

La gestion des files d'attente par les règles de priorité

Zoheir KARAOUZENE, Zaki SARI.

LAT : Laboratoire d'Automatique de Tlemcen

Faculté des sciences de l'ingénieur

Université ABOU-BEKR BELKAÏD

BP 230 Tlemcen, ALGERIE

zoheir_karaouzene@yahoo.fr, zaki_sari@yahoo.com

Résumé- Le contexte des ateliers n'est pas stable, en face des imprévus comme la panne d'une machine, existence d'une station goulot ou recevoir une commande urgente, il faut réagir devant ces problèmes et trouver une solution capable d'assurer la bonne continuité du système de production. Dans cette situation, plusieurs approches sont proposées pour résoudre les problèmes d'ordonnement, la gestion des files d'attente par des règles de priorité est l'une des approches les plus simples pour ordonner en temps réel les opérations dans un atelier flexible.

Nous proposons dans ce travail une simulation d'un ensemble de règles de priorité les plus utilisées, sur un modèle Job Shop à quatre machines implémenté par le logiciel de simulation ARENA, nous présentons par la suite, les résultats de la simulation sous plusieurs conditions opératoires pour un objectif fixé.

I. INTRODUCTION

Le système de production rassemble l'ensemble des moyens qui permettent la transformation de ressources en produits ou en services [1].

La fonction d'ordonnement, dans un système de production, a pour rôle d'assurer la production des quantités de produits fixés dans le plan de production (plan directeur) dans les délais [2].

Dans un problème d'ordonnement Job-Shop $n \times m$, n jobs doivent être effectués par m machines, sachant que chaque machine ne peut travailler que sur un seul job à la fois [3].

Le problème Job-Shop est NP-difficile, considéré parmi les problèmes les plus difficiles à traiter, le nombre de solutions possibles est de l'ordre de $(n!)^m$, où n est le nombre de tâches à effectuer et m le nombre de machines.

la gestion des files d'attente par des règles de priorité est l'une des approches les plus simples pour ordonner en temps réel les opérations dans un atelier flexible.

II. LES REGLES DE PRIORITE

Les règles de priorité permettent de sélectionner parmi une liste de lots en attente le prochain lot à traiter sur la machine en fonction de critères purement locaux (données dépendant uniquement du lot) ou plus globaux (prenant en compte tout ou partie de l'état du système) [3].

Une règle peut être très simple par exemple « choisir un travail au hasard (aléatoire) », « choisir le travail avec le plus grand temps d'attente (FIFO) », ou extrêmement complexe par exemple « choisir le travail dont la prochaine opération est réalisé sur la ressource la moins utilisée ». Le choix d'une telle

règle est réalisé en fonction des objectifs fixés et les conditions opératoires propres à cet atelier.

La règle choisie peut apparaître la plus performante pour satisfaire un objectif comme elle peut être la plus mauvaise pour d'autres objectifs, alors l'utilisation d'une seule règle pour ordonner les opérations n'est pas toujours efficace. Pour satisfaire le plus d'objectifs possible, plusieurs chercheurs proposent de combiner entre les règles de priorité d'autres proposent de faire une sélection dynamique de ces règles en fonction des conditions opératoires, des objectifs de production et de l'état de l'atelier. La classification des règles de priorité est la phase la plus importante pour implémenter ces approches.

Notre but est de comparer les règles de priorité, afin de les classer selon certains critères de performance, tels que le retard et le retard conditionnel. De plus, nous proposons des techniques permettant d'implémenter à partir les résultats obtenus, un système de sélection dynamique de ces règles dans un d'atelier de type Job-Shop.

III. CRITERES D'EVALUATION

A. Le temps de cycle

Temps de cycle pour le job i : $F_i = C_i - r_i$ représente le temps passé par le job i dans le système.

C_i : date de fin effective du job i .

r_i : date d'arrivée du job i

B. Les délais

Plusieurs paramètres à définir:

- Le retard algébrique représenté par : $L_i = C_i - d_i$.

d_i : date échue du job i .

Deux cas possibles:

$L_i < 0$: c'est-à-dire $C_i > d_i$, alors le job i est en avance.

$L_i > 0$: $C_i < d_i$, dans ce cas le job i est en retard.

- Le retard vrai (retard global) :

$T_i = \max(0, L_i)$, c'est le retard maximal sur l'ensemble des jobs.

- Le Nombre de job en retard : défini par $NT = \sum U_i$

$$\text{Avec } U_i = \begin{cases} 1 & \text{si } C_i > d_i \\ 0 & \text{si } C_i \leq d_i \end{cases}$$

Pourcentage de jobs en retard (Pourcentage tardy) :

$PT = NT/n$, on s'intéresse dans ce cas au nombre de jobs en retard est non pas au retard lui-même.

n : nombre total de job.

Retard conditionnel (Conditional Tardiness):

$CT = \text{Max}(0, C_i - d_i)$

La moyenne est calculée uniquement sur les jobs effectivement en retard.

IV. CONDITIONS OPERATOIRES

A. Dates d'échéance

La manière d'établir la date échue influe grandement sur les performances des règles de priorités, plusieurs méthodes ont été proposées pour le calcul des délais, le bon choix d'une méthode est important et très fragile, car plusieurs critères d'évaluation sont liés aux dates échues.

Nous utilisons la règle Total Work Content (TWK) pour le calcule les dates échues

$$TWK \quad a_j = \text{coefficient} * p_j \quad [4],[5], [6]$$

a_i délai du job i

p_i la durée totale des opérations du job i .

coefficient est une constant

B. Dates de fin au plus tard des opérations

Plusieurs auteurs ont proposé de découper, pour chaque job, le délai accordé (a_i) afin de créer pour chaque opération du job une date de fin au plus tard de l'opération, en anglais operational due date [3]. Ce qui nous permet de détecter un job qui n'est pas en retard mais risque d'être en retard, dans le cas ou les opérations déjà réalisées ont pris du retard par rapport à leur date de fin programmée, à ce moment le job doit être accéléré pour qu'il ne prenne aucun retard. La figure montre le découpage du délai, pour un job en dates de fin au plus tard des opérations. On calcule la date de fin d'une opération par la méthode TWK :

$$d_{i,j} = d_{i-1,j} + p_{i,j} a_j / \sum_i p_{i,j} \quad [6],[7].$$

Les indices (i,j) pour définir la i ème opération du job j .

$d_{0,j} = r_j$ (date d'arrivée du job j). a_j le délai autorisé du job j (allowance).

C. La charge de l'atelier

C'est le taux moyen d'utilisation des machines, c'est égal à :

$$\rho = \frac{1}{m} \sum_{i=1}^m \rho_i$$

- ρ_i , c'est la somme des durées opératoires des opérations réalisées sur la machine i / période considérée. Cette valeur représente le taux d'utilisation en régime permanent de la i ème machine ;

- m le nombre de machines dans l'atelier ;

Un autre critère peut influencer sur la performance des règles de priorité par rapport au retard, le retard sera plus ou moins important suivant la valeur de la charge, plusieurs chercheurs (Mebarki 1995 [3] ; Baker, 1984 [6]; Schultz, 1989 [7]) avaient utilisé les taux moyens d'utilisation de 80% et 90%, Un taux d'utilisation des ressources de 90% ou plus correspond à une charge élevée, un taux compris entre 80% et 90% correspond à une charge modérée.

V. LE MODELE TEST

Notre objectif est de simuler les règles de priorité les plus connues, pour cela nous avons utilisé un modèle job-shop à 4 machines.

Ramasesh (1990) [8] décrit les hypothèses habituellement faites dans les études consacrées à l'ordonnancement dynamique en Job-Shop :

- Un job ne peut allouer deux fois de suite la même machine ;
- Pas de préemption des opérations sur les machines ;
- Pas de gamme de remplacement, chaque job suivra une séquence d'opérations qui lui est propre, celle-ci pouvant être choisie aléatoirement lors de la simulation ;
- Pas de pannes sur les machines ;
- Pas de temps de changement d'outils entre les gammes ;
- Pas de temps de transport entre les machines ;
- Pas d'opérations d'assemblage.

Le modèle a été créé à l'aide du logiciel Arena, de Rockwell Software. Ce logiciel a été choisi de par sa disponibilité et son aptitude à la création d'une interface graphique pour modéliser les flux.

Ce modèle d'atelier Job-Shop présente les caractéristiques suivantes :

- Ce modèle d'atelier comporte quatre machines; chaque machine ne peut réaliser qu'une seule opération à la fois ;
- Pour chaque pièce entrant dans l'atelier, le nombre d'opérations, compris entre deux et six inclus, suit une loi discrète uniforme ;
- Pour chaque pièce entrant dans l'atelier, sa gamme est aléatoire. Les machines sur lesquelles sont réalisées les opérations sont choisies aléatoirement, avec la contrainte suivante : deux opérations consécutives ne peuvent avoir lieu sur la même machine ;
- Les durées opératoires sont aléatoires et suivent toutes une distribution exponentielle de moyenne 1. En moyenne, la durée totale des opérations d'un job est de quatre unités de temps ;
- Le processus d'arrivée des pièces est modélisé par une loi de Poisson. Du fait des particularités de ce modèle, (nombre de stations, nombre moyen d'opérations, moyen de la durée totale des opérations), la charge moyenne de l'atelier est égale au taux moyen d'arrivée des pièces ;
- Les dates échues sont établies en utilisant la méthode TWK .

VI. LA SIMULATION

Le tableau 1 représente les règles de priorité, parmi les plus connues et les plus efficaces, nous allons simuler le modèle

job-shop présenté dans les paragraphes précédentes, pour étudier les performances de ces règles en tenant compte des conditions opératoires et des critères de performances les plus utilisés.

TABLEAU 1
LISTE DES RÈGLES DE PRIORITÉ À SIMULER

Regle	Description
CR/SPT	Critical Ratio Shortest Processing Time $\text{Min}([\text{Max}(p + t_{ij}, p + t_{ij} * (d_i - p) / \sum_{j \in S_i} t_{ij})]$
EDD	Earliest Due Date: $\text{Min}(d_i)$
FIFO	First In First Out: on choisit l'opération dont l'O.F. est arrivé le premier (seule règle à respecter une priorité des O.F.).
LIFO	Last In First Out: on choisit l'opération dont l'O.F. est arrivé le premier
LUS	Lowest Utilized Station: Sélectionner le job dont l'opération suivante doit être traitée sur la ressource ayant été le moins utilisée jusqu'à présent
MOD	Modified Operation Due date: $\text{Min}[\text{Max}(d_{ij}, p + t_{ij})]$
OCR	Operation Critical Ratio: $\text{Min}(d_{ij} - p / t_{ij})$
ODD	Operation Due Date: $\text{Min}(d_{ij})$
OST	Operation Slack Time: $\text{Min}(d_{ij} - p - t_{ij})$
SCR	Smallest Critical Ratio: $\text{Min}(d_i - p / \sum_{j \in S_i} t_{ij})$
SLACK	SLACK: $\text{Min}(s_i / \sum_{j \in S_i} t_{ij})$
SPT	Shortest Processing Time: $\text{Min}(t_{ij})$.
WINQ	Work IN Queue: Sélectionner le job dont l'opération suivante doit être traitée sur la ressource ayant la plus faible durée opératoire totale en attente (pour tous les jobs en attente, somme des durées opératoires).

- i Job concerné par l'opération.
 j Rang de l'opération concerné dans le job i.
 t Temps courant.
 r_i Date d'arrivée du job i dans le système.
 d_i Date échu du job i.
 d_{ij} Date de fin au plus tard de la $j^{\text{ème}}$ opération du job i.
 p Date courante (present date).
 p_i Durée totale des opérations du job i
 t_{ij} Durée opératoire de la $j^{\text{ème}}$ opération du job i.
 S_i Ensemble des opérations restant à réaliser pour le job i
 s_i Marge du job i $s_i = d_i - p - \sum_{j \in S_i} t_{ij}$.

Nous avons déterminé à l'aide de la valeur du coefficient, trois types des dates échues, serrées dans le cas où les dates échues pour les produits sont très proches à leurs dates de

livraison et alors il y'aura un grand risque qu'un produit soit en retard, modérées dans le cas où les dates échues sont un peu loin par rapport aux dates de livraison et larges dans le cas où il y a un temps libre large entre les dates échues et les dates de livraisons.

Les différents degrés de proximité des dates échues correspondant aux différents délais d'obtention des jobs. Sont présentés dans le tableau 2.

TABLEAU 2
DEGRES DE PROXIMITE DES DATES ECHUES.

allowance	30	50	60
Charge élevée ($\rho=90\%$)	dates échues serrées (coefficient = 7.5)	dates échues modérées (coefficient = 12.5)	dates échues larges (coefficient = 15)
allowance	20	30	35
Charge modérée ($\rho=80\%$)	Dates échues serrées (coefficient = 5)	dates échues modérées (coefficient = 7.5)	dates échues larges (coefficient = 8.75)

Nous allons donc simuler $2 \times 3 = 6$ configurations possibles :

- Charge d'atelier élevée ($\rho=90\%$) avec les dates échues :
 - Serrées (allowance=30) ;
 - Modérées (allowance=50) ;
 - Larges (allowance=60) ;
- Charge d'atelier modérée ($\rho=80\%$) avec les dates échues :
 - Serrées (allowance=20) ;
 - Modérées (allowance=30) ;
 - Larges (allowance=35) ;

ρ C'est le taux moyen d'utilisation des machines.

$\rho=80\%$ correspond à une charge modérée, et 90 % pour une charge élevée

Dans le cas d'une charge élevée, nous avons utilisé trois dates d'échéance pour les jobs, 30 pour une date de livraison serrée, 50 pour une date de livraison modérée et 60 pour une date de livraison large.

Dans le cas d'une charge modérée, les dates d'échéances sont 20 pour une date de livraison serrée, 30 pour une date de livraison modérée et 35 pour une date de livraison large.

Pour chacune des six configurations testées, dix répliques indépendantes sont réalisées. Pour chaque réplique, la durée transitoire est estimée à 500 jobs, et les mesures de performances sont collectées pour 5000 jobs. Les mesures de performance estimées sont le temps de cycle moyen (\bar{F}), le retard moyen (\bar{T}), le retard conditionnel moyen (\bar{CT}) et le pourcentage moyen de retard PT . Ces mesures sont calculées sur dix répliques indépendantes.

Nous avons retenu quatre mesures de performances suivantes :

- La moyenne du temps de cycle, souvent utilisée pour évaluer la performance des ateliers ;
- La moyenne du retard, qui est un des critères les plus utilisés dans la littérature pour mesurer le comportement de l'atelier par rapport aux dates échues ;
- La moyenne du retard conditionnel, qui est très importante pour juger de l'efficacité de l'ordonnement par rapport au retard des pièces ;
- Le pourcentage de jobs en retard.

Nous présentons dans le tableau 3, les trois meilleures règles pour chaque configuration, ainsi que les valeurs numérique obtenus par la simulation pour chaque critère, par rapport à la moyenne du temps de cycle, le retard moyen, le retard conditionnel moyen et le pourcentage des retards.

TABLEAU 3
LES RESULTATS DE LA SIMULATION.

Par rapport au temps de cycle moyen \bar{F}				
	Charge élevée		Charge Modéré	
Dates échues serrées	SPT	16,53	SPT	11,49
	LUS	18,92	LUS	12,6
	WINQ	20,3	CR/SPT	12,79
Dates échues modérées	SPT	16,51	SPT	11,49
	LUS	18,90	LUS	12,6
	WINQ	21,13	CR/SPT	13,16
Dates échues larges	SPT	16,53	SPT	11,49
	LUS	18,92	LUS	12,6
	CR/SPT	21,03	CR/SPT	13,63
Par rapport au retard moyen \bar{T}				
Dates échues serrées	CR/SPT	0,91	MOD	0,43
	MOD	1,3	CR/SPT	0,6
	OCR	1,7	OCR	0,86
Dates échues modérées	CR/SPT	0,03	CR/SPT	0,03
	MOD	0,15	MOD	0,1
	OST	0,15	EDD	0,23
Dates échues larges	CR/SPT	0,01	CR/SPT	0,02
	OST	0,01	EDD	0,04
	SLACK	0,01	MDD	0,06
Par rapport au retard conditionnel moyen \bar{CT}				
Dates échues serrées	OCR	4,38	OCR	2,6
	CR/SPT	7,31	CR/SPT	3,27
	SCR	7,52	SCR	3,76
Dates échues modérées	CR/SPT	1,74	OCR	1,55
	MOD	2,48	SCR	1,64
	OCR	2,69	CR/SPT	2,05
Dates échues larges	SCR	1,12	SCR	1,31
	CR/SPT	1,39	OCR	1,56
	OCR	1,57	CR/SPT	1,83

Par rapport au pourcentage de job en retard \bar{PT}				
Dates échues serrées	SLACK	7,13	CR/SPT	8,34
	WINQ	8,42	SPT	9,52
	LUS	8,75	MOD	10,17
Dates échues modérées	CR/SPT	0,91	CR/SPT	1,51
	MOD	1,55	MOD	2,41
	SPT	2,54	SPT	3,08
Dates échues larges	CR/SPT	0,27	CR/SPT	0,92
	SLACK	0,43	MOD	1,1
	ODD	0,68	EDD	4,42

A partir de ces résultats, nous remarquons que la règle SPT figure comme la meilleur règle pour réduire la moyenne du temps de cycle dans toutes les configurations, quelque soit la charge d'atelier élevée ou modérée et les dates échues serrées, modérées, ou larges. Ceci est dû à sa capacité d'accélérer la grande majorité des jobs, donc réduire le temps de cycle. Notons que les règles LUS, WINQ, CR/SPT donnent des résultats acceptables, les règles FIFO, LIFO figurent comme mauvaises stratégies.

Pour réduire la moyenne de retard, les règles basées sur les dates de fin au plus tard des opérations, MOD, ODD, OCR, OST. Donnent de très bons résultats, selon Baker (1984) [6], la règle MOD apparaît comme la plus efficace pour réduire le retard moyen. Dans notre simulation, la règle CR/SPT figure comme la plus performante par rapport au retard moyen, pour des conditions opératoires très variées.

Pour le retard conditionnel moyen, les règles basées sur le ratio critique (SCR, OCR, CR/SPT) sont les plus performantes. Par contre, La règle SPT figure comme une mauvaise règle par rapport au retard conditionnel moyen. La règle SPT a pour effet d'accélérer le flux et donc de diminuer la moyenne des retards. Mais si un retard va se produire, il sera très important concernant les opérations à longue durée opérationnelle.

Concernant le pourcentage de jobs en retard, la conclusion des auteurs semble unanime: la meilleure règle est sans conteste CR/SPT. Ceci est dû à sa faculté d'accélérer la grande majorité des jobs, donc à générer relativement peu de jobs en retard. D'autre part, n'oublions pas que les dates d'échéance internes sont calculées à partir des temps opératoires. Cette capacité à produire peu de jobs en retard explique ainsi sa bonne performance par rapport au retard moyen. Néanmoins, avec la règle SPT, si très peu de jobs sont en retard, ceux qui le sont ont souvent des retards très importants.

La classification sert à choisir la règle la plus performante dans une situation donnée, par exemple :

SI [OBJECTIF = Réduire le retard] et CONDITIONS OPERATOIRES = Dates échues larges] ALORS [Appliquer SPT].

Cette condition est appelée une règle de production qu'on peut l'implémenter dans un système intelligent de sélection dynamique des règles de priorité. A partir de la simulation plusieurs règles peuvent être tiré, signalons qu'on peut tirer à

partir de l'étude bibliographique d'autres règles en se basant sur les caractéristiques de la règle de priorité par exemple :

Si [un job est trop retardé] alors [appliquer FIFO], la règle FIFO donne l'avantage au job les plus anciens.

Pour sélectionner une règle dynamiquement (DR), le programme prend en considération les conditions opératoires (La charge d'atelier et les dates échues), les objectifs fixés (par exemple réduire le temps de cycle moyen), et l'état du poste de travail (par exemple la longueur des files d'attente), le système doit réagir en temps réel à chaque événement inattendu dans l'atelier est appliquer la meilleur règle qui peut diminuer l'effet de cet événement.

VII. CONCLUSION ET PERSPECTIVE

Dans ce travail, nous avons simulé les règles de priorité les plus utilisées, sur un modèle job-shop, nous avons classifié ces règles selon des différents objectifs et conditions opératoires, nous avons définis quatre objectifs, réduire le temps de cycle moyen, réduire le retard moyen, réduire le retard conditionnel moyen et réduire le pourcentage des retard, nous avons remarqué à partir des résultats obtenues qu'une seule règle de priorité figure comme la plus performante dans certaine conditions et mauvaise dans d'autres conditions d'où la nécessité de changer dynamiquement la stratégie de gestion des files d'attente selon l'état de l'atelier.

Notre perspective et d'implémenter un système automatique pour la sélection dynamique des règles de priorité, basée sur les techniques de l'intelligence artificielle pour piloter en temps réel, testé sur un vrai FMS.

REFERENCES

- [1] C.Artigues , "Ordonnancement en temps réel d'atelier avec prise en compte des temps de préparation des ressources", thèse PhD , Laboratoire d'Analyse et d'Architecture des Systemes du C.N.R.S, INSA,1998.
- [2] G.Doumeings, D.Breuil, et L.Pun, "La gestion de production assistée par ordinateur", Hermès, 1983.
- [3] N.Mebarki, "Une approche d'ordonnancement temps réel basée sur les règles de priorité des files d'attente", PhD thèse, Université de Claude Bernard Lyon1 1995.
- [4] Conway, R.W., 1965, "Priority dispatching and job lateness in a job shop", *Journal of Industrial Engineering*, Vol. 16, No. 4, 288-237.
- [5] A.S.Kiran, M.S.Smith, "Simulation studies in job shop scheduling: a survey", *Computers and Industrial Engineering*,1984, Vol. 8, No. 2, 46-51.
- [6] K.R.Baker, "Introduction to sequencing and scheduling", John & Wiley, 1974.
- [7] C.R.Schultz, "An expediting heuristic for the shortest processing time dispatching rule", *International Journal of Production Research*,1989, Vol. 27, No. 1, 31-41.
- [8] R.Ramasesh, "Dynamic Job Shop Scheduling: A Survey of Simulation Research", *OMEGA*, 1990 Vol. 18, No. 1, 43-57.