**Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014**

Cyclic Scheduling Steady-State Analysis and Improved Mathematical Models

Hongchang ZHANG *, Simon COLLART-DUTILLEUL **
Khaled MESGHOUNI ***

*LAGIS : Laboratoire d'Automatique, Génie Informatique et Signal, EC-LILLE : Ecole Centrale de Lille, 59650,Villeneuve d'Ascq, France (Tel:+33320335455; e-mail: hongchang.zhang@ec-lille.fr).
**ESTAS, IFSTTAR, France (e-mail: simon.collart-dutilleul@ifsttar.fr)
*** LAGIS, EC-LILLE, France (e-mail: khaled.mesghouni@ec-lille.fr)

Abstract: This paper gives a new viewpoint for steady-state analysis of classical job-shop cyclic scheduling (CJCS). After refining the constraints in the analysis of steady-state, improved models are achieved. By using the improved models, two kinds of problems can be solved, which are minimizing work-in-progress (WIP) with optimal cycle time and minimizing the cycle time with limited WIP. Moreover, the mathematical models are programmed in CPLEX 12.5 and good experimental results are presented.

Keywords: Cyclic Scheduling, Steady-State, Job-Shop, WIP, Mathematical Model, CPLEX.

1. INTRODUCTION

The cyclic scheduling is seen as a scheduling problem in which some set of activities is to be repeated an infinite number of times (Draper et al., 1999). More precisely, if x(n) is the starting time (or ending time) of one activity, and n means the repeat numbers, then there is a constant C (called the cycle time which is the inverse of the periodic throughput) and one integer K such that

$$x(n+k) = x(n) + k * C \quad \text{for } n \in N, k \in N^{+}, C \geq 0$$

It is has been shown that the cyclic scheduling is NP-hard (Carlier and Chrétienne, 1988; Hanen, 1994; Bodin et al., 2012), for which the heuristics algorithm are always recommended (Kampmeyer, 2006; Jalivand-Nejad and Fattahi, 2013). The k-cyclic scheduling is studied in (Chrétienne et al., 1997; Brucker and Kampmeyer, 2008; Amraoui et al., 2013). Especially, in (Camus et al., 1996; Korbaa et al., 2003; Bourdeaud'huy and Korbaa, 2006; Amar et al., 2011), the 1-cyclic (k=1) scheduling of job-shop is considered. It is assumed that, in our work, only the 1-cyclic scheduling technique is studied. It is admitted that 1-cyclic scheduling is more easily to be controlled, because if the scheduling of system is known in just one cycle time C, we can predict all the system's activities in the future.

In this paper, the cyclic scheduling is applied on the classical job-shop (CJCS). In this kind of job-shop scheduling problem, a finite set of jobs is processed on a finite set of machines. Each job is characterized by a fixed order of operations, each of which is to be processed on a specific machine for a specified duration. Each machine can process at most one job at a time and once a job initiates processing on a given machine it must complete processing uninterrupted. The objective of the problem is to find a schedule either to minimize the cycle time of job-shop or to minimize the work-in-progress (WIP) in job-shop.

In our model, the pallets allocated to products are used to transport the job between machines, and the pallets will only be removed from the products after completely finishing the job, so in this paper, the number of pallets for one job is equal to the number of WIP for this job.

Factually, this kind of job-shop problem can be described as a strongly connected event graph (SCEG) (Ohl et al., 1995). It has been shown that the cycle time of system should be the maximal cycle time of all the elementary circuits in SCEG. This conclusion can be used as a good cut technique to set a low bound for system's cycle time, which obviously reduces the research space of admissible solutions.

Basically, the paper is organized as the follows: in the second part, a new viewpoint of cyclic scheduling on steady-state analysis will be presented step by step; after the explanation of the new viewpoint, two refined mathematical model are given, one is to resolve the job-shop cyclic scheduling with WIP minimization, another is to resolve the job-shop cyclic scheduling with limited WIP for minimizing the cycle time of system; in the end, the experimental results by using CPLEX are presented, and comparisons with models in (Bourdeaud'huy and Korbaa, 2006; Amar et al., 2011) are also made.

2. STEADY-STATE ANALYSIS

2.1 Basic Notation

Before the analysis of steady-state in job-shop cyclic scheduling, some basic notation would be given firstly.

- p_j : jobs or products, j is the index of jobs, $j \in N^*$
- m_l: machines, l is the index of machines, $l \in N^*$
- o_{jkl} : the kth operation of p_j processed on m_l, $k \in N^*$

- t_{jkl} : start date of operation o_{jkl}, $t_{jkl} \geq 0$
- d_{jkl} : the duration of operation o_{jkl}, $d_{jkl} \geq 0$
- WIP_{jm} : the mth WIP for p_j, $m \in N^*$
- W_j : total number of pallets for p_j, $W_j \in N^*$
- W_{limit} : limit number of pallets for system, $W_{limit} \in N^*$
- C : variable stands for cycle time of system, $C > 0$
- C_{max} : the optimal cycle time for cyclic scheduling of a production system, it is also the low bound of system's cycle time, $C_{max} > 0$
- J : $J = max\{j\}$, the quantity of jobs
- K_j: number of operations for p_j
- L : $L = max\{l\}$, the quantity of Machines
- α_{jk} : Boolean variables
- $\gamma_{ij}^{i'j'}$: Boolean variables
- β_{ij} : integer number, $\beta_{ij} \in N$

2.2 *Explanation of CJCS*

The new view point to explain CJCS will be presented step by step. A simple CJCS with saturation of machines in figure.1 is taken as an example.

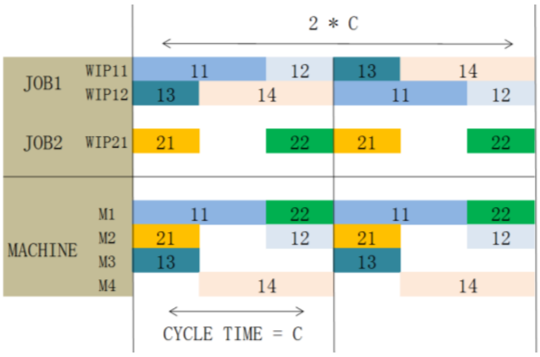


Fig.1.One simple CJCS example.

In this example, the job-shop has two kinds of jobs: job1 and job2. For job1, it has two WIP or pallets, WIP11 and WIP12, and it has 4 operations for one complete part; for job2, it has one WIP, WIP21, and it has 2 operations for one complete part. All the operations of job1 and job2 should be executed on 4 different machines, and it is easily to find that m_1 is saturated, which implies the cycle time of system is the sum of all the durations of operations on m_1 which is the bottleneck machine.

Constraints on viewpoint of both p_j and m_l :

However, as shown in figure 1, if we observe the cyclic scheduling in one cycle time C, all the start date of operations should be in the interval [0, C[. Constraints family (1) permits the cross cycle of one operation, which means one operation start in one cycle but end in another. Hence, (1) doesn't lose the generality for arranging the start time of one operation.

$$0 \leq t_{jkl} < C \tag{1}$$

Constraints on viewpoint of m_l :

If we observe all the operations which are executed on the same machine m_l , one machine can only process one operation at a moment, so no overlap between the operations executed on the same machine should be respected. Constraints families (2) and (3) give proper constraints for avoid the overlapping of operations o_{jkl} and $o_{i'j'l}$ which are executed by m_l.

$$t_{jkl} + d_{jkl} - t_{j'k'l} \leq \gamma_{ij}^{i'j'} * C \tag{2}$$

$$t_{j'k'l} + d_{j'k'l} - t_{jkl} \leq \gamma_{ij}^{i'j'} * C \tag{3}$$

Constraints on viewpoint of p_j :

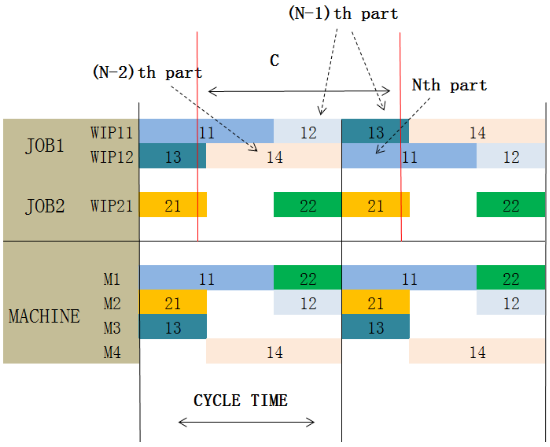


Fig.2. Different parts in one cycle time.

For 1-cyclic scheduling, in one cycle time, different parts for the same job can be observed. For example, in one cycle time as in figure 1, two parts for job1 are observed; in one cycle time as shown in figure 2, three parts for job1 are observed. Surely there is an order between the parts for the same job. Hence, constraints family (4) should be given.

$$0 = \beta_{j1} \leq \beta_{jk} \leq \beta_{jk+1} \leq K_j \tag{4}$$

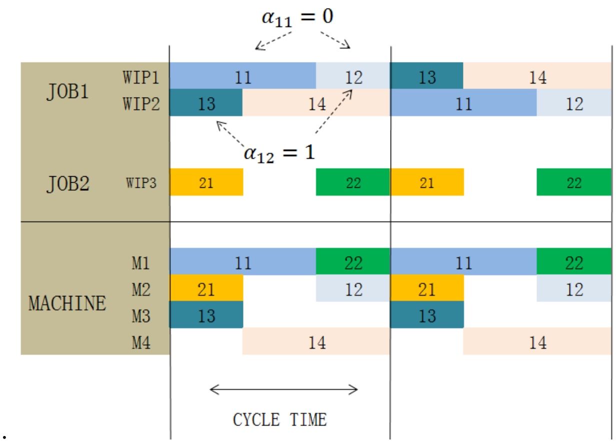
As shown in figure 2, in one cycle time, it exists (N-β_{j1})th part, (N-β_{j2})th part, and (N-β_{j3})th part. The up bound of β_{jk} is K_j, which implies the situation that on one pallet just one operation is executed in one cycle time.

The value of difference between β_{jk} and β_{jk+1} is 0 or 1, which can be determined by one Boolean variable α_{jk}. As shown in figure 3, when operation o_{jk+1l} starts after the end of o_{jkl}, it stands for these two consecutive operations belong to the same part, thus $\beta_{jk+1} - \beta_{jk} = \alpha_{jk} = 0$, else they belong to different parts, $\beta_{jk+1} - \beta_{jk} = \alpha_{jk} = 1$. The % means complementation in math, which can connect the operations of one part as a cycle.

$$\beta_{j(k\%K_j+1)} - \beta_{jk} = \alpha_{jk} \tag{5}$$

$$\alpha_{jk} = 0 \text{ or } 1 \tag{6}$$

$$(t_{jkl} + d_{jkl}) - t_{j(k\%K_j+1)l} \leq \alpha_{jk} * C \tag{7}$$



.

Fig.3. Value of the Boolean variable α_{jk}.

WIP or pallets quantity formula:

Factually, $\alpha_{jk} = 1$ means the necessity to add one pallet for operation o_{jkl} of job p_j. And operation % can connect the last operation with the first operation of p_j. It is interesting to find that the sum of α_{jk} is exactly the quantity of pallets as presented in the following formula:

$$\sum_1^J W_j = \sum_1^J \sum_1^{K_j} \alpha_{jk} \tag{8}$$

Optimal cycle time formula:

The optimal cycle time C_{max} for the production system is defined by the following expression:

$$C_{max} = max\left(\sum_{d_{jkl}\, executed\ on\ m_l} d_{jkl}\right) \tag{9}$$

the expression $\sum_{d_{jkl}\, executed\ on\ m_l} d_{jkl}$ describes the sum of durations of the operations associated to the machine m_l. Let us denote by bottleneck machine the machine for which this sum is the greatest. Thus, the optimal cycle time is the sum of durations of the operations associated to the bottleneck machine. It is possible to obtain an optimal total production time by saturating the bottleneck machine of the considered production system (Bourdeaud'huy and Korbaa, 2006).

2.3 Refined Mathematical Models

Generally, two kinds of problems are studied for CJCS: one is to find the minimal number of pallets or minimal WIP with machines saturated, another is to find the minimal cycle time (not necessary the optimal cycle time C_{max}) with limited pallets or limited WIP. These two kinds of models are presented as follows.

Model for CJCS with machines saturated for WIP minimization

This model could be directly refined by the constraints in section 2.2: (4) (5) (6) (7) could be combined into (13). This is a linear model. And some machines are saturated, so the optimal cycle time C_{max} is reachable, let us change C with C_{max}.

Objective function:

$$\text{Minimize } \sum_1^J \sum_1^{K_j} \alpha_{jk} \tag{9}$$

Constraints both on viewpoint of p_j and m_l:

$$0 \leq t_{jkl} < C_{max} \tag{10}$$

Constraints on viewpoint of m_l :

$$t_{jkl} + d_{jkl} - t_{j'k'l} \leq \gamma_{ij}^{i'j'} * C_{max} \tag{11}$$

$$t_{j'k'l} + d_{j'k'l} - t_{jkl} \leq \left(1 - \gamma_{ij}^{i'j'}\right) * C_{max} \tag{12}$$

Constraints on viewpoint of p_j :

$$(t_{jkl} + d_{jkl}) - t_{j(k\%K_j+1)l} \leq \alpha_{jk} * C_{max} \tag{13}$$

Thus a refined model with only 4 constraints families (10) - (13) is gotten, which is simpler than the model with 7 constraints families in (Bourdeaud'huy and Korbaa, 2006).

However some cut techniques about bounds of objective function to reduce the research space could be used (Amar et al., 2007).

The low bound for WIP:

$$\sum_1^J \sum_1^{K_j} \alpha_{jk} \geq \sum_1^J \left\lceil \frac{\sum_1^{K_j} d_{jkl}}{C_{max}} \right\rceil \tag{14}$$

⌈ ⌉ stands for the ceiling operator. The above bound means if the sum of operations for one part is superior than C_{max}, more pallets should be used to guarantee the system run in C_{max} cycle time.

The up bound for WIP:

$$\sum_1^J \sum_1^{K_j} \alpha_{jk} \leq \sum_1^J K_j \tag{15}$$

Constraint (15) implies that, in one cycle time, one pallet is proposed as transport resources for each operation.

Model for CJCS with limited WIP for minimizing cycle time

This is an un-linear model. First, it needs to change the objective function as follow:

$$\text{Minimize } C \tag{16}$$

For the constraints, it needs to change $C_{optimal}$ with C in the constraints families. The minimal cycle time is not the optimal cycle time, because the number of pallets is inferior than the minimal quantity of pallets to realize optimal cycle time, other words, the transport resources are limited, thus no one machine is saturated.

Constraints both on viewpoint of p_j and m_l:

$$0 \leq t_{jkl} < C \quad (17)$$

$$\sum_1^J \sum_1^{K_j} \alpha_{jk} \leq W_{limit} \quad (18)$$

Constraint (18) means the job-shop can use at most W_{limit} pallets.

Constraints on viewpoint of m_l :

$$t_{jkl} + d_{jkl} - t_{j'k'l} \leq \gamma_{ij}^{i'j'} * C \quad (19)$$

$$t_{j'k'l} + d_{j'k'l} - t_{jkl} \leq \left(1 - \gamma_{ij}^{i'j'}\right) * C \quad (20)$$

Constraints on viewpoint of p_j :

$$\left(t_{jkl} + d_{jkl}\right) - t_{j(k\%K_j+1)l} \leq \alpha_{jk} * C \quad (21)$$

Thus a refined model with only 5 constraints families (17) - (21) is gotten, which is simpler than the model with 8 constraints families in (Amar et al., 2011).

However some cut techniques about bounds of objective function to reduce the research space can be used.

The low bound for cycle time:

$$C > C_{max} \quad (22)$$

Because the WIP quantity is limited to be less than the minimal number of WIPs to realize optimal cycle time, low bound of cycle time can't be reachable as shown in (22).

The up bound for cycle time:

$$C \leq \sum_1^J \sum_1^{K_j} d_{jkl} \quad (23)$$

The up bound stands for the job-shop process only one job at a moment, only when this job is finished, the new job can be processed. Obviously, this is a stupid scheduling with low production efficiency.

3. EXPERIMENTAL RESULT

In section 2, two improved models and the relative cut techniques are given. In this section, the experimental results for the two models are presented using example in (Seo and Lee, 2002). CPLEX 12.5 is used for the constraints programming, on a computer with Intel (R) Core (TM) i3-2310M at 2.10 GHz and 4 Go RAM, under Window 7 (64 bit).

3.1 Example Description

This example consists of three machining centres: m_1, m_2, m_3, three set-up stations s_1, s_2, and s_3 (that can be also considered as machines). The job-shop produces three types of parts: p_1, p_2, and p_3. The pallets fixed to p_1, p_2, and p_3 are allocated or removed at s_1, s_2, and s_3 respectively. The pallets are transported by stack cranes, and when there is no available machine for one part, this part can be stocked into a stocker which has enough capacity to store the parts to release the move of cranes. However, the time of cranes' move isn't considered to avoid excessive complexity.

The machine visit sequences of p_1, p_2, and p_3 are $m_1 \rightarrow m_2$, $m_3 \rightarrow m_1$, and $m_2 \rightarrow m_1$, respectively. A part should visit a set-up station for fixing the part on a pallet before starting each machining operation. After finishing all operations, the part is removed from the pallet at the set-up station. The detailed visit sequences of the parts and the required times (written in the parentheses) are given in Table 1.

Table 1. Visit sequences and processing duration

Part	Visiting sequences				
p_1	$s_1(2)$	$m_1(8)$	$s_1(4)$	$m_2(6)$	$s_1(2)$
p_2	$s_2(2)$	$m_3(10)$	$s_2(4)$	$m_1(4)$	$s_2(2)$
p_3	$s_3(2)$	$m_2(4)$	$s_3(4)$	$m_3(4)$	$s_3(2)$

3.2 Numerical experiments for WIP minimization

In this kind of problem, some machines are saturated. Specially, in this example, m_3 is saturated. This condition implies that in one cycle m_3 process the parts without waiting time. Thus, it is easy to get the value of C_{max}.

$$C_{max} = 10 + 4 = 14 \quad (24)$$

By (14) and (15), the bounds for WIP could be gotten.

$$6 \leq W_j \leq 15 \quad (25)$$

By programming the model in CPLEX 12.5, a cyclic scheduling with optimal number of pallets 6 and cycle time 14 can be gotten as shown in the following Gantt diagram figure 4. It is easy to find that, in one cycle time, the machine m_3 is saturated. For this machine, there is no free time for waiting the next job. It is the bottleneck machine of the production system. The cycle time of system is exactly the sum of operations on this bottleneck machine in one cycle time.

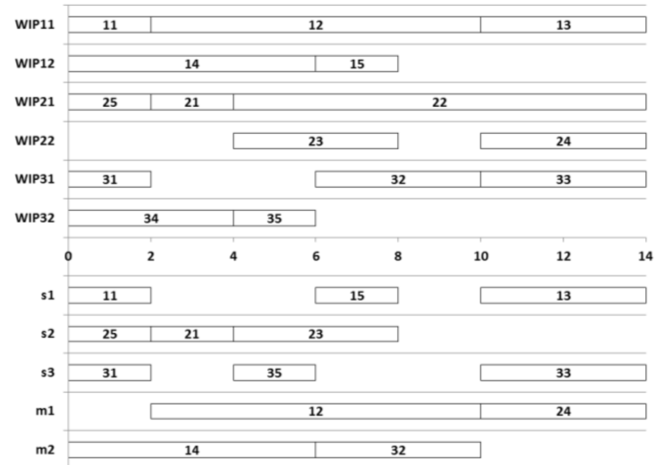


Fig.4. Scheduling of WIP minimization by using model in this paper.

However, by using the model in (Bourdeaud'huy and Korbaa, 2006), a good result can also be achieved by CPLEX 12.5 on the same computer (figure 5). Even the two modes could both give 1-cyclic scheduling, but the computing time is different, the comparison will be made in section 3.4.

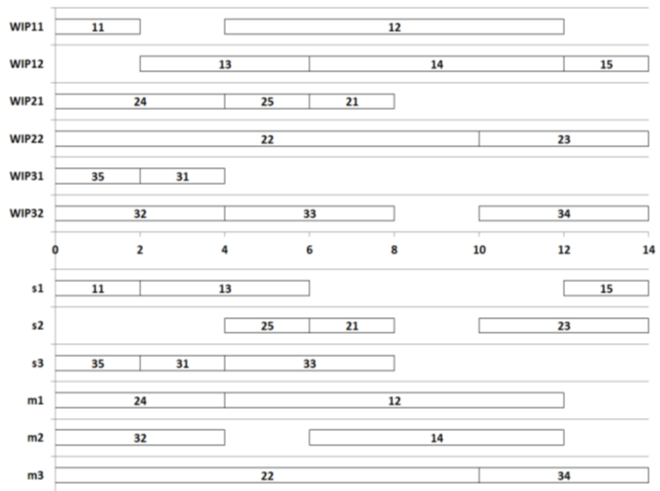


Fig.5. Scheduling for minimizing WIP by using model in (Bourdeaud'huy and Korbaa, 2006).

3.3 Numerical experiments for cycle time minimization

In this kind of problem, the number of WIP or pallets is limited as 5, for instance, 2 WIPs for p_1, 2 WIPs for p_2, and 1 WIP for p_3. The aim is to get a minimal cycle time. By (22) and (23), the bound for cycle time can be gotten.

$$14 < C \leq 60 \tag{26}$$

Note that the cycle time found is 16 equal to the result in (Seo and Lee, 2002). This value is optimal if we take into account that the limit of WIP is fixed to 5. Indeed, under this fixed level of WIP, the job p_3 has to be processed with only one pallet. On considering the process cycle, the cycle time is fixed by this job. Thus the low bound of cycle time is 16 exactly.

First, the result by using the model in this paper is shown as the following Gantt diagram.

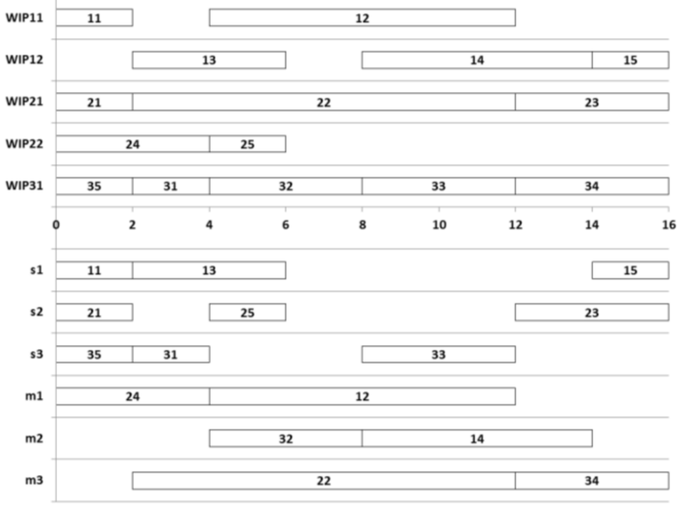


Fig.6. Scheduling for minimizing cycle time by using model in this paper.

However, by using the model in (Amar et al., 2011), a cyclic scheduling with cycle time 16 can also be given as shown in figure 7. The processing times of these two models are different, and the comparison will be made in section 3.4.

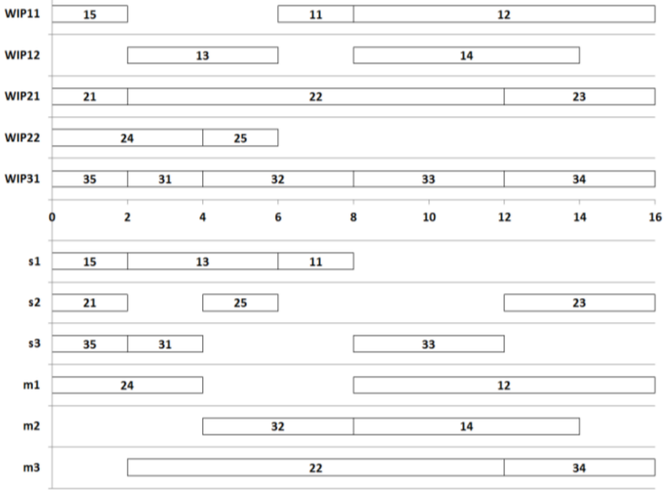


Fig.7. Scheduling for minimizing cycle time by using model in (Amar et al., 2011).

3.4 Comparisons of the Numerical experiments

The models in this paper have two improvements on comparing with models in (Bourdeaud'huy and Korbaa, 2006) and (Amar et al., 2011).

First, the models in paper have less number of constraints families as said in 2.3. The new model (call it NMW) for WIP minimization in this paper has only 4 constraints families, on considering 7 families in model (Bourdeaud'huy and Korbaa, 2006) (call this model MBK); The models (call it NMC) for cycle time minimization in this paper has only 5 constraints families, on considering 8 constraints in model in (Amar et al., 2011) (call this model MAR).

Second, the models in this paper shorten obviously the computing time as shown in Table 2 and Table 3. The average computing time $\bar{t}$ for 10 times of experiments is used to make the comparison among the models. The unit of the data is second (s).

Table 2. Computing time for WIP minimization

Model	Processing time t (s)										$\bar{t}$
NMW	0.07	0.12	0.06	0.09	0.10	0.04	0.10	0.07	0.07	0.07	0.079
MBK	0.15	0.20	0.21	0.24	0.17	0.15	0.15	0.17	0.15	0.24	0.183

Table 3. Computing time for cycle time minimization

Model	Processing time t (s)										$\bar{t}$
NMC	0.07	0.12	0.10	0.12	0.12	0.12	0.10	0.09	0.12	0.10	0.106
MAR	0.14	0.15	0.14	0.17	0.15	0.12	0.12	0.17	0.20	0.17	0.153

The gain for computing time can be calculated by the following formulation (Amar et al., 2007).

$$Gain = 100 * \left(\frac{Processing\ time\ using\ original\ model}{Processing\ time\ using\ new\ model} - 1\right) \tag{27}$$

Gain for Table 2:

$$Gain = 100 * \left(\frac{0.183}{0.079} - 1\right) = 131.65 \quad (28)$$

Gain for Table 3:

$$Gain = 100 * \left(\frac{0.153}{0.106} - 1\right) = 44.34 \quad (29)$$

Thus gain of the new model in this paper for WIP minimization problem is 131.65, and for cycle time minimization with limited WIP is 44.34. The results demonstrate that the new models in this paper have an obvious improvement for computing time. Anyway, the new models have much simpler structure with less constraints families. And, less Boolean variables in the new models reduce the branches for the searching tree.

6. CONCLUSIONS

In this paper, firstly, a new viewpoint to analyse steady-state of CJCS (1-cyclic) is presented step by step with simple examples. After a complete analysis about CJCS, two new models are given for WIP minimization with optimal cycle time and for cycle time minimization with limited WIP. By using CPLEX 12.5, good results of the new models compared to the original models are shown. In the end, the improvement for the new models can be found in the computing time gain.

The contribution of this paper is to propose new MIP models for CJCS (1-cyclic) with simpler structure (less constraints families) and better performance in computing time than previous models. However, more works still should be done in the future:

First, the models should be applied to examples more complex to evaluate the efficiency of models in general conditions.

Second, more details in real production process could be taken into account. However, in this paper, the time for moving the part by cranes between the machines is neglected, and the duration for the operations is fixed (in reality, the duration may change in a proper time interval). It is interesting to assign the operations' duration with time windows and to add the constraints of cranes' moving time, thus the robustness margin of machines or transportation resources in system (Collart-Dutilleul et al., 2013) might be studied.

ACKNOWLEDGEMENT

This research is partially supported by the project PERFECT of the ANR. We sincerely thank the kindly assistance.

REFERENCES

Amar, M.A.B., Bourdeaud'huy, T., and Korbaa, O. (2007). Cyclic scheduling MIP implementation: cutting techniques. *19th International Conference on Production Research.*

Amar, M.A.B., Camus, H., and Korbaa, O. (2011). A Mathematical model for cyclic scheduling with limited work-in-process and cycle time minimization. In: *VECoS 2011,* Tunis, Tunisia.

Amraoui, A.E., Manier, M.A., Moudni, A.E., and Benrejeb, M. (2013). A linear optimization approach to the heterogeneous r-cyclic hoist scheduling problem, *Computers & Industrial Engineering*, Vol. 65, Iss. 3, 360-369.

Bodin, B., Munier-kordon, A., and Dinechin, B.D. (2012). K-periodic schedules for evaluating the maximum throughput of a Synchronous Dataflow graph. *Embedded Computer Systems (SAMOS)*, 152-159.

Bourdeaud'huy, T. and Korbaa, O. (2006). A mathematical model for cyclic scheduling with work-in-progress minimization. *12^th IFAC Symposium on Information Control Problems in Manufacturing,* Vol. 12, part 1, Saint Etienne, France.

Brucker, P. and Kampmeyer, T. (2008). A general model for cyclic machine scheduling problems. *Discrete Applied Mathematics*, Vol. 156, Issue 13, 2561-2572.

Camus, H., Ohl, H., Korbaa, O., and Gentina, J.C. (1996). Cyclic schedules in flexible manufacturing systems with flexibilities in operating sequences. In: *ICATPN'96.*

Carlier, J., and Chrétienne, P. (1988). *Problèmes d'ordonnancement: modélisation / complexité / algorithmes*. Editions Masson.

Chrétienne, P., Coffman, E.G., Lenstra, J.K., and LIU, Z. (1997). *Scheduling theory and its applications*,.Wiley editorials.

Collart-Dutilleul, S., Mhalla, A., Craye, E., and Benrejeb, M. (2013). Active robustness of a milk manufacturing workshop with time constraints. *International Journal of Production Research*, Vol. 51, Iss. 1.

Draper, D.L., Jonsson, A.K., Clements, D.P., and Joslin, D.E. (1999), Cyclic scheduling. In *Proceedings of the 16th international joint conference on Artificial intelligence – Volume 2 (IJCAI'99)*, Vol. 2. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1016-1021.

Hanen, C. (1994). Study of a NP-hard cyclic scheduling problem: The recurrent job-shop. European journal of operational research, Vol. 72, No. 1, 82-101.

Jalilvand-Nejad, A. and Fattahi, P. (2013). A mathematical model and genetic algorithm to cyclic flexible job shop. *Journal of Intelligent Manufacturing.*

Kampmeyer, T. (2006). Cyclic scheduling problems. PhD theses, University of Osnabruck, Germany.

Korbaa, O., Benasser, A., and Yim, P. (2003). Two FMS scheduling methods based on Petri nets: a global and a local approach. *International Journal of Production Research* 41(7), 1349–1371.

Ohl, H., Camus, H., Castelain, E., and Gentina, J.C. (1995). Petri-net modelling of ratio-driven flexible manufacturing systems and implications on the WIP for cyclic schedules. *Systems, Man and Cybernetics, 1995. Intelligent Systems for the 21st Century., IEEE International Conference* ,Vol. 4, 3081-3086.

Seo, J.W. and Lee, T.E. (2002). Steady-State Analysis and Scheduling of Cyclic Job Shops with Overtaking. *The International Journal of Flexible Manufacturing Systems*, Vol. 13, 291-318.