856	6667	5831
116	500	20	18494	12618	9655	7893	6699	5889
117	500	20	18348	12510	9595	7845	6665	5806
118	500	20	18399	12595	9643	7883	6700	5881
119	500	20	18313	12480	9559	7857	6647	5798
120	500	20	18496	12597	9680	7911	6713	5872

Best solutions for the DBFSP

References

[1] J. Behnamian and S. M. T. Fatemi Ghomi, "A survey of multi-factory scheduling," *J. Intell. Manuf.*, no. October 2012, pp. 1–19, 2014.

[2] S. M. Johnson, "Optimal two-and three-stage production schedules with set up times included," *Nav. Res. Logist. Q.*, vol. 1, pp. 61–68, 1954.

[3] B. Naderi and R. Ruiz, "The distributed permutation flowshop scheduling problem," *Comput. Oper. Res.*, vol. 37, no. 4, pp. 754–768, Apr. 2010.

[4] J. Gao and R. Chen, "A hybrid genetic algorithm for the distributed permutation flowshop scheduling problem," Mar. 2012.

[5] J. Gao, R. Chen, and Y. Liu, "A Knowledge-based Genetic Algorithm for Permutation Flowshop Scheduling Problems with Multiple Factories," *Int. J. Adv. Comput. Technol.*, vol. 4, no. 7, pp. 121–129, 2012.

[6] S.-W. Lin, K.-C. Ying, and C.-Y. Huang, "Minimising makespan in distributed permutation flowshops using a modified iterated greedy algorithm," *Int. J. Prod. Res.*, vol. 51, no. 16, pp. 5029–5038, Aug. 2013.

[7] H. Liu and L. Gao, "A discrete electromagnetism-like mechanism algorithm for solving distributed permutation flowshop scheduling problem," *Proc. - 2010 Int. Conf. Manuf. Autom. ICMA 2010*, pp. 156–163, 2010.

[8] V. Fernandez-Viagas and J. M. Framinan, "A bounded-search iterated greedy algorithm for the distributed permutation flowshop scheduling problem," *Int. J. Prod. Res.*, vol. 53, no. 4, pp. 1111–1123, Aug. 2014.

[9] B. Naderi and R. Ruiz, "A scatter search algorithm for the distributed permutation

flowshop scheduling problem," *Eur. J. Oper. Res.*, vol. 239, no. 2, pp. 323–334, Dec. 2014.

[10] S. Wang, L. Wang, M. Liu, and Y. Xu, "An effective estimation of distribution algorithm for solving the distributed permutation flow-shop scheduling problem," *Int. J. Prod. Econ.*, vol. 145, no. 1, pp. 387–396, Sep. 2013.

[11] Y. Xu, L. Wang, S. Wang, and M. Liu, "An effective hybrid immune algorithm for solving the distributed permutation flow-shop scheduling problem," *Eng. Optim.*, vol. 46, no. 9, pp. 1269–1283, Oct. 2013.

[12] J. Gao, R. Chen, and W. Deng, "An efficient tabu search algorithm for the distributed permutation flowshop scheduling problem," *Int. J. Prod. Res.*, vol. 51, no. 3, pp. 641–651, Feb. 2013.

[13] J. Gao, R. Chen, W. Deng, and Y. Liu, "Solving multi-factory flowshop problems with a novel variable neighbourhood descent algorithm," *J. Comput. Inf. Syst.*, vol. 8, no. 5, pp. 2025–2032, 2012.

[14] H. Gong, L. Tang, and C. W. Duin, "A two-stage flow shop scheduling problem on a batching machine and a discrete machine with blocking and shared setup times," *Disrupt. Manag.*, vol. 37, no. 5, pp. 960–969, 2010.

[15] S. Martinez, S. Dauzère-Pérès, C. Guéret, Y. Mati, and N. Sauer, "Complexity of flowshop scheduling problems with a new blocking constraint," *Eur. J. Oper. Res.*, vol. 169, no. 3, pp. 855–864, 2006.

[16] S. P. Sethi, C. Sriskandarajah, G. Sorger, J. Blazewicz, and W. Kubiak, "Sequencing of parts and robot moves in a robotic cell," *Int. J. Flex. Manuf. Syst.*, vol. 4, pp. 331–358, 1992.

[17] E. Taillard, "Benchmarks for basic scheduling problems," *Eur. J. Oper. Res.*, vol. 64, no. 2, pp. 278–285, 1993.

[18] V. Fernandez-Viagas and J. M. Framinan, "A bounded-search iterated greedy algorithm for the distributed permutation flowshop scheduling problem," *Int. J. Prod. Res.*, vol. 53, no. 4, pp. 1111–1123, Aug. 2014.