An accelerated MIP model for the single machine scheduling with preventive maintenance

Omar Souissi, Rachid Benmansour, Abdelhakim Artiba

Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnline CONFERENCE PAPER ARCHIVE

IFAC-PapersOnLine 49-12 (2016) 1945–1949

An accelerated MIP model for the single machine scheduling with preventive maintenance

Omar SOUISSI * **Rachid BENMANSOUR** ** **Abdelhakim ARTIBA** ***

* *University of Technology of Troyes, France, (e-mail: omar.souissi@utt.fr).*
** *University of Valenciennes, France, (e-mail: Rachid.Benmansour@univ-valenciennes.fr)*
*** *University of Valenciennes, France, (e-mail: abdelhakim.artiba@univ-valenciennes.fr)*

Abstract: One of the key requirements for progress in supply chain and manufacturing management is to optimize conjointly production and maintenance scheduling. The main observation we made while reviewing the literature is the lack of approaches that yield the inter-dependence of those activities. This research investigates the single machine scheduling problem tacking into account preventive maintenance failures failures. In this paper we propose an hybrid approach which combine a mixed integer program and the well known LPT (longest processing time first) heuristic in order to solve this problem. A simulation approaches also is proposed to deal with the stochastic failures. We developed a simulation model in order to assess the performance of these approaches.

Keywords: Single machine scheduling; Preventive Maintenance; Production; Heuristic; Algorithm; MIP; Hybrid approach

1. INTRODUCTION

In manufacturing industry, the supply chain and production management are usually subject to stochastic events such demand fluctuation, random machines failure, etc. In this context, development of new industrial strategies which consider both production and maintenance has attracted several researchers. Indeed, consequences of unexpected failures could lead to significant losses.

In this research, we explore the problem of tasks scheduling in a single machine subject to random failures. We assume that the tasks execution is non-preemptive, and there is no precedence constraints between tasks. We formulate an exact method and an LPT-based heuristic in order to deal with makespan minimization.

The effort to develop an exact method was motivated by its importance in the static initial phase, especially in view of the lack of contributions in the literature review. We developed the LPT-based heuristic in order to accelerate the exact method using an upper bound. Indeed, we propose a preprocessing procedure to reduce the solution space of the problem.

This paper is structured as follows. Section 2 reviews some representative related contributions. Sections 3 describes the problem statement. An exact method and a hybrid approach are presented in detail in Sections 4 and 5; respectively. In Section 7, we present experimental results on different instances of our scheduling problem. Section 8 summarize the contributions of this paper and presents the perspectives of the future research.

2. RELATED WORKS

Production scheduling and preventive maintenance both are undoubtedly among the most studied problems in operation research literature. Several excellent reviews can be found in Pinedo (2012) and Valdez-Flores (1989) respectively about scheduling and preventive maintenance. Recently, research efforts are increasingly oriented towards integrating preventive maintenance in scheduling for industrial problems. Indeed several works in the literature proves that production systems within preventive maintenance strategies are less subject to random failures see Nakagawa and Yasui (1991) and R.E. Barlow and Hunter (1996). Thus, in the last two decades several works have been proposed for integrating preventive maintenance in scheduling. Ashayeri et al. proposed in J. Ashayeri and Selen (1996) a production and maintenance planning model for the process industry. Graves and Lee Graves and Lee (1999) consider a single-machine scheduling problem with a unique maintenance activity and the objective of minimizing the total weighted completion time. Lee and Chen Lee and Chen (2000) extend the problem to parallel machines keeping a unique maintenance activity. Kaabi J. Kaabi and Zerhouni (2003) presented a joint production and preventive maintenance policy for a two machines flow shop. (A. Berrichi, 2009) and Berrichi and Yalaoui (2013) proposed bi-objective approach for scheduling with integrated preventive maintenance on par-

10.1016/j.ifacol.2016.07.915

allel machine while considering consumable resources always available. Other authors have explicitly integrated random failures in the joint optimization of maintenance and production to minimize the makespan and the the sum of maximum weighted earliness and tardiness cost Benmansour et al. (2012) and Benmansour et al. (2014)

3. PROBLEM DESCRIPTION

We address the problem of scheduling several jobs on a single machine subject to random failures. The objective is to find an optimal sequence of jobs to reduce the makespan. A job interrupted by a breakdown is re-executed with its initial processing time, and the work done on this job before failure (added value) is lost. The machine undergoes a minimal repair whenever it fails. The proposed joint management of the production and the preventive maintenance consists in coordinating the scheduling of several jobs and the preventive maintenance actions in order to minimize the total time of the schedule C_{max}. All the jobs are available at time 0 and the machine is assumed to be simultaneously available at the beginning of the scheduling period. The preventive maintenance actions are performed regularly on the machine in a cyclic manner and with a periodicity equal to T. After a preventive maintenance action, the machine is considered to be "*as good as new*".

This problem is proven to be NP-Hard, see Lee and Liman (1992).

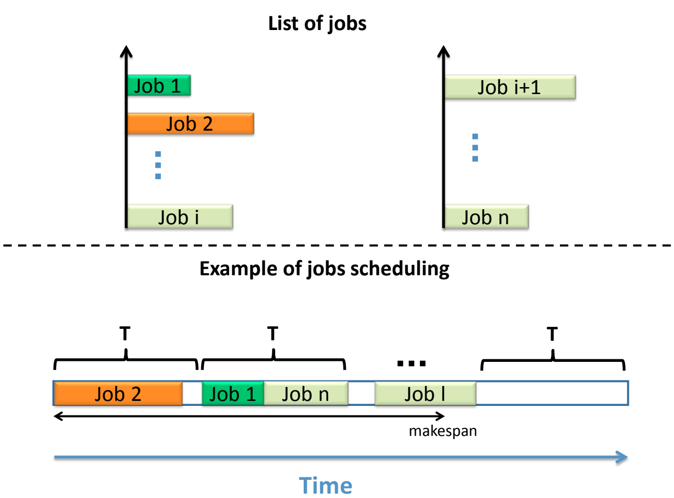

Fig. 1. Example of jobs scheduling with a preventive maintenance cycle T

4. MATHEMATICAL MODEL

4.1 Notation

Symbol	Definition
S_i	Start time of job i
P_i	Processing time of job i
x_{ij}	binary decision variable to assign a job i to a batch j
a_{ik}	binary variable for precedence constraint

4.2 System constraints

- First, each task must be assigned to a batch, where N is the set of batches and $\mid N \mid = $ n : $\forall i \in \{1 \cdots n\}$.

$$\sum_{j=1}^{m} x_{ij} = 1 \tag{1}$$

  This equation forces each job to have its $x_{ij} = 1$ for one and only one batch j.

- Secondly, the maximum load of each batch must be respected $\forall j \in \{1 \cdots n\}$

$$\sum_{i=1}^{n} P_i * x_{ij} \leq T \tag{2}$$

  This equation forces the load of each batch to be less than or equal to the preventive maintenance cycle T.

- Thirdly, execution of jobs is forbidden during preventive maintenance
  $\forall i \in \{1 \cdots n\}$, $\forall k \in \{2 \cdots n\}$

$$\begin{aligned} S_i &\geq (j-1) * T * x_{ij} \\ S_i + P_i &\leq j * T * x_{ij} + (1 - x_{ij}) * M \end{aligned} \tag{3}$$

- Finally, the preemption is not allowed:
  If two jobs, say i and k are assigned to the same batch then they must be separated by the processing time of the earliest one. Since we do not know beforehand which job will precede which other job on any batch, let a_{ik} be a binary variable as presented in Figure 2. The no-preemption condition is then provided by the following two inequalities, in which M is a large number.
  $\forall i, k \in \{1 \cdots n\}$ $\forall k \in \{i+1 \cdots n\}$

$$\begin{aligned} S_i + P_i &\leq S_k + M \times a_{ik} \\ S_k + P_k &\leq S_i + M \times (1 - a_{ik}) \end{aligned} \tag{4}$$

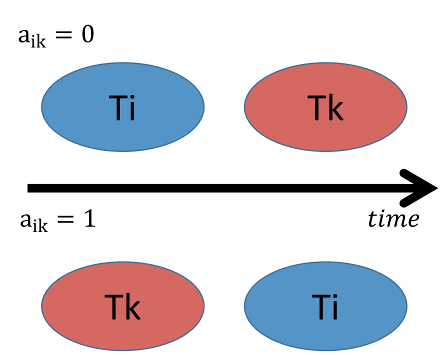

Fig. 2. Definition of the a_{ik} variables

4.3 Objective function

First lets define the objective variable C_{max} as follow:
$\forall i \in \{1 \cdots n\}$

$$C_{max} \geq S_i + P_i \tag{5}$$

In this work we focus on minimizing the makespan. This is realized by adopting the following objective function:

$min\ C_{max}$

5. THE LPT-BASED HEURISTIC

LPT priority rule is one of the oldest methods in scheduling. Already in 1969, Graham (1969) gave results about "the worst-case ratio bound of the LPT rule" for the parallel machine makespan minimization problem. The use of the LPT rule is still widespread nowadays. In Koulamas and Kyparisis (2009), the authors demonstrated the robustness of this approach by implementing it for a two identical parallel machine scheduling problem.

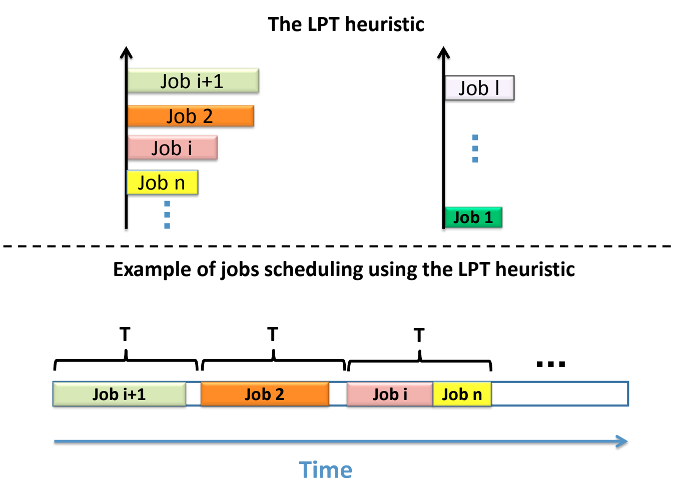

Fig. 3. Example of jobs scheduling using the LPT heuristic

The deployement of the LPT heuristic requires first a "preprocessing" step in which we rank the jobs according to their completion time. Afterwards the jobs are scheduled on the machine between two successive periods of preventive maintenance. The LPT heuristic processing is detailed

Algorithm 1 LPT heuristic

Initialize "L" the list of jobs
Preprocessing STEP: Rank the N tasks of "L" in order of decreasing processing time P[j]
Set completion time of the first job in "L" equal to its processing time C[0]=P[0]
while there are unscheduled jobs in the list of N jobs **do**
  if Completion time of job "j-1" C[j-1] + Processing Time of job "j" P[j] < Current period T **then**
    Assign job j before preventive maintenance
  else
    Assign job j after preventive maintenance
  end if
end while
Set the makespan value equal to the completion time of each job "N" $makespan = C[N]$

6. THE HYBRID APPROACH WITH MIP AND LPT-BASED HEURISTIC

The major drawback of the branch and bound approach is the prohibitive number of solutions. Thus, the enumeration steps are very important and largely affects the solution efficiency. Indeed the efficiency of a Branch-and-Bound algorithm is affected by the node branching criteria.

In order to improve the quality of cuts and bounds and accelerate the MIP resolution, we expect to use the LPT heuristic. Indeed, as exposed in the second case study in the experimental results paragraph, the LPT heuristic produces results of high quality in terms of makespan which are even optimal for some instances. Thus, the LPT heuristic will be used in order to produce an upper bound that can fathom subproblems. The key idea of the well known branch and bound algorithm is to decompose a given optimization problem into several subproblems. The decomposition is repeatedly applied and each subproblem is either solved or proved not to lead to an optimal solution of the initial problem. Thus, our idea is to discard each node of the tree (See Figure) whose lower bound is greater than the upper bound produced by the LPT heuristic.

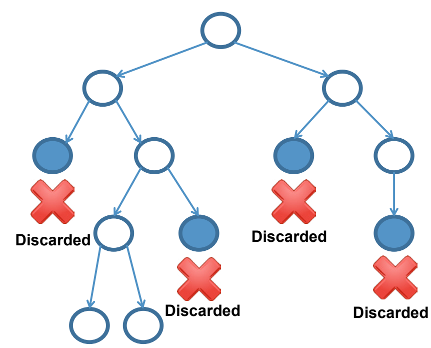

Fig. 4. Branch and bound

7. SIMULATION MODEL

Predicting the scheduling outcome without running the machine implies an uncertainty factor. Advanced technological developments in the last deacdes have enabled the use of simulation models to test the performance of the manufacturing lines even before they exist to consolidate the chosen scheduling approaches.

The strategy we adopted for our simulation model consists in the determination of the probabilistic moments of the next failure given the age of the machine. To minimize makespan C_{max}, preventive maintenance actions are performed regularly on the machine to ensure that its reliability is always greater than or equal to the minimum level of reliability of the machine R_{min}.

8. EXPERIMENTAL RESULTS

In this section we present a comparison between the original MIP and the accelerated version using the LPT heuristic. We used a machine equipped with a 2,5GHz Intel 2520 processor and 16 GB RAM Memory. To solve the integer linear program described in Section 4, we used

Algorithm 2 Simulation Algorithm

while n_{sim} (number of simulations) is not reached **do**
  Initialize Age,waste,fail,n_{PM} (number of preventive maintenance) to "0"
  while Set of jobs N is not empty **do**
    Generate the new instant of failure x (given that the machine has age Age)
    if (Completion time of job "j" C[j] is lower than instant of failure x) and (Completion time of job "j" C[j] is lower than preventive maintenance period T) **then**
      Assign job "j"
    else
      Do preventive maitenance
    end if
  end while
  for each job j in the list of N jobs **do**
    Set the completion time of each job "j" C[j]
  end for
  Set the makespan value $C_{max} = C[N]$
end while

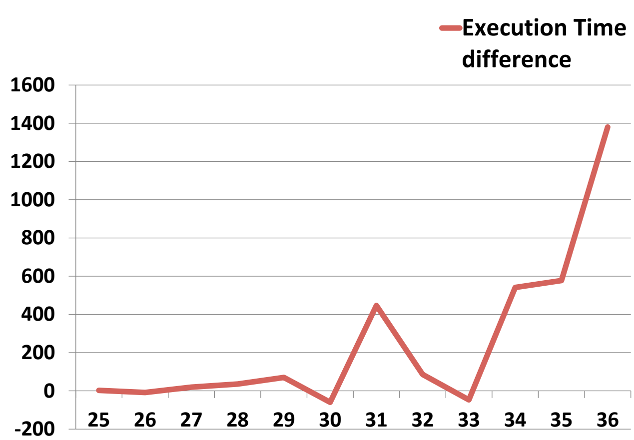

Fig. 5. MIP Vs Accelerated MIP

the IBM ILOG optimization tool CPLEX 12.6.1. The LPT heuristic and the simulation model were developped with the C language.

	Time		C_{max}	
Instances	MIP	A. MIP	MIP	A. MIP
25	5,61	3,2	1319	1561
26	21,71	30	1325	1583
27	52,91	33,39	1405	1663
28	67	31,15	1485	1743
29	99	28,89	1485	1780
30	60	120	1565	1860
31	480	33,1	1644	1940
32	506	420	1644	1982
33	108	155	1724	2062
34	682	141	1804	2142
35	840	263	1884	2222
36	1860	480	1884	2260

We observe that once we execute instances with a high number of jobs the the MIP mixed to LPT outperform the basic MIP. Thus, experiments validate the idea of an accelerated MIP while using for each instance the LPT solution as an upper bound.

9. CONCLUSION

In this paper we studied the problem of single machine scheduling with preventive maintenance. This problem is proven to be NP hard, see Lee and Liman (1992). The main observation we made while reviewing the literature is the lack of approaches that yield the exact solution for single machine scheduling with prventive maintenance. Indeed most of published works comsider the production and the maintenance separately or sequentially.

In this paper, we proposed an original and an accelerated MIP for a joint management of the production and preventive maintenance. The objective function of the MIP is the makespan Cmax minimization.

In order to enrich this work, in future research, we will compare the performance of our approaches with the achievement of several experiments. We plan to extend our algorithms to deal with the multi-objective "makespan minimization" and "load balancing" of batches. Finally, we intend to investigate the dynamic mapping in order to deal with re-allocation of jobs so as to anticipate the onset of failures.

REFERENCES

A. Berrichi, L. Amodeo, F.Y.E.C. (2009). Bi-objective optimization algorithms for joint production and maintenance scheduling: application to the parallel machine problem. *Journal of Intelligent Manufacturing*, 20, 389–400.

Benmansour, R., Allaoui, H., Artiba, A., and Hanafi, S. (2014). Minimizing the weighted sum of maximum earliness and maximum tardiness costs on a single machine with periodic preventive maintenance. *Computers & Operations Research*, 47, 106–113.

Benmansour, R., Artiba, A., Duvivier, D., Ramat, E., and Ait-Kadi, D. (2012). scheduling of production and maintenance activities under reliability constraint. In *9th International Conference on Modeling, Optimization & SIMulation*.

Berrichi, A. and Yalaoui, F. (2013). Efficient bi-objective ant colony approach to minimize total tardiness and system unavailability for a parallel machine scheduling problem. *International Journal of Advanced Manufacturing Technology*, 68, 2295–2310.

Graham, R.L. (1969). Bounds on multiprocessing timing anomalies. *SIAM Journal on Applied Mathematics*.

Graves, G.H. and Lee, C.Y. (1999). Scheduling maintenance and semiresumable jobs on a single machine. *Nav. Res. Logist*, 46, 845–863.

J. Ashayeri, A.T. and Selen, W. (1996). A production and maintenance planning model for the process industry. *Int. J. Production Res.*, 34, 3311–3326.

J. Kaabi, C.V. and Zerhouni, N. (2003). Odonnancement de la production et de la maintenance : cas dun atelier de type flow shop a deux machines. *APIIJESA*, 37, 641–660.

Koulamas, C. and Kyparisis, G.J. (2009). A modified lpt algorithm for the two uniform parallel machine

makespan minimization problem. *European Journal of Operational Research*, 196, 61–68.
Lee, C.Y. and Chen, Z.L. (2000). Scheduling jobs and maintenance activities on parallel machines. *Nav. Res. Logist*, 47, 145–165.
Lee, C. and Liman, S. (1992). Single machine flow-time scheduling with scheduled maintenance. *Acta Informatica*, 29, 375–382.
Nakagawa, T. and Yasui, K. (1991). Periodic-replacement models with threshold levels. *IEEE Transactions on Reliability*, 40, 395–397.
Pinedo, M. (2012). Scheduling theory, algorithms, and systems. *Prentice Hall*, 4th edition.
R.E. Barlow, F.P. and Hunter, L. (1996). Mathematical theory of reliability. *Siam Philadelphia*, 17.
Valdez-Flores, C. (1989). Survey of preventive maintenance models for stochastically deteriorating single-unit systems. *Nav. Res. Logist.*, 4, 419–446.