J Sched (2006) 9: 389–396
DOI 10.1007/s10951-006-7042-y

A note on the complexity of the concurrent open shop problem

Thomas A. Roemer

Abstract The concurrent open shop problem is a relaxation of the well known open job shop problem, where the components of a job can be processed in parallel by dedicated, component specific machines. Recently, the problem has attracted the attention of a number of researchers. In particular, Leung et al. (2005) show, contrary to the assertion in Wagneur and Sriskandarajah (1993), that the problem of minimizing the average job completion time is not necessarily strongly NP-hard. Their finding has thus once again opened up the question of the problem's complexity. This paper re-establishes that, even for two machines, the problem is NP-hard in the strong sense.

Keywords Order scheduling · Total completion time · Complexity · Internal and external performance measures

1. Introduction

In the concurrent job shop, a job consists of up to m different components to be processed on a specific one of m dedicated machines. Components are independent of each other, such that components of the same job can be processed in parallel on different machines. A job is completed once all of its components are completed. Leung, Li and Pinedo (2005a), refer to this scenario as the parallel machine environment with m fully dedicated machines and denote it as PDm in the notation by Graham et al. (1979). I will follow this notation, but denote the problem simply by PD if m is arbitrary. Thus, for example, $PD4|r_j|\Sigma w_j T_j$ designates the concurrent job shop problem with 4 machines, job release dates r_j, and the objective of minimizing total weighted tardiness. Notice that $PD1$ simplifies to the single machine problem, for which well established results can be found in standard texts (i.e. Baker, 1974; Morton and Pentico, 1993; Pinedo, 2002)

Occurrences of the concurrent open shop problem are abundant (Leung, Li and Pinedo, 2005b) and include airplane maintenance (Wagneur and Sriskandarajah, 1993), fire engine assembly

T. A. Roemer (✉)
Leaders for Manufacturing Program & Sloan School of Management,
Massachusetts Institute of Technology, Cambridge, MA 02142
e-mail: troemer@mit. edu

(Lee, Cheng, and Lin, 1993), just-in-time systems (Potts et al., 1995), supply chain assembly systems (Chen and Hall, 2000), paper processing (Leung, Li, and Pinedo, 2005a), accounting services (Ahmadi, Bagchi, and Roemer, 2005) and part kitting (Yang and Posner, 2005). Despite this prevalence throughout industries, the discussion of *PD* has, at least until recently, been somewhat subdued. With the exception of Ng, Cheng and Yuan (2003), who discuss minimizing the weighted number of tardy jobs ($PD\|\Sigma w_j U_j$,), all other authors focus primarily on minimizing the average (weighted) completion time ($PD\|\Sigma w_j C_j$). The primary focus of this paper is also on the latter problem. In particular, I will show that even the simplest version of this problem, namely $PD2\|\Sigma C_j$ is strongly NP-hard. Before providing a concise proof, the next section will outline existing complexity results in the literature. Unfortunately, much of the literature seems to have evolved concurrently, but in isolation, leading to both redundancies in efforts and some confusion regarding the complexity of $PD\|\Sigma w_j C_j$ in particular. It is hoped that this paper serves three purposes: First, to provide a comprehensive, yet brief overview of the evolution of complexity results for *PD*; second, to firmly establish the complexity of $PD\|\Sigma w_j C_j$; and third, to introduce a group of new problems within the framework of *PD* that deserve additional attention from interested researchers.

2. Literature review

To the best of my knowledge, $PD\|\Sigma w_j C_j$ was first introduced by Ahmadi and Bagchi (1990), who referred to it as the *Problem of Scheduling Customer Orders*. In their 1993 working paper, these authors establish unary NP-hardness for $PD2\|\Sigma w_j C_j$. Since then, discussions of *PD* have been sporadic and isolated. Wagneur and Sriskandarajah (1993) refer to *PD* as *Openshops with Jobs Overlap*. Based on a reduction from *Numerical Matching with Target Sums*, they show that even for *PD*2, minimizing the average completion time (ΣC_j) and tardiness (ΣT_j) are strongly NP-hard. They further show that minimizing the number of late jobs (ΣU_j) is binary NP-hard for two machines and provide polynomial time algorithms for minimizing the maximum lateness ($L_{\max}$), makespan ($C_{\max}$), and the maximum tardiness ($T_{\max}$) for *PD*. In a predecessor to this paper, Roemer and Ahmadi (1997a and b) also establish unary NP-hardness for $PD2\|\Sigma C_j$. In their updated working paper, Ahmadi and Bagchi (1997) introduce polynomial and pseudo-polynomial procedures for $PD\|T_{\max}(C_{\max}, L_{\max})$ and $PD\text{m}\|w_j U_j$ respectively. Sung and Yoon (1998) establish that $PD2\|\Sigma w_j C_j$ is unary NP-hard based on a reduction from *3 Partition*, which is identical to that in Ahmadi and Bagchi (1993). In the context of a manufacturer with several suppliers, Chen and Hall (2000) establish unary NP-hardness for $PD4\|\Sigma C_j(\Sigma T_j)$, binary NP-hardness for $PD2\|\Sigma U_j$, provide polynomial algorithms for $PD\|C_{\max}(L_{\max})$ and a pseudo-polynomial procedure for $PD\text{m}\|w_j U_j$. Ng, Cheng and Yuan (2003) establish that $PD\|\Sigma U_j$ is strongly NP-hard. Throughout the literature, almost all of the proposed heuristics for $PD\|\Sigma w_j C_j$ are *m*-approximation algorithms. Notable exceptions are the linear programming relaxation based $2-$, $\frac{16}{3}-$, and 2-approximations by Chen and Hall (2000), Wang and Cheng (2003), and Leung, Li, and Pinedo (2005b), respectively. Ahmadi, Bagchi, and Roemer (2006) show that for $PD2\|\Sigma w_j C_j$, the relative worst case performance of the *Weighted Shortest Processing Time* heuristic is tightly bound by $\frac{\sqrt{5}+3}{\sqrt{5}+1}$.[1]

Figure 1 gives an overview of the evolving literature stream of the concurrent open shop problem and exhibits its relative fragmentation. Leung, Li and Pinedo (2005c) provide a more

[1] Interestingly, the value of this bound is identical to the celebrated *Golden Ratio* Φ, which occurs throughout nature and the arts.

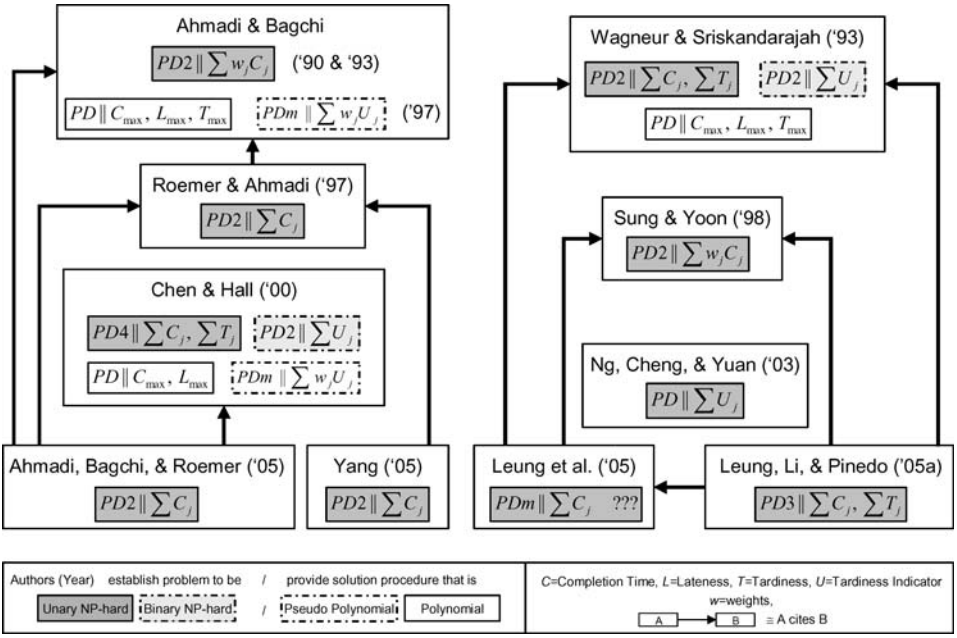

Fig. 1 Literature overview

general overview of the by now substantial literature on order scheduling models, but seem to be largely unaware of the research on the left hand side in Fig. 1. Due to this fragmentation, complexity results have been somewhat redundant. Indeed, with the exception of the complexity results relating to the number of tardy jobs by Ahmadi and Bagchi (1997), Chen and Hall (2000) and Ng, Chen, and Yuan (2003), every single complexity result was already contained in the early paper by Wagneur and Sriskandarajah (1993). Unfortunately, it seems that none of the authors working on *PD* (including myself) was aware of their comprehensive discussion of *PD*, until Leung et al. (2005) showed that the strong unary NP-hardness result for $PD2\|\Sigma C_j$ in Wagneur and Sriskandarajah (1993) was based on an erroneous proof. With the complexity of $PD\mathrm{m}\|\Sigma C_j$ seemingly open again, Leung, Li and Pinedo (2005a) showed, based on a reduction from *Numerical Matching with Target Sums*, that $PD3\|\Sigma C_j$ is strongly NP-hard. Thus, despite the early (unpublished) result by Roemer and Ahmadi (1997a, b), the complexity of $PD2\|\Sigma C_j$ was once again an open question. While Ahmadi, Bagchi and Roemer (2005) also address this question, the next section provides a concise and quite intuitive proof that $PD2\|\Sigma C_j$ is indeed unary NP-hard.

Quite recently, Yang (2005) provided an alternative proof for the same problem that bears some similarities to the proof presented here. Both approaches are based on a reduction from 3-Partition, and share that any triplet from a specific subset of jobs (referred to in this paper as *genuine jobs*) will on one machine exceed the size of the 3-Partition by the same amount that it falls short of it on the other machine. Beyond this principal breakthrough approach to the problem, which was first introduced by Roemer and Ahmadi (1997a), the proofs are quite different. Yang provides an elegant way of rendering the objective function value independent of the relative sequence of the genuine jobs. In contrast, this proof forces a sufficiently large number of *filler jobs* between triplets of genuine jobs, thereby compensating for the impact of different scheduling sequences on the objective function value. The reader, who is not merely interested in the complexity result itself, but also in its deduction, is urged to read and compare the two proofs, as both provide interesting techniques for reductions from 3-Partition.

3. The complexity of $PD\mathrm{m}\|\Sigma C_j$

Theorem 1. *$PDm\|\Sigma C_j$ is unary NP-hard for any $m \geq 2$.*

To prove the theorem, we use a reduction from the *3-Partition Problem* (3*PP*), defined as follows: Given nonnegative integers $b_1, b_2, \dots b_{3t}, b$, such that $b_j \leq b \,\forall j$ and $\sum_{j=1}^{3t} b_j = bt$, does there exist a partition $(T_1, T_2, \dots, T_t)$ of $T = \{1, \dots, 3t\}$, such that $|T_h| = 3$ and $\sum_{j \in T_h} b_j = b$ for $h = 1, 2, \dots, t$? To reduce the amount of subsequent notation, we assume without loss of generality that $\frac{b}{3}$ is an integer number. Given such an instance of $3PP$, an instance of the decision version of $PD2\|\Sigma C_j$ is constructed as follows: There are $3t$ *genuine jobs* with processing times $p_{j,1} = \frac{B+b}{3} - \delta_j$ and $p_{j,2} = b_j$ where $\delta_j = b_j - \frac{b}{3}$ and $B = 12bt + 3t$. Notice that for the processing times of the genuine jobs, there exists a 3-Partition of size $B + b$ on machine 1 and of size b on machine 2, if and only if a solution to 3*PP* exists. The remaining jobs are *filler jobs* from $2t - 1$ sets $F_s (s = 1, 2, \dots, 2t - 1)$. The processing times of each job in set F_s are $p_{j,1} = b + \lfloor \frac{s}{2} \rfloor$ and $p_{j,2} = b + \lceil \frac{s}{2} \rceil$. *Odd set* $F_{2s-1} (s = 1, 2, \dots t)$ contains B *odd jobs* and *even set* $F_{2s} (s = 1, 2, \dots t - 1)$ contains $A = 3t \cdot (Bb + Bt + b) + B$ *even jobs*.

Before defining the threshold value y, I will first explore some general properties of the optimal solution structure for this problem instance. It is easy to see, and ubiquitously observed in the literature, that there is an optimal permutation schedule where job J_h precedes job J_k if $p_{h,1} \leq p_{k,1}$ and $p_{h,2} \leq p_{k,2}$. Similarly, if equality holds in both cases, then there is an optimal schedule where jobs J_h and J_k are adjacent. Consequently, the relative order of the filler jobs is fixed, and the sets are scheduled in increasing order of their indices. Moreover, all jobs within an even set can be scheduled consecutively, such that no genuine job is scheduled between two even jobs from the same set. Figure 2 shows such a schedule.

Notice that in Fig. 2, exactly 3 genuine jobs are scheduled "within" each odd set F_{2s-1} and that, in the instance depicted, a 3-Partition exists. It is easy to see that, in this schedule, the two components from the first job from set F_s start concurrently at time

$$\tau_s = b \cdot \left\lfloor \frac{s}{2} \right\rfloor + B \cdot \sum_{k=1}^{\lfloor \frac{s}{2} \rfloor} (b + k) + A \cdot \sum_{k=1}^{\lceil \frac{s}{2} \rceil - 1} (b + k) \tag{1}$$

and that each set F_s contributes $K_s^* = \sum_{j=1}^{|F_s|} (\tau_s + j \cdot (b + \lceil \frac{s}{2} \rceil))$ to the objective function value. Moreover, each triplet of genuine jobs scheduled "within" odd set F_s is completed by time τ_{s+1} and hence the contribution K_G of all genuine jobs to the objective function is bounded by $3 \sum_{s=1}^{t} \tau_{2s} \leq K_G$. Consequently, if a solution to $3PP$ exists, then there must be a solution to the problem whose objective function value, say y^{yes}, satisfies $y^{\text{yes}} \leq \sum_{s=1}^{2t-1} K_s^* + 3 \sum_{s=1}^{t} \tau_{2s} \equiv y$.

It remains to show that the threshold value y must be exceeded if no solution to 3*PP* exists. The principal idea of the proof is the following. Even if the answer to 3*PP* is no, it is still optimal to schedule exactly 3 genuine jobs "within" each odd set, but this will delay the completion times of *all* even jobs from at least one even set by at least one unit of time. Choosing A large enough thus forces the objective function value to exceed y. To proceed, the following result must first be established.

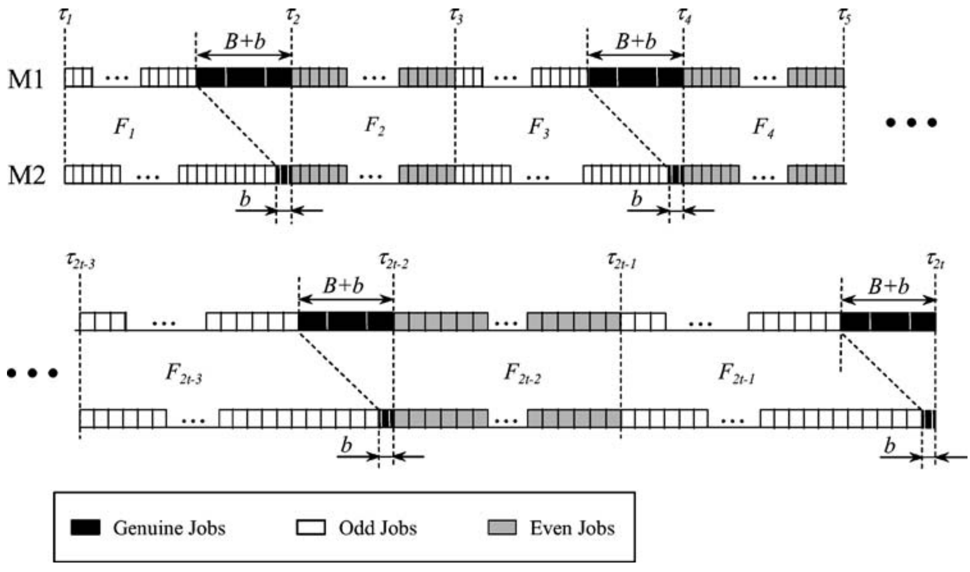

Fig. 2 A schedule with a 3-partition for the genuine jobs

Proposition 2. *Define a genuine job to be scheduled within odd set F_{2s-1} if it is adjacent to a job from set F_{2s-1} or to a genuine job that is scheduled within set F_{2s-1}. In any optimal schedule, exactly three genuine jobs are scheduled within each odd set $F_{2s-1}(s = 1, 2, \ldots, t)$.*

Proof: Let the genuine jobs be indexed according to their sequence in a given schedule. If genuine job $h \le 3t$ is scheduled within set F_{2s-1} then the difference between the completion times of the components of job J_h on the two machines is

$$C_{h,1} - C_{h,2} = \sum_{j=1}^{h} p_{j,1} - \sum_{j=1}^{h} p_{j,2} - B \cdot (s-1) - \beta \tag{2}$$

where $\beta \in [0, B]$ is the number of jobs from set F_{2s-1} scheduled before job h. Let job h be the first genuine job scheduled within set F_{2s-1} and suppose $h < 3\,(s-1)$ for some $s \in \{2, \ldots t\}$. In that case $C_{h,2} > C_{h,1}$ and, the predecessor of job J_h, say job $J_{\underline{h}} \in F_{2s-1} \cup F_{2s-2}$, completes at $C_{\underline{h}} = C_{\underline{h},2} = C_{h,2} - p_{h,2}$. Exchanging the two jobs reduces the completion time of job J_h to $C'_h \le C_{h,2} - b - s + 1 < C_{h,2} - p_{h,2} = C_{\underline{h}}$ and the schedule cannot be optimal. Conversely, let job h be the last genuine job scheduled within set F_{2s-1} and suppose that $h > 3s$ for some $s \in \{1, 2, \ldots, t-1\}$. In that case $C_{h,1} - C_{h,2} > 4bt + t - 4b \cdot (s + \frac{1}{3})$ and exchanging job J_h with its immediate successor, say job $J_{\overline{h}} \in F_{2s-1} \cup F_{2s}$, yields

$$C_{\bar{h}} \le \max \begin{bmatrix} C_{h,1} - p_{h,1}, \\ C_{h,1} - 4bt - t + 4b \cdot \left(s + \dfrac{1}{3}\right) - p_{h,2} \end{bmatrix} + b + s < C_{h,1} = C_h \tag{3}$$

and the schedule cannot be optimal. □

Given this result, the contribution K_G of the genuine jobs is easily seen to be strictly bounded by $3\sum_{s=1}^{t}\tau_{2s-1} < K_G$. More importantly though, the starting times of even set F_{2s} on machines 1 and 2 are now given by $\tau_{2s} - \Delta_{2s}$ and $\tau_{2s} + \Delta_{2s}$ respectively, where $\Delta_{2s} \equiv \sum_{j=1}^{3s}\delta_j$. The contribution K_{2s} of set F_{2s} to the objective function thus becomes $K_{2s} = K_{2s}^* + |\Delta_{2s}| \cdot A$. Clearly, if there is no solution to $3PP$, then $|\Delta_{2s}|$ cannot vanish for all s and a sufficiently large choice for A will force the objective function to exceed y. To see that this is indeed true, we still need to establish a lower bound on the contribution K_{2s+1} from odd set F_{2s+1}:

$$K_{2s+1} \geq \sum_{j=1}^{B}(\tau_{2s+1} - |\Delta_{2s}| + j \cdot (b + s + 1)) = K_{2s+1}^* - |\Delta_{2s}| \cdot B \tag{4}$$

Summing up the three lower bounds for the objective function value if $3PP$ does not have a solution, say y^{no}, yields

$$\begin{aligned} y^{\text{no}} &> y - 3\sum_{s=1}^{t}(\tau_{2s} - \tau_{2s-1}) + |\Delta_{2s}| \cdot \sum_{s=1}^{t-1}(A - B) \\ &\geq y - 3t \cdot (Bb + Bt + b) + A - B = y \end{aligned}$$

which completes the proof.

4. Conclusion

This paper established that minimizing average completion time in the concurrent job shop with two machines (i.e. $PD2\|\Sigma C_j$) is strongly NP-hard. A trivial corollary to this result is that $PD2\|\Sigma T_j$ is also strongly NP-hard, as $PD2|d_j = 0|\Sigma T_j$ and $PD2\|\Sigma C_j$ are equivalent problems. Given this result and the existing complexity results in the literature, the complexity of *PD* is now firmly established for all typically employed (regular) performance measures. Moreover, the extant *PD* literature provides many proven heuristics, that provide near optimal or very good solutions with reasonable computational effort.

However, basically all measures studied so far involve only *external measures* that aim at optimizing some measurement as perceived by the customer such as delivery times or tardiness. In the context of *PD, internal* or *flow-type measures* of how long (finished) customer orders, and thus inventory, stay within the confines of the manufacturing facility can also be of paramount importance, especially when coupled with external measures. For example, one of the most persistent operational challenges at Dell is to manage storage of partially filled orders, waiting for remaining items to be completed. To avoid this problem altogether, Amazon provides partial shipments of orders at substantial additional costs. An objective function that addresses this type of problem would be the minimization of $\sum_j \sum_i w_{i,j}(C_j - C_{i,j})^2$. Clearly, if no external measure is addressed, then this problem is trivial, since inserting sufficient idle time between orders would always yield an objective function value of 0. Ahmadi, Bagchi and Roemer (2005) introduce problem $PD\|\sum_j \sum_i w_j(C_j - C_{i,j})$ s.t. $C_i = C_i^*$, where C_i^* is the optimal completion time

[2] Notice that, to address the impact on inventory, the weights are now item specific and no longer just order or customer specific.

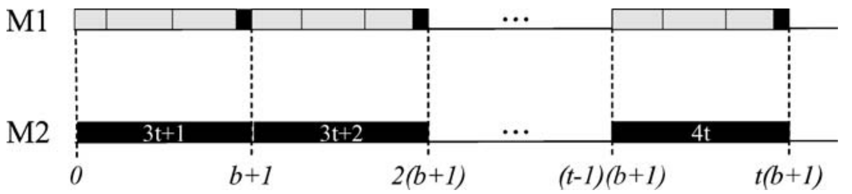

Fig. 3 A simple 3-partition schedule

obtained from solving *PD* for an external measure. They show that for given C_i^* the problem is equivalent to *m Single Machine Weighted Sum of Completion Time Problems with Release Times*, which is unary NP-hard even if all weights are equal. An alternative restriction is $C_{\max} = \max_i \sum_j p_{i,j}$, which warrants that no bottleneck machine is ever idle. If only the time that an order spends in the system matters, then under the same restriction, $\sum_j w_j(C_j - \min_j(C_{i,j} - p_{i,j}))$ is an appropriate measure. Such a scenario arises if considerable overhead costs, such as continued customer interactions or change requests, are prevalent once an order has been started. As the next proposition shows, both problems are unary NP-hard for any $m \geq 2$.

Proposition 3. *For the restriction* $C_{\max} = \max_i \sum_j p_{i,j}$*, problem* $PDm\|\sum_i(C_i - \min_j C_{i,j})$ *and problem* $PDm\|\sum_j \sum_i(C_i - C_{i,j})$ *are unary NP-hard for any* $m \geq 2$.

Proof: To prove the proposition, we use the following reduction from the *3-Partition Problem* (*3PP*) as defined above. Let there be $4t$ jobs, with $p_{j,1} = b_j$ and $p_{j,2} = 0$ for $j = 1, \ldots, 3t$, and $p_{j,1} = 1$, and $p_{j,2} = b + 1$ for $j = 3t + 1, \ldots, 4t$. The answer to 3*PP* is yes, if and only if the objective function value for the instance is $y = 0$. Suppose a 3-Partition exists. In that case the jobs can be scheduled as in Fig. 3 and it is easy to see that the objective function value is 0. Conversely, if no 3-Partition exists then $C_{i,1} \neq C_{i,2}$ for some $i > 3t$ and hence $y > 0$. □

Ahmadi and Bagchi (1997) suggest minimizing total weighted earliness, subject to meeting due dates, $PD|C_i \leq d_i|\sum_j w_j(d_j - C_j)$ and show that this problem is also unary NP-hard when all weights are equal. This problem is relevant if customers refuse to accept tardy orders or if transportation dates are scheduled in advance, such as for hazardous or oversized products.

Objective	Restrictions	Machines m	Complexity	Source: Complexity	Source: Solution Procedures
$L_{max}, C_{max}, T_{max}$		Arbitrary m	Polynomial	[3], [6], [20]	
ΣU_j		Any fixed m	Binary NP-Hard only	[6], [20]	
$\Sigma w_j U_j$				[3], [6]	
ΣU_j		Arbitrary m	Unary NP-Hard	[14]*	
$\Sigma C_j, \Sigma T_j$		m≥2		here, [22]	[1]-[4], [6], [9], [10], [20]*
$\Sigma_j (C_j - \min_i C_{i,j})$, $\Sigma\Sigma_j (C_j - C_{i,j})$	$C_{max} = \min_i \Sigma_j p_{i,j}$			here	None
$\Sigma_i\Sigma_j (C_j - C_{i,j})$	$C_j = C_j^*$			[4]	
$\Sigma_i\Sigma_j (d_j - C_j)$	$C_j \leq d_j$			[3]	

*Solution procedures refer to weighted problem

Fig. 4 Complexity results for the concurrent job shop problem

Given the relevance and complexity of these internal measures, a need for suitable heuristics arises. While external measures have been discussed in detail, work on internal measures remains sparse and offers ample opportunities to enhance the state of the art. Figure 4 provides a guide to the extant work and suggests further areas for exploration.

References

Ahmadi, R. H. and U. Bagchi, "Scheduling of multi-job customer orders in multi-machine environments," ORSA/TIMS, Philadelphia, 1990.

Ahmadi, R. H. and U. Bagchi, "Coordinated scheduling of customer orders," *Working Paper, Anderson School at UCLA* (1993).

Ahmadi, R. H. and U. Bagchi, "Coordinated Scheduling of Customer Orders," updated *Working Paper, Anderson School at UCLA* (1997).

Ahmadi, R. H., U. Bagchi, and T.A. Roemer, "Coordinated scheduling of customer orders for quick response," forthcoming in *Naval Research Logistics* (2006).

Baker, K. R., *Introduction to sequencing and scheduling*, Wiley, New York 1974.

Chen Z. L. and N. G. Hall, "Supply chain scheduling: Assembly systems," *Working Paper, Department of Systems Engineering, University of Pennsylvania* 2000.

Graham, R. L., E. L. Lawler, J. K. Lenstra, and A. H. G. Rinnooy Kan, "Optimization and Approximation in Deterministic Sequencing and Scheduling: A Survey," *Annals of Discrete Mathematics,* **5**, 287–326 1979.

Lee, C.-Y., T. C. E. Cheng, and B. M. T. Lin, "Minimizing the Makespan in the 3-Machine Assembly-Type Flowshop Scheduling Problem," *Management Science*, **39**(5), 616–625 1993.

Leung, J.Y.-T, H. Li, and M. Pinedo, "Order scheduling in an environment with dedicated resources in parallel," *Journal of Scheduling*, **8**, 355–386 2005a.

Leung, J.Y.-T, H. Li, and M. Pinedo, "Scheduling orders for multiple product types to minimize total weighted completion time," *Working Paper* 2005b.

Leung, J.Y.-T, H. Li, and M. Pinedo, "Order scheduling models: An Overview," In Multidisciplinary scheduling: Theory and applications , G. Kendall, E. K. Burke, S. Petrovic and M. Gendreau (eds.), Springer, New York, 2005c.

Leung, J.Y.-T, H. Li, M. Pinedo and C. Sriskandarajah, "Open shops with jobs overlap—revisited," *European Journal of Operational Research,* **163**, 569–571 2005.

Morton, T. E. and D. W. Pentico, *Heuristic Scheduling System*, Wiley, New York (1993).

Ng, C.T., T. C. E. Cheng, and J. J. Yuan, "Concurrent open shop scheduling to minimize the weighted number of tardy jobs," *Journal of Scheduling,* **6**, 405–412 (2003).

Pinedo, M., Scheduling: Theory, algorithms and systems, Prentice Hall, Englewood Cliffs, NJ (2002).

Potts, C. N., S. V. Sevast'Janov, V. A. Strusevich, L. N. Van Wassenhove and C. M. Zwaneveld, "The two-stage assembly scheduling problem: Complexity and approximations," *Operations Research*, **43**(2), 346–355 1995.

Roemer, T. A. and R. Ahmadi, "The Complexity of Scheduling Customer Orders," *Working Paper, Anderson School at UCLA* (1997a).

Roemer, T. A. and R. Ahmadi, "The Complexity of scheduling customer orders," *INFORMS*, Dallas (1997b).

Sung, C. S. and S. H. Yoon, "Minimizing total weighted completion time at a pre-assembly stage composed of two feeding machines," *International Journal of Production Economics,* **54**, 247–255 (1998).

Wagneur, E. and C. Sriskandarajah, "Open shops with jobs overlap," *European Journal of Operational Research,* **71**, 366–378 (1993).

Wang, G. and T. C. E. Cheng, "Customer order scheduling to minimize total weighted completion time," In *Proceedings of the first Multidisciplinary Conference on Scheduling Theory and Applications*, 2003, pp. 409–416.

Yang, J., "The complexity of customer order scheduling problems on parallel machines," *Computers & Operations Research,* **32**, 1921–1939 2005.

Yang, J. and M. E. Posner, "Scheduling parallel machines for the customer order problem," *Journal of Scheduling,* **8**, 49–74 (2005).