

Heuristique pour l'ordonnancement cyclique multi-produits des lignes de traitement de surface

Adnen ELAMRAOUI^{1,2}, Marie-Ange MANIER¹, Abdellah EL MOUDNI¹, Mohamed BENREJEB²

¹Laboratoire Systèmes et Transport,
Université de Technologie Belfort-Montbéliard - France
UTBM Site de Belfort 90010 Belfort Cedex, France.

adnen.elamraoui@utbm.fr, marie-ange.manier@utbm.fr, abdellah.el-moudni@utbm.fr

²Unité de recherche LARA Automatique,
Ecole Nationale d'Ingénieurs de Tunis - Tunisie
BP 37, 1002 Belvédère, Tunis, Tunisie.
[Adnen.ElAmraoui @enit.rnu.tn](mailto:Adnen.ElAmraoui@enit.rnu.tn), mohamed.benrejeb@enit.rnu.tn

Résumé— Dans cet article nous nous intéressons au problème d'ordonnancement cyclique multi-produits des lignes de traitement de surface. La commande cyclique est adaptée aux commandes de grandes séries vu l'avantage qu'elle fournit en réduisant la combinatoire de résolution, dès lors que les ratios de produits sont connus à l'avance. En effet, ce type de commande nous permet d'éviter d'ordonnancer la totalité des opérations et de se limiter à un motif de taille réduite, qui va se répéter autant de fois que nécessaire pour réaliser la commande totale.

L'optimisation de la production pour un tel type d'atelier consiste à rechercher le meilleur ordonnancement des tâches à accomplir en un temps raisonnable.

Nous proposons une nouvelle approche heuristique pour la résolution de ce problème, plus connu dans la littérature sous l'appellation "Cyclic Hoist Scheduling Problem" (CHSP).

Mots-clés— Ordonnancement Cyclique, Ligne de Traitement de Surface, Heuristique, Robot de Manutention, Multi-Produits.

I. INTRODUCTION

Une ligne de traitement de surface est constituée d'une succession de cuves dans lesquels une opération chimique, de durée strictement définie sur un intervalle de temps [min, max], doit être réalisée.

Ce type de ligne est en particulier contraint par un ou plusieurs robots, se déplaçant sur un rail au dessus des cuves et assurant le transport des produits à traiter. Le problème d'ordonnancement consiste alors à trouver une séquence répétitive des mouvements du ou des robots de durée (temps de cycle) minimale.

La gamme opératoire est la séquence des opérations à effectuer pour traiter un objet : elle définit ainsi une suite de cuves de traitement, avec une fourchette de temps de traitement pour chaque bain et pour chaque produit.

Les problèmes d'ordonnancement d'ateliers sont connus pour être complexes et hautement combinatoires [1] et il a été prouvé [2] que le problème cyclique mono-produit dans le cas des ateliers de traitement de surface est déjà NP-difficile avec un seul type de produit et une seule ressource de transport. Ainsi la forte complexité du problème nous a incités à avoir recours à une approche heuristique pour sa résolution.

Cet article traite le problème de pilotage cyclique des lignes de traitement de surface lorsque l'on traite plusieurs produits ayant différentes gammes opératoires. L'originalité des

travaux, présentés dans ce papier réside dans le fait que, nous cherchons un cycle dont le degré est inconnu a priori.

Le reste de notre article est organisé de la façon suivante. Dans la partie 2, nous commençons par présenter les spécificités du problème d'ordonnancement des lignes de traitement de surface. Un bref état de l'art sur les travaux réalisés sur le problème CHSP multi-produits est présenté dans la section 3. Dans la section 4, une analyse mathématique du problème est proposée. Nous présentons dans la section 5 une nouvelle heuristique baptisée "Cyclic Multi-Parts Earliest Starting Time" (CMPEST). Des résultats et des comparaisons sur des exemples sont donnés en section 6. Et en conclusion, des pistes de recherches prometteuses pour des futurs travaux sont envisagées.

II. LIGNES DE TRAITEMENT DE SURFACE : SPECIFICITES

L'automatisation des lignes de traitement de surface vise à maximiser la productivité tout en utilisant les ressources (cuves, robots et bains) au maximum de leurs capacités et sans avoir à dégrader la qualité des produits (respect de la tenue et de l'épaisseur des dépôts sur les produits). Pour ce faire, une bonne planification des tâches à accomplir s'avère impérative.

Ce problème revient à ordonnancer au mieux les mouvements du ou des robots de la ligne afin d'optimiser le ou les critères choisis, tout en respectant les contraintes auxquelles est soumis notre système (gamme, capacités).

Fig. 1 illustre l'exemple d'une ligne de traitement de surface.

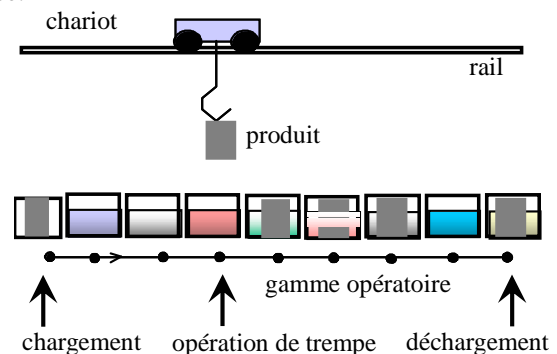


Fig. 1. Exemple d'une ligne de traitement de surface

Dans cet article, nous considérons le cas de lignes simples comportant un seul robot, des cuves de capacité unitaire (dites mono-bac) dans lesquelles chaque produit passe une seule fois au cours de son traitement (cuves mono-fonction).

Le problème d'ordonnement que nous traitons est connu dans la littérature sous l'appellation du "Hoist Scheduling Problem" (HSP). En plus des contraintes de capacité des ressources de traitement et de transport, il présente des spécificités qui le distinguent des autres types d'ordonnement d'ateliers :

- (H1) : les durées de trempage dans les cuves sont bornées par valeur inférieure et supérieure.
- (H2) : les temps de transport ne sont pas négligeables par rapport aux durées d'immersion.
- (H3) : les robots ne doivent pas entrer en collision.
- (H4) : un robot doit trouver le temps pour se déplacer à vide entre deux opérations de transport : entre la cuve où il vient de déposer un porteur et la cuve suivante d'où il doit retirer un autre porteur.

Il n'y a ici ni attente dans les baignoires, ni stock intermédiaire entre les cuves.

Le problème cyclique des lignes de traitement de surface connu sous le nom du "CHSP" est une classe du problème "HSP", où on se limite à étudier une séquence de mouvement des robots de manutention [3], [4]. L'ordonnement cyclique recherché doit spécifier la séquence des mouvements ainsi que les dates d'exécution de ces mouvements.

Vu la complexité du problème due notamment aux contraintes et à la taille du système physique, on se propose dans la suite de nous limiter à l'étude du cas des lignes avec un seul robot et des cuves à capacité unitaire.

III. ETAT DE L'ART

L'ordonnement 1-cyclique mono-produit des mouvements du ou des robots dans une ligne de traitement de surface est le problème le plus étudié dans la littérature [3], [4], [5], [6], [7], [8], [9]. Toutefois, quelques travaux se sont intéressés à l'étude du problème cyclique multi-produits. Dans ce cas, une ligne de production peut traiter simultanément deux ou plusieurs produits sans nécessiter de vider au préalable la ligne de traitement. Cela lui confère une certaine flexibilité.

Ptuskina propose une approche qui cherche à minimiser le critère C_{max}/T (où C_{max} est le makespan et T : le temps de cycle) en considérant le cas de traitement de n produits différents. Toutefois, il suppose le cas particulier où tous ces produits possèdent la même gamme opératoire et la seule différence réside dans le temps de trempage. L'auteur suppose que la gamme opératoire est définie au préalable contrairement aux dates d'entrées des produits sur la ligne de traitement qui doivent être déterminées. La méthode proposée par Ptuskina consiste à subdiviser le problème multi-produit en n problèmes mono-produit et à adapter une période commune à tous ces problèmes comme solution du problème multi-produits considéré [10].

Varnier et al. proposent une méthode de type branch and bound qui permet d'obtenir un ordonnancement cyclique pour les divers produits traités sur la ligne. Dans cette optique, ils proposent de mixer les ordonnancements cycliques mono-

produits et considèrent toutes les combinaisons possibles. Ils se focalisent sur un problème de changement de type de produit en régime transitoire dans le cas 1-cyclique [11].

Mangione propose une analogie entre les problèmes de flow-shop robotisé sans attente et le problème du CHSP. Pour cela, il considère le cas d'une ligne à deux cuves et applique l'algorithme de Gilmore et Gomory pour obtenir une solution réalisable du cas multi-produits. Bien que cette solution soit réalisable, elle n'est pas forcément optimale [12].

Dans [13], les auteurs proposent une procédure "branch and bound" pour la détermination d'un ordonnancement 2-cyclique. L'objectif de l'optimisation est la minimisation du temps de cycle total. Ils considèrent le cas de deux types de produits mais ayant la même séquence opératoire et seules les fenêtres de temps liées aux durées peuvent varier.

Pour le même problème, El Amraoui et al. proposent une formulation linéaire en nombres entiers et binaires pour minimiser le temps de cycle pour un cycle de degré 2 et qu'ils envisagent de généraliser pour le cas général n -cyclique [14], [15].

IV. ANALYSE MATHÉMATIQUE DU PROBLÈME

On propose la notation suivante pour analyser le problème considéré :

- np : nombre de produits de la liste P ,
- nt^k : nombre de traitements de la gamme du produit k , ($1 \leq k \leq np$)
- o_i^k : durée de la $i^{\text{ème}}$ opération de trempage du produit k , ($1 \leq k \leq np$ et $1 \leq i \leq nt^k$)
- do_i^k : début de la $i^{\text{ème}}$ opération de trempage du produit k , ($1 \leq k \leq np$ et $1 \leq i \leq nt^k$)
- fo_i^k : fin de la $i^{\text{ème}}$ opération de trempage du produit k , ($1 \leq k \leq np$ et $1 \leq i \leq nt^k$)
- a_i^k : borne inférieure de la $i^{\text{ème}}$ opération de trempage du produit k , ($1 \leq k \leq np$)
- b_i^k : borne supérieure de la $i^{\text{ème}}$ opération de trempage du produit k , ($1 \leq k \leq np$)
- d_i^k : durée de l'opération de transfert du produit k de la cuve de départ correspondant au traitement i à la cuve d'arrivée correspondant au traitement $i+1$, ($1 \leq k \leq np$ et $1 \leq i \leq nt^k$), incluant les temps :
 - de prise du porteur (levée),
 - d'égouttage au dessus de la cuve d'arrivée,
 - de transport du produit de la cuve de départ à la cuve d'arrivée,
 - de stabilisation au dessus de la cuve d'arrivée,
 - de déchargement du porteur dans la cuve d'arrivée (descente).
- dt_i^k : date de début de l'opération de transport du produit k de la cuve de départ correspondant au traitement i à la cuve d'arrivée correspondant au traitement $i+1$ de la même gamme opératoire, ($1 \leq k \leq np$)
- ft_i^k : date de fin de l'opération de transport du produit k de la cuve de départ correspondant au traitement i à la cuve d'arrivée correspondant au traitement $i+1$ de la même gamme opératoire, ($1 \leq k \leq np$)
- $c_{i,j}^{k,l}$: durée de transport à vide du chariot de la cuve d'arrivée du $i^{\text{ème}}$ traitement du produit k jusqu'à la cuve de départ du $j^{\text{ème}}$ traitement du produit l , ($1 \leq k, l \leq np$ et $1 \leq i, j \leq nt^k$)
- T : temps de cycle du cas multi-produits.

O_i^k : i^{ème} opération de transport en charge du produit k de la cuve de départ correspondant au traitement i à la cuve d'arrivée correspondant au traitement i+1.

Compte tenu que ce problème est sans attente et sans stock, ordonnancer les opérations de trempe équivaut à ordonnancer les opérations de transport. Ainsi, le début d'une opération de trempe correspond à la fin d'une opération de transport en charge et la fin d'une opération de transport en charge correspond au début d'une opération de trempe (fig. 2).

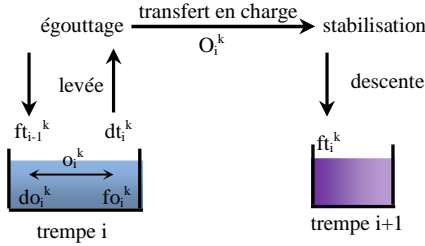


Fig. 2. Analyse du mouvement en charge du chariot

Les relations suivantes sont donc vérifiées :

$$dt_i^k = fo_i^k \quad (1)$$

$$fo_i^k = do_i^k + o_i^k \quad (2)$$

$$ft_i^k = do_{i+1}^k \quad (3)$$

Le robot doit être disponible chaque fois qu'une opération de trempe se termine dans une cuve, aucun retard (ni aucune avance) n'étant admis, et aucune tâche ne pouvant être préemptée (aucune pause en charge n'est autorisée).

Dans la recherche de la solution cyclique multi-produits, les inconnues sont les durées réelles des opérations de trempe o_i^k , les dates de début des opérations de transport en charge du chariot dt_i^k , ainsi que la durée du temps de cycle T.

N le nombre total de variables de notre problème a pour valeur:

$$N = 1 + \sum_{k=1}^{np} 2 \cdot nt^k \quad (4)$$

Toutefois, si l'on considère que l'on peut déduire les durées o_i^k à partir des dates dt_i^k alors le nombre de variables du problème peut être réduit à:

$$N = 1 + \sum_{k=1}^{np} nt^k \quad (5)$$

V. HEURISTIQUES

A. Heuristique 1 : "EST"

Nous présentons tout d'abord l'approche heuristique de Song, Sabinsky et Storch dont nous nous sommes inspirés

pour la résolution du problème cyclique multi-produits des lignes de traitement de surface [16].

Cette approche est elle-même une adaptation de l'heuristique d'insertion au plus tôt des produits à traiter, plus connu en littérature sous le nom du "Earliest Starting Time" [17].

Les auteurs se sont limités à l'étude du cas mono-produit et pour simplifier leur modèle de résolution, ils ont fixé les durées opératoires aux bornes inférieures des fenêtres de temps. De ce fait ils ne traitent pas un problème de type HSP au sens strict du terme, mais un problème connexe de type "Cyclic robotic scheduling".

De plus, l'exemple illustratif proposé ci-dessous montre l'intérêt de la flexibilité accordée par les fenêtres de temps et notamment dans le cas multi-produits.

Considérons l'exemple (table I et II, fig. 3) de deux produits traités sur une ligne à cinq cuves incluant les cuves de chargement (cuve 0) et de déchargement (cuve 4) et possédant la même gamme opératoire : d'abord la cuve 1 puis la cuve 2 et enfin la cuve 3 avant d'être mis à la cuve de déchargement. Les fenêtres de temps de chaque type de produit sont données dans table 1.

produit	cuve 1	cuve 2	cuve 3
1	80 – 96	41 – 46	61 – 66
2	76 – 91	21 – 26	30 – 66

Le mouvement à vide du chariot entre deux cuves successives est de 5 unités de temps et de 10 u.t. pour un mouvement en charge (incluant les mouvements de stabilisation et d'égouttage).

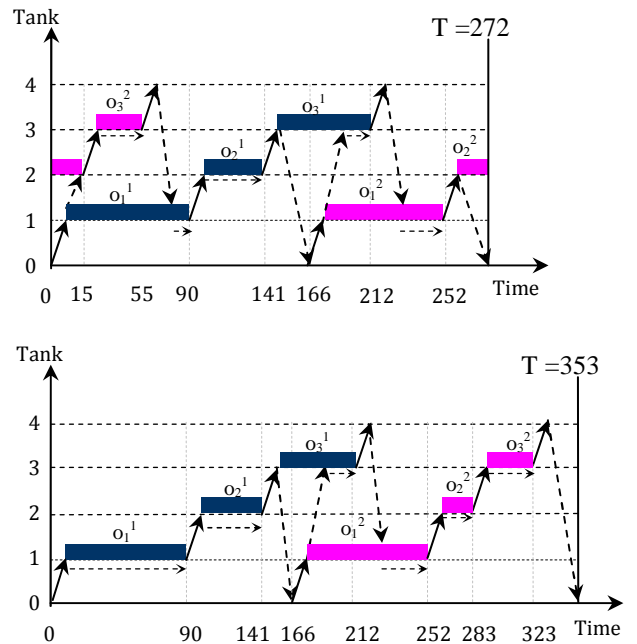


Fig. 3. Réduction du temps de cycle en utilisant la flexibilité de la durée opératoire

Dans fig. 3, les flèches en trait continu représentent les mouvements en charge du chariot alors que les flèches en discontinu obliques représentent les déplacements à vide du chariot et les discontinus horizontaux symbolisent les temps d'attente à vide du chariot au dessus des cuves de chargement.

TABLE II

RESULTATS : DUREES OPERATOIRES DE LA SOLUTION OPTIMALE

produit	cuve 1	cuve 2	cuve 3
1	80	41	61
2	76	25	30

Les durées opératoires pour la solution optimale sont égales aux bornes minimales des fenêtres de temps excepté pour la seconde opération de trempe du second produit (voir table II). Ainsi, pour une durée opératoire comprise dans la fenêtre de temps et supérieure à la borne minimale on a réduit le temps de cycle de 81 u.t. soit à 23% de la valeur du temps de cycle.

B. Heuristique 2 : "CMPEST"

Tenant compte de la complexité du problème cyclique multi-produits et du grand écart entre la solution optimale et la solution fournie par l'heuristique "EST" de [16], nous proposons une nouvelle approche de résolution qui se base sur la flexibilité offerte par les durées opératoires bornées et non fixées. L'originalité de cette approche réside dans le fait que le degré du cycle est non fixé à l'avance.

L'idée de base de notre heuristique, baptisée "CMPEST" (Cyclic Multi-Parts Earliest Starting Time) consiste à imposer une séquence d'entrée cyclique pour les produits à traiter (par exemple A-B/A-B/.../A-B). L'insertion des produits sur la ligne de traitement est basée sur le principe d'insertion au plus tôt. Soumis à des contraintes de disponibilité des ressources et des durées opératoires, un régime cycle permanent est établi, pour ce système de données. Ce régime va nous fournir le degré et le temps du cycle.

L'algorithme que nous avons développé est basé sur un algorithme de liste où à chaque étape une liste de solutions candidates sont testées ; puis une seule opération est rajoutée à la liste des solutions selon un ordre de priorité bien défini et explicité dans les procédures suivantes :

Procédure d'initialisation

```
/*Initialisation de la première liste des solutions candidates*/
L1 = <O11>
```

Procédure de passage à l'étape k+1

```
/* A chaque étape, l'itération k+1 est formée ainsi */
- Tous les éléments de Lk sont copiés dans Lk+1
- Oij, l'opération i du produit j, choisie à l'itération k est enlevée de la liste Lk+1.
- La date dtkp pour chaque opération Okp dans la liste Lk+1 est actualisée par l'expression suivante:
max { dtkp, ftij + ci+1,kj,p } (6)
```

Procédure d'insertion de nouvelles opérations candidates

```
/* Insertion de nouvelles opérations candidates selon la solution choisie à l'itération précédente : Oij */
```

- Si l'opération O_i^j n'est pas la dernière opération du produit j alors insérer l'opération O_{i+1}^j dans la liste L^{k+1} et calculer le dt_{i+1}^j par la formule suivante:

$$dt_{i+1}^j = ft_i^j + a_{i+1}^j \quad (7)$$

- Si l'opération O_i^j est pas la première opération du produit j (i=0) alors insérer l'opération O_{i+1}^j et la première opération du produit suivant dans la séquence cyclique d'entrée, définie précédemment (O₀^{j+1}) dans la liste L^{k+1} et actualiser leurs dates de début respectivement par les équations suivantes :

$$dt_{i+1}^j = ft_i^j + a_{i+1}^j \quad (8)$$

$$dt_{i+1}^{j+1} = ft_i^j + c_{i+1,0}^{j,j+1} \quad (9)$$

L'algorithme de l'heuristique "CMPEST"

Procédure d'initialisation.

Tant que (il reste des opérations de transport dans la séquence d'entrée cyclique) faire

Procédure de passage à l'étape k+1,

Procédure d'insertion de nouvelles solutions candidates,

Trier la liste L^{k+1} par ordre croissant des temps de début des opérations de transport,

Choisir la première opération de la liste triée comme solution candidate,

Si (la cuve d'arrivée de la solution candidate est libre)

Si (la durée opératoire de l'opération de transport vérifie la contrainte de fenêtre de temps)

Alors choisir la première solution candidate de la liste.

Si non

Tant que (pas de solution) faire

Retour à la liste précédente L^k et choisir la solution candidate suivante.

Fin Tant que

Si non

Alors choisir la solution candidate suivante de la liste L^{k+1}.

Fin Si

Fin Si

/* contrôle de la contrainte de fenêtres de temps */

Vérifier que la durée de trempage pour chaque solution candidate vérifie l'équation suivante :

$$dt_i^j \leq do_i^j + b_i^j \quad (10)$$

C. Exemple illustratif pour la "CMPEST"

On considère le même exemple, présenté au paragraphe VI mais avec les fenêtres de temps de la table III.

TABLE III
DONNEES : FENETRES DES TEMPS OPERATOIRES

produit	cuve 1	cuve 2	cuve 3
1	40 – 50	30 – 45	60 – 80
2	30 – 40	40 – 50	50 – 70

Pour cet exemple, on impose une séquence d'entrée cyclique, des produits de type 1 et 2 à traiter. Soit la séquence suivante: job1-job2-job1- job2-job1- job2... Et le quadruplet (j, c, n, dt_i^k) , définissant l'opération de transport en charge du produit de type j et de rang n dans la séquence cyclique d'entrée de la cuve c à la cuve suivante.

Le première liste d'initialisation contient la première opération de transport du produit de type 1 de la cuve 0 à la cuve 1. Le temps de début de cette opération de transport est pris comme origine des temps, soit $(1, 0, 1, 0)$. Cette solution candidate est choisie comme solution définitive et la nouvelle liste de solutions candidates contient deux nouvelles opérations : $(1, 1, 1, 40)$ et $(2, 0, 2, 15)$. Après un tri par ordre croissant des temps de début, on optera pour la deuxième option. Toutefois, la cuve 1 est déjà occupée par le produit de type 1 et donc on devrait choisir la première solution candidate. La nouvelle liste contient les deux options suivantes $(1, 2, 1, 80)$ et $(2, 0, 2, 60)$. D'après le critère de sélection adopté on choisira la solution $(2, 0, 2, 60)$. En se basant sur le critère d'insertion, la nouvelle liste de solutions candidates contient trois options : $(1, 2, 1, 80)$, $(2, 1, 2, 100)$ et $(1, 0, 3, 75)$. La première cuve est déjà occupé par le produit de type 2 donc on optera pour la solution $(1, 2, 1, 80)$. La nouvelle liste contient alors: $(1, 3, 1, 150)$, $(2, 1, 2, 100)$ et $(1, 0, 3, 90)$. La cuve 1 est toujours occupée donc on choisit la solution $(2, 1, 2, 100)$ et à l'itération suivante on a le choix parmi les possibilités suivantes : $(1, 3, 1, 150)$, $(2, 2, 2, 130)$ et $(1, 0, 3, 120)$. La cuve 1 est libre donc on retient la solution $(1, 0, 3, 120)$. La nouvelle liste contient : $(1, 3, 1, 150)$, $(2, 2, 2, 150)$ et $(1, 1, 3, 170)$ et $(2, 0, 4, 125)$, or les cuves 1 et 3 sont occupées donc on choisit la solution $(1, 3, 1, 150)$. La nouvelle liste contient : $(2, 2, 2, 170)$ et $(1, 1, 3, 175)$ et $(2, 0, 4, 180)$; or en calculant la durée de trempage du produit de type 2 dans la cuve 2 on trouve 60 u.t. et donc soit un dépassement de 10 u.t. du temps opératoire maximum autorisé. Ainsi un retour arrière (backtrack) s'impose. La liste des solutions candidate à l'itération précédente contient : $(1, 3, 1, 150)$, $(2, 2, 2, 150)$ et $(1, 1, 3, 170)$ et $(2, 0, 4, 125)$, or les cuves 1 et 3 sont occupées donc un nouveau "backtrack" s'impose. Et la procédure se répète dans ce cas jusqu'à ce qu'on arrive à la

première liste : $(1, 2, 1, 80)$ et $(2, 0, 2, 60)$ et dans ce cas, on choisit la $(1, 2, 1, 80)$ au lieu de la $(2, 0, 2, 60)$. Et l'algorithme prend fin quand toutes les opérations de transport en charge sont ordonnées.

Finalement, on définit notre cycle ou k -cycle (ou encore cycle k -périodique), comme le plus petit entier k , où dans une séquence cyclique d'activités C_k , chaque activité (opération de transport en charge) est réalisée k fois ($k/2$ fois pour le produit A et $k/2$ fois pour le produit B).

VI. RESULTATS DES SIMULATIONS

Dans la littérature, il existe très peu de benchmarks sur le HSP, et encore moins sur le CHSP multi-produits. Nous avons donc en majorité validé notre heuristique (de fig. 4 à fig. 6) pour des instances proposées par Matéo [18]. A notre connaissance, ce sont les seuls instances disponibles traitant du CHSP multi-produits suivant la même séquence opératoire mais avec des bornes de durées différentes [18].

Table IV présente les résultats de quelques unes des instances que nous avons simulées.

TABLE IV
SOLUTIONS OPTIMALES 2-CYCLIQUE ET SOLUTIONS OBTENUES PAR L'HEURISTIQUE "CMPEST"

Référence exemple	Solution Optimale			Solution "EST"			Solution "CMPEST"		
	N	T	T/n	N	T	T/n	N	T	T/n
[18]									
117071706	2	362	181	2	746	373	4	700	175
125043007	2	653	326,5	2	1115	557,5	6	1868	311
137073308	2	583	291,5	2	1157	578,5	10	2858	285
140043108	2	585	292,5	2	1077	538,5	6	1694	282
256085510	2	756	378	2	1410	705	6	2128	354

L'analyse de ces résultats montre clairement que nous avons réussi à améliorer le temps de cycle moyen pour des degrés de cycles supérieurs à deux.

Les simulations présentées ci-dessous sont les résultats de plus de 150 exemples correspondants à diverses fourchettes de fenêtres de temps et des vitesses de chariot.

Les fourchettes de fenêtres de temps peuvent être définies par une relation liant la borne min à la borne max. Les bornes min des fenêtres de temps ont été générées par une distribution uniforme entre 20 et 80.

Les figures 4 et 5 présentent les taux d'écart moyens par rapport au temps de cycle moyen optimal pour les heuristiques "EST" et "CMPEST" en fonction du nombre de cuves de traitement.

Alors que les résultats de l'heuristique du "EST" présentent un grand écart par rapport à la solution optimale atteignant les 97,61% pour le cas des lignes à 7 cuves et les 86% en moyenne, notre heuristique (CMPEST) n'est qu'à 7,54% en moyenne du temps de cycle moyen optimal.

Au vu des résultats de l'heuristique "CMPEST", on remarque une nette amélioration du temps de cycle moyen par rapport à la première heuristique atteignant les 39,46% en moyenne (voir fig. 6).

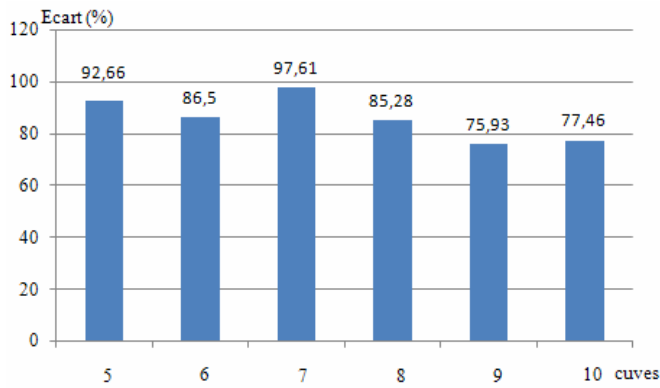


Fig. 4. Ecart (%) de la solution de l'heuristique "EST" par rapport au temps de cycle moyen optimal

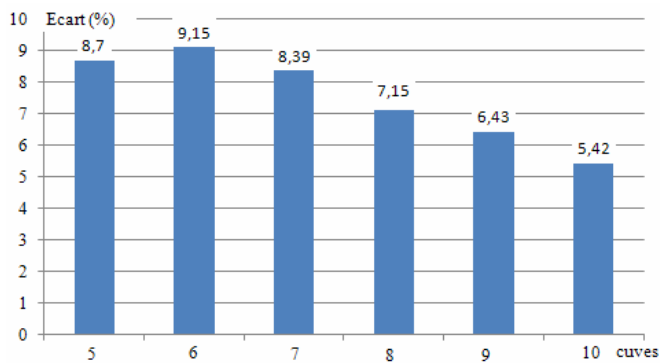


Fig. 5. Ecart (%) de la solution de l'heuristique "CMPEST" par rapport au temps de cycle moyen optimal

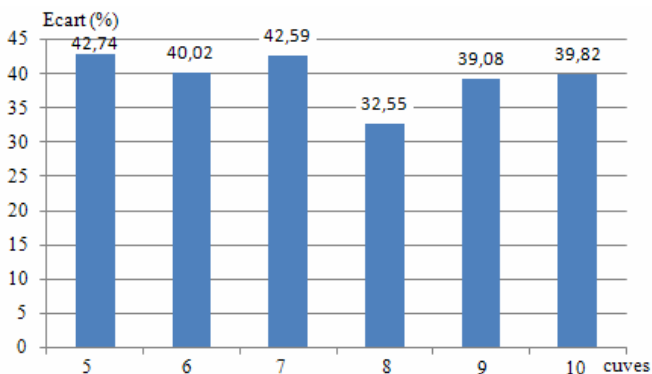


Fig. 6. Ecart (%) entre les solutions des heuristiques "EST" et "CMPEST"

VII. CONCLUSION

L'objectif recherché par le problème cyclique d'ordonnement des lignes de traitement de surface est de minimiser à la fois, le temps de cycle et la combinatoire de résolution. Pour ce fait, nous avons présenté une approche heuristique originale qui vise à minimiser le temps de cycle pour le cas multi-produits, peu abordé jusqu'à présent.

Nous avons mis en valeur l'intérêt que présente notre heuristique (CMPEST) par rapport à l'heuristique (EST) qui traite en réalité un cas connexe du problème "CHSP". En outre, cette nette amélioration que nous avons réussi à atteindre est due aux plages [min, max] des opérations de trempe.

Nous avons également, mis l'accent sur l'écart du temps de cycle moyen obtenu par l'heuristique CMPEST par rapport au temps de cycle moyen optimal qui reste en moyenne inférieur à 10% pour l'ensemble des instances générées.

Dans la suite de nos travaux, nous envisageons d'étudier l'influence du panachage sur le temps de cycle. Une amélioration de l'algorithme "CMPEST" ainsi que des comparaisons avec des méthodes approchées et notamment les algorithmes génétiques sont en cours.

RÉFÉRENCES

- [1] Bellmann, R., Esogbue, A., Nabeshima, I. *Mathematical Aspects of Scheduling and Applications*. Pergamon Press, 1982.
- [2] Lei, L., Wang, T.J. A Proof: The Cyclic Hoist Scheduling Problem is NP-Complete, Working Paper, n° 89-0016. Rutgers University. New Jersey, 1989.
- [3] Armstrong, R., Lei, L., Gu, S. A Bounding Scheme for Deriving the Minimal Cycle Time of a Single-Transporter N-Stage Process with Time-Window Constraints. *European Journal of Operational Research*, vol. 75, pp. 1-11, 1994.
- [4] Phillips, L.W., Unger, P.S. Mathematical Programming Solution of a Hoist Scheduling Program. *IIE Transactions*, vol. 8, n° 2, pp. 219-225, 1976.
- [5] Manier, M.A., Bloch, C. A classification for Hoist Scheduling Problem. *The International Journal of Flexible Manufacturing Systems*, vol. 15, n° 1, Edition Kluwer, academic Publishers, pp. 37-55, 2003.
- [6] Zhou, Z., Ling, L. Single hoist cyclic scheduling with multiple tanks: a material handling solution. *Computers & Operations Research*, vol. 30, pp. 811-819, 2003.
- [7] Lei, L. Determining optimal cyclic hoist Schedules in a single-hoist electroplating line. *IIE Transactions*, vol. 26, pp. 25-33, 1994.
- [8] Baptiste, P., Legeard B., Manier M.A., Varnier C. Résolution d'un problème d'Ordonnement avec la PLC, *Journal Européen des Systèmes Automatisés. Intelligence Artificielle et Automatique*, vol. 30, pp. 201-230, 1996.
- [9] Shapiro, G. W., Nuttle, H.L.W. Hoist Scheduling for a PCB Electroplating Facility. *IIE Transactions*, vol. 20, n° 2, pp. 157-167, 1988.
- [10] Ptuskin, A.S. No-wait Periodic Scheduling of Non-identical Parts in Flexible Manufacturing Lines with Fuzzy Processing Times. International Workshop on Intelligent Scheduling of Robots and Flexible Manufacturing Systems. Center for Technological Education Holon, Israel, 1995.
- [11] Varnier, C. and Jeunehomme, N. A cyclic approach for the multi-product hoist scheduling problem. In 7th International Workshop on Project Management and Scheduling - PMS, Osnabrueck, 2000.
- [12] Mangione, F. Ordonnement des ateliers de traitement de surface pour une production cyclique et mono-produit. Thèse de Doctorat de l'Institut National Polytechnique de Grenoble, 2003.
- [13] Mateo M., Companys, R. Hoist Scheduling in a chemical line to produce batches with identical sizes of different products. Sixième Conférence Francophone de Modélisation et SIMulation, MOSIM'06. Rabat, Maroc, 2006.
- [14] El Amraoui^(a), A., Manier, M.A., El Moudni, A., and Benrejeb, M. A Comprehensive Linear Program to Solve Cyclic Hoist Scheduling Problem. The 9th International Conference on Sciences and Techniques of Automatic control and computer engineering, STA'2008, STA-SSI-487, Sousse, 2008.
- [15] El Amraoui^(b), A., Manier, M.A., El Moudni, A., and Benrejeb, M. A Mixed Linear Program for a Multi-Part Cyclic Hoist Scheduling Problem. *The International Journal of Sciences and Techniques of Automatic control & computer engineering IJ-STA*, Special Issue, CEM, pp. 612-623, 2008.
- [16] Song, W. Zabinsky, Z. B., Storch, R. L. An algorithm for scheduling a chemical processing tank line, Symposium on Emerging Technologies and Factory Automation, INRIA, IEEE, vol. 1, pp. 475-482, 1995.
- [17] Yih, Y. An algorithm for hoist scheduling problems. *International Journal of Production Research*, pp. 501-16, 1994.
- [18] Mateo, M. Procedimientos de secuenciación y programación en un sistema productivo de estaciones en serie con transportadores asincronos de material. Thèse de Doctorat de l'Université Polytechnique de Catalunya, 2003.