



HAL
open science

COLORATION DE GRAPHERS MIXTES ET SON APPLICATION A L'ORDONNANCEMENT JOB SHOP

Ahmed Kouider, Hacene Ait Haddadene, Samia Ourari, Ammar Oulamara

► **To cite this version:**

Ahmed Kouider, Hacene Ait Haddadene, Samia Ourari, Ammar Oulamara. COLORATION DE GRAPHERS MIXTES ET SON APPLICATION A L'ORDONNANCEMENT JOB SHOP . MOSIM 2014, 10ème Conférence Francophone de Modélisation, Optimisation et Simulation, Nov 2014, Nancy, France. hal-01166673

HAL Id: hal-01166673

<https://hal.science/hal-01166673>

Submitted on 23 Jun 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

COLORATION DE GRAPHES MIXTES ET SON APPLICATION A L'ORDONNANCEMENT JOB SHOP

A.KOUIDER^{1,2}, H.AITHADDADENE², S.OURARI¹

A.OULAMARA

¹Centre de Développement des Technologies Avancées
Division Productique et Robotique BP 17, Baba Hassen,
16303, Alger

Université de Lorraine, Laboratoire LCOMS
Ile de Saulcy, 57000, Metz, France
ammar.oulamara@univ-lorraine.fr

²Université des Sciences et de la Technologie Houari
Boumediene, Faculté des Mathématiques, BP 32, El
Alia, Bab-Ezzouar, 16111,
Alger { akouider, sourari } @cdta.dz,
aithaddadenehacene@yahoo.fr

RESUME : Dans ce papier, nous nous intéressons au problème d'ordonnancement d'atelier job-shop en considérant que toutes les opérations ont des durées unitaires, et à sa réduction à un problème de coloration de graphes. Nous modélisons le problème par la recherche de la coloration optimale d'une classe spéciale de graphes mixtes, où le graphe partiel engendré par les arcs représente une union de chemins et le graphe partiel engendré par les arêtes représente une union de cliques. Le problème étant NP-difficile, son étude nous a menés à la proposition de deux méthodes de résolution (exacte et approchée) et d'une borne inférieure pour le nombre chromatique mixte. La méthode exacte est basée sur la formulation mathématique sous forme d'un programme linéaire en variables mixtes. L'approche heuristique est basée sur l'algorithme du recuit simulé adapté pour la coloration du graphe mixte. Les résultats des expérimentations réalisées, en utilisant des Benchmarks adaptés de la littérature et des instances générées, montrent l'efficacité du programme linéaire, justifient également l'efficacité de l'algorithme du recuit simulé et attestent de la qualité de la borne inférieure.

MOTS-CLES : Coloration de graphes mixtes, Ordonnancement job-shop, Nombre chromatique mixte, Makespan.

1 INTRODUCTION

Le travail présenté dans ce papier s'intéresse à deux domaines de recherche, à savoir, la coloration de graphes mixtes et l'ordonnancement de production. Dans ces deux problèmes, étant donné un ensemble d'objets, il s'agit d'attribuer un nombre entier à chaque objet de manière à respecter certaines contraintes, tout en minimisant une fonction objectif.

Le problème d'ordonnancement d'ateliers job shop est l'un des problèmes d'ordonnancement les plus étudiés dans la littérature. Il est extrêmement complexe, vu l'explosion combinatoire du nombre de solutions, et est classé parmi les problèmes NP-difficile au sens fort (Lenstra & Rinnooy Kan, 1979). Ce problème reste NP-difficile au sens fort même dans le cas où les opérations des travaux possèdent la même durée unitaire (Lenstra and Rinnooy Kan, 1979 ; Timkovsky V. G., 1985).

De même, la coloration de graphes est parmi les problèmes d'optimisation combinatoire les plus étudiés. De nombreux problèmes pratiques peuvent, en effet, se ramener à un problème de coloration de graphes, et notamment ceux issus des domaines technologiques, où nous pouvons trouver les problèmes d'allocation de fréquences dans les

télécommunications (Maan and Sangwan, 2010), l'allocation de registres (Chaitin et al. 1981), les emplois du temps (Rong et al., 2009), la gestion des flux de trafic aérien (Barnier and Brisset, 2004), les chaînes logistiques (Lim and Wang, 2005) et les problèmes d'ordonnancement (Sotskov et al., 2001; Al-Anzi et al., 2006).

Certains problèmes pratiques peuvent être modélisés par la coloration de graphes mixtes composés à la fois par des arcs et des arêtes. Mais à notre connaissance, peu de travaux de recherche ont étudié la coloration de ce type de graphes. Le problème de coloration de graphes mixtes a été introduit pour la première fois dans (Sotskov & Tanaev, 1976). Ce problème est NP-complet dans le cas général, et aussi dans certains cas particuliers comme le cas d'un graphe biparti planaire ou biparti de degré maximum 3 (Ries, 2007).

La coloration de graphes mixtes a été aussi étudiée pour le cas particulier d'un graphe biparti planaire de degré maximum 4 en considérant que chaque sommet incident à un arc est de degré maximal 2. Ce problème de coloration est aussi NP-Complet (Ries & de Werra, 2008).

D'autres travaux de littérature ont proposé des algorithmes exacts pour la coloration de certaines classes de graphes. Nous citons entre autres les travaux de Hansen et al. (1997) qui ont proposé un algorithme de complexité

$O(n^2)$ pour la coloration des arbres mixtes. Hanna et al. (2008) ont aussi proposé un algorithme polynomial pour le calcul du nombre chromatique des arbres mixtes, et leur algorithme est venu améliorer celui proposé par Hansen et al. (1997).

Parmi les travaux récents, nous citons ceux de Ries and de Werra (2008) qui ont étudié deux types différents de coloration de graphes mixtes. Le premier impose aux sommets adjacents d'avoir des couleurs différentes, et au sommet de l'extrémité initiale d'un arc d'avoir une couleur inférieure à celle du sommet de l'extrémité terminale. Dans le second type de coloration, les sommets liés par un arc sont contraints d'avoir la même couleur. Pour ces deux types de coloration, les auteurs ont proposé des bornes pour le nombre chromatique.

Pour des cas pratiques de problèmes d'ordonnement, Sotskov et al. (2001) ont utilisé la coloration de graphes mixtes pour résoudre le problème d'ordonnement $J|p_{ij} = 1|C_{max}$. Ils ont proposé trois méthodes de résolution basées sur la séparation et évaluation, et les ont testées sur des instances aléatoires de graphes mixtes d'ordre inférieur ou égal à 200 sommets.

Dans ce même contexte, Al-Anzi et al. (2006) ont étudié le problème d'ordonnement $J|p_{ij} = 1|\sum C_i$. Ce problème est aussi modélisé par la coloration d'un graphe mixte avec comme critère la minimisation de la somme des couleurs maximales. Les auteurs ont développé un algorithme par séparation et évaluation et ont proposé trois bornes inférieures pour la fonction objectif du problème (la somme coloration). Ils ont testé leurs méthodes sur des graphes mixtes générés aléatoirement. Dans les expérimentations, ils ont considéré deux types d'instances de graphes mixtes : le premier type regroupe des instances de graphes avec des chemins de même longueur, alors que le second type est composé d'instances de graphes avec des chemins de longueur différentes. Signalons que le premier type d'instances de graphes est scindé, en deux catégories : la première catégorie comporte des instances de graphes avec des chemins de longueur égale au nombre de cliques maximales, et la seconde catégorie regroupe des instances de graphes avec des chemins de longueur différente au nombre de cliques maximales. Pour ce premier type d'instances, les auteurs ont coloré des graphes mixtes d'ordre inférieur ou égal à 200 sommets.

Notons que la technique de génération de graphes et le modèle d'instances de graphes mixtes d'ordre 200 sommets utilisés par les auteurs ont été adoptés pour la validation d'une partie de nos travaux.

Ce papier est organisé comme suit. Dans la deuxième section, le problème de coloration de graphes mixtes est défini. Ensuite dans la troisième section, la problématique d'ordonnement est présentée. La quatrième section décrit la réduction du problème d'ordonnement à la coloration de graphes mixtes et propose une borne inférieure

pour le nombre chromatique mixte. Puis dans la cinquième section, une modélisation du problème est donnée, suivie dans la sixième section, de l'adaptation de l'algorithme du recuit simulé adoptée pour la coloration du graphe mixte. Dans la septième section, des expérimentations sont présentées dans le but de valider la modélisation proposée et de juger sur l'efficacité de l'algorithme du recuit simulé proposé.

2 COLORATION DE GRAPHES MIXTES

Considérons un graphe mixte $G = (V, A, E)$, où V est l'ensemble de sommets, A est l'ensemble d'arcs et E est l'ensemble d'arêtes. Soit K un ensemble de nombres naturels. La fonction $\varphi: V \rightarrow K$, est appelée coloration du graphe mixte, si elle associe un nombre $\varphi(v_i) \in K$ à chaque sommet $v_i \in V$ de manière que :

- Chaque arc $v_i v_j \in A$ vérifie la condition $\varphi(v_i) < \varphi(v_j)$.
- Chaque arête $u_i u_j \in E$ vérifie la condition $\varphi(u_i) \neq \varphi(u_j)$.

Le nombre chromatique mixte de G , noté $\chi(G)$, est le minimum de couleurs nécessaire pour colorer les sommets de G .

3 PROBLEME D'ORDONNANCEMENT

Dans ce papier, le problème d'ordonnement job shop que nous étudions, noté $J|p_{ij}=1|C_{max}$, est NP-difficile au sens fort (Lenstra and Rinnooy Kan, 1979 ; Timkovsky V. G., 1985). Il est défini par un ensemble de n travaux (jobs) J_1, \dots, J_n à exécuter sur m machines M_1, \dots, M_m . Chaque travail $J_i, i=1, \dots, n$, possède sa propre gamme de production. Une gamme de production est composée d'une suite de n_i opérations devant être exécutées selon un ordre bien défini sur les différentes machines de l'atelier, cet ordre traduit les contraintes de précédence entre les différentes opérations du même travail. Nous supposons que les machines sont disjonctives, i.e., que chaque machine ne peut exécuter plus d'une opération à un instant donné, et enfin qu'une opération O_{ij} ($i=1, \dots, n; j=1, \dots, n_i$) ne peut être exécutée que par une et une seule machine, et que son exécution nécessite une durée opératoire unitaire ($p_{ij}=1, i=1, \dots, n; j=1, \dots, n_i$).

Une solution réalisable (S) à ce problème d'ordonnement consiste à définir la date de début d'exécution t_{ij} pour chaque opération O_{ij} en respectant les contraintes de précédence entre les opérations d'un même travail et les contraintes de disjonction des ressources. L'objectif considéré est de déterminer une solution réalisable S qui optimise le *makespan*. (i.e. la date de fin d'exécution de la dernière opération ordonnancée).

4 REDUCTION DU PROBLEME $J/P_{ij}=I/C_{MAX}$ A LA COLORATION D'UN GRAPHE MIXTE

Basant sur la définition de la coloration de graphes mixtes, le problème d'ordonnancement $J/P_{ij}=I/C_{max}$ peut être réduit à la coloration d'un graphe mixte $G = (V, A, E)$, où l'ensemble de sommets de V représente les opérations des travaux à réaliser. L'ensemble des arcs A représente les contraintes de précédence reliant les opérations appartenant à la même gamme de produit, i.e., un arc $v_i v_j$ existe si l'opération j est le successeur immédiat de l'opération i . L'ensemble des arêtes E représente les contraintes de disjonction entre les opérations à exécuter par la même ressource, i.e. une arête $u_i u_j$ existe lorsque deux opérations i et j sont à exécuter par la même machine et ne sont pas exécutées dans le même intervalle de temps (rappe-lons que la machine où l'opération doit être exécutée est connue à priori).

De cette modélisation, le graphe mixte $G = (V, A, E)$ possède les propriétés suivantes :

- La partition $G_a = (V, \emptyset, E) = \cup_{i=1}^m (V_i, \emptyset, E_i)$ tel que, $V_j \cap V_k = \emptyset \forall j \neq k$, où chaque (V_i, \emptyset, E_i) $i=1, \dots, m$ est une clique maximale.
- La partition $G_b = (V, A, \emptyset) = \cup_{i=1}^n (V^i, A^i, \emptyset)$ tel que $V^j \cap V^k = \emptyset \forall j \neq k$, où chaque (V^i, A^i, \emptyset) , $i=1, \dots, n$ est un chemin.
- Chaque sommet du graphe mixte appartient à la fois à un et un seul chemin, et appartient aussi à une et une seule clique maximale.

Notons que les valeurs m et n données précédemment représentent respectivement les cardinalités des ensembles des machines $M=\{M_1, \dots, M_m\}$ et des travaux $J=\{J_1, \dots, J_n\}$. Signalons qu'à partir de ce modèle, le sommet v correspond à l'opération v et lorsqu'un sommet $v \in V_i$ d'une clique maximale (V_i, \emptyset, E_i) alors l'opération correspondant au sommet v est affectée à la machine M_i . D'un autre côté, quand un sommet $v \in V^l$ d'un chemin (V^l, A^l, \emptyset) alors l'opération représentée par le sommet v appartient au travail J_l . En d'autres termes, sachant que chaque sommet du graphe mixte représente une opération, les sommets d'un même chemin correspondent aux opérations d'un même travail selon la gamme opératoire, alors que les sommets d'une même clique maximale correspondent aux différentes opérations à exécuter par une même machine. Par conséquent : une coloration ϕ correspond à un ordonnancement ϕ (i.e. la date de fin d'exécution de chaque opération correspond à la coloration du sommet associé).

Le nombre chromatique mixte correspond à la date de fin d'exécution de la dernière opération de l'ordonnancement. Ainsi le makespan correspond au nombre chromatique mixte ($C_{max} = \chi(G)$).

Par la réduction présentée ci-dessus, nous pouvons utiliser à la fois la terminologie de la théorie de graphes et la terminologie de l'ordonnancement pour le problème étudié.

4.1 Borne inférieure

Dans ce qui suit, nous proposons une borne inférieure pour le nombre chromatique mixte représentant la coloration optimale du graphe mixte modélisant le problème d'ordonnancement étudié.

Rappelons:

- V^i est l'ensemble des sommets du chemin (V^i, A^i, \emptyset) issu du graphe mixte, $i=1, \dots, n$.
- V_j est l'ensemble des sommets de la clique maximale (V_j, \emptyset, E_j) issue du graphe mixte, $j=1, \dots, m$.

Posons $\omega_j = |V_j|, j=1, \dots, m$, et soient :

- l^k : le sous chemin succédant le sommet k , tel que $k \in V^i$ et $k \in V_j, i=1, \dots, n$ et $j=1, \dots, m$.
- h^k : le sous chemin précédant le sommet k , tel que $k \in V^i$ et $k \in V_j, i=1, \dots, n$ et $j=1, \dots, m$.

Proposition :

$$LB = \max_{1 \leq j \leq m} (\min_{k \in V_j} |h^k| + \omega_j + \min_{k \in V_j} |l^k|)$$

est une borne inférieure pour le nombre chromatique mixte représentant la coloration optimale du graphe mixte modélisant le problème d'ordonnancement $J/P_{ij}=I/C_{max}$.

Preuve :

Considérons le graphe mixte $G = (V, A, E)$, et soient $G_j = (V, A, E_j), j=1, \dots, m$, les graphes partiels de G engendrés respectivement par l'ensemble des arcs A et le sous ensemble des arêtes E_j . (où E_j est l'ensemble des arêtes de la clique maximale $G_j = (V_j, \emptyset, E_j), j=1, \dots, m$).

Notons par χ_j le nombre chromatique mixte du graphe $G_j = (V, A, E_j), j=1, \dots, m$.

Il est évident que la cardinalité de la coloration optimale du graphe mixte G , notée χ , est telle que $\chi \geq \max_{1 \leq j \leq m} (\chi_j)$.

Concentrons-nous à présent sur le sous graphe G_j , et rappelons que chaque sommet k du graphe G_j est à la fois l'extrémité initiale d'un sous chemin l^k et l'extrémité terminale d'un sous chemin h^k .

Ainsi, le nombre chromatique mixte de ce graphe est au moins égal à la somme constituée de : (i) le minimum des longueurs des sous-chemins précédant les sommets de sa clique maximale $G_j' = (V_j, \emptyset, E_j)$, (ii) la cardinalité de sa clique maximale $G_j' = (V_j, \emptyset, E_j)$, ainsi que (iii) le minimum des longueurs des sous-chemins succédant les sommets de sa clique maximale $G_j' = (V_j, \emptyset, E_j)$.

De manière formelle, cela s'écrit par l'expression suivante :

$$\chi_j \geq \min_{k \in V_j} |h^k| + \omega_j + \min_{k \in V_j} |l^k|, \forall j \in \{1, \dots, m\}$$

Par conséquent :

$$\chi \geq \max_{1 \leq j \leq m} (\min_{k \in V_j} |h^k| + \omega_j + \min_{k \in V_j} |l^k|)$$

5 FORMULATION MATHÉMATIQUE

Cette section est consacrée à la formulation mathématique des différentes contraintes ainsi que la fonction objectif à minimiser du programme linéaire proposé pour la coloration du graphe mixte.

5.1 Les paramètres

- n : Le nombre de chemins.
- m : Le nombre de cliques maximales.
- n_i : Le nombre de sommets du chemin (V^i, A^i, \emptyset) .
- u_{ik} : Le kème sommet du chemin (V^i, A^i, \emptyset) .

5.2 Les indices

- i, h : Indices des chemins, $i=1, \dots, n$ et $h=1, \dots, n$.
- j : Indice d'une clique maximale, $j=1, \dots, m$.
- k, l : Indices relatifs à l'ordre du sommet dans le chemin (V^i, A^i, \emptyset) , $k=1, \dots, n_i$ $l=1, \dots, n_i$.
- S_j : Ensemble des indices des sommets appartenant à la clique maximale (V_j, \emptyset, E_j) .
- M : Un entier assez grand.

5.3 Les variables de décision

- $\varphi(u_{ik})$: Un entier positif indiquant la couleur du sommet u_{ik} .
- $x_{ik,hl} = \begin{cases} 1 & \text{si } \varphi(u_{ik}) < \varphi(u_{hl}) \\ 0 & \text{sinon} \end{cases}$
- χ : Le nombre chromatique mixte.

5.4 Les contraintes

5.4.1 Contraintes liées à la coloration des sommets de chaque chemin

Ces contraintes assurant la précédence entre les sommets d'un chemin et indiquent que la couleur d'un sommet $u_{i,k+1}$ ne peut être inférieure ou égale à la couleur de son prédécesseur $u_{i,k}$ dans le chemin (V^i, A^i, \emptyset) :

$$\varphi(u_{ik}) + 1 \leq \varphi(u_{i,k+1}), \forall i \in \{1, \dots, n\} \text{ et } \forall k \in \{1, \dots, n_i-1\} \quad (1)$$

5.4.2 Contraintes relatives au nombre chromatique mixte à optimiser

Ces contraintes permettent d'assurer que les couleurs des sommets soient toutes inférieures au nombre chromatique mixte (qu'il faut minimiser). Sachant que pour un chemin donné, la couleur maximale est attribuée à son dernier sommet, ces contraintes s'écrivent :

$$\varphi(u_{in_i}) \leq \chi \quad \forall i \in \{1, 2, \dots, n\} \quad (2)$$

5.4.3 Contraintes liées à la coloration des sommets de chaque clique maximale

Ces contraintes veillent à ce que deux sommets (u_{ik}, u_{hl}) de la même clique maximale (V_j, \emptyset, E_j) aient des couleurs différentes.

Les contraintes s'écrivent, dans le cas où la couleur du sommet u_{hl} est inférieure à la couleur du sommet u_{ik} .

$$\varphi(u_{hl}) + 1 \leq \varphi(u_{ik}), \forall (h, l), (i, k) \in S_j \quad (3)$$

Par contre, dans le cas où la couleur du sommet u_{ik} est inférieure à la couleur du sommet u_{hl} , les contraintes s'écrivent :

$$\varphi(u_{ik}) + 1 \leq \varphi(u_{hl}), \forall (i, k), (h, l) \in S_j \quad (4)$$

Ces deux contraintes disjonctives sont linéarisées par les inéquations (5) et (6) données plus bas, où M est un entier positif suffisamment grand :

$$\varphi(u_{ik}) + 1 \leq \varphi(u_{hl}) + M(1 - x_{ik,hl}), \quad \forall (i, k), (h, l) \in S_j \quad \forall j \in \{1, \dots, m\} \text{ et } (i, k) \neq (h, l) \quad (5)$$

$$x_{ik,hl} + x_{hl,ik} = 1 \quad \forall (i, k), (h, l) \in S_j, (i, k) \neq (h, l) \text{ et } \forall j \in \{1, \dots, m\} \quad (6)$$

5.5 La fonction objectif

L'objectif est de minimiser le nombre de couleurs attribuées aux sommets (nombre chromatique mixte), soit : $\min \chi$.

6 ADAPTATION DE L'ALGORITHME DU RECUIT SIMULÉ POUR LA COLORATION DU GRAPHE MIXTE

Plusieurs travaux de recherche ayant adopté l'algorithme du recuit simulé ont montré l'efficacité de son utilisation à de nombreux problèmes d'optimisation combinatoire. Notons toutefois, que dans le cas des problèmes de coloration de graphes mixtes, son efficacité dépend essentiellement de l'adaptation opérée au niveau de ses paramètres, particulièrement, au codage d'une solution qui doit satisfaire les deux types de contraintes (contraintes de précédence entre les sommets appartenant aux mêmes chemins et contraintes liées aux cliques). Aussi, les autres paramètres qui influent sur l'efficacité de cet algorithme sont:

- La fonction de génération d'une solution voisine
- La température initiale.
- La fonction de décroissance de la température (comment fait-on décroître la température).
- Le nombre de paliers de température que l'on souhaite effectuer.
- La longueur de chaque palier (combien fait-on d'itérations pour une température donnée).
- La manière dont la probabilité $\text{Prob}(\Delta E, T)$ dépend de la température (T) et de (ΔF) .
- La fonction d'évaluation d'une solution.

6.1 L'algorithme du recuit simulé

L'algorithme 1 décrit le fonctionnement de l'algorithme du recuit simulé proposé.

Algorithme 1. Recuit simulé pour la coloration du graphe mixte.

Entrée V : vecteur de sommets (avec la précision de la clique et du chemin pour chaque sommet).
 Nb, T_0, L, α, K_B : les paramètres de l'algorithme.

Sortie S^* : la meilleure solution du problème

Début
 Fixer les paramètres de l'algorithme
 Construire une solution initiale S_0
 $S \leftarrow S_0$
 $S^* \leftarrow S_0$

Tant que le nombre de palier de température n'est pas atteint **faire**
 $T \leftarrow \alpha T$
 $Val \leftarrow F(S)$

Tant que la longueur du palier de température n'est pas atteinte **faire**
 Générer une solution voisine S_v
 $\Delta F \leftarrow F(S_v) - F(S)$
Si ($\Delta F \leq 0$)
 $S \leftarrow S_v$
Si ($F(S) < F(S^*)$)
 $S^* \leftarrow S$
fin si
fin si
Si ($\Delta F > 0$)
 $Prob(\Delta F, T) \leftarrow \exp\left(\frac{-\Delta F}{K_B T}\right)$
 Générer q uniformément dans l'intervalle $[0, 1[$
Si ($q < Prob(\Delta F, T)$)
 $S \leftarrow S_v$
fin si
fin si
fin tant que
Si ($Val < F(S)$)
 $T \leftarrow T/\alpha$
fin de si
fin tant que

Fin

Notons qu'au niveau de chaque itération de l'algorithme, la solution courante est évaluée. Si elle est acceptée par la règle de Metropolis (Metropolis et al., 1953), elle devient alors une solution courante, alors que si elle améliore la meilleure solution, elle devient elle-même la meilleure solution. Ensuite, la température est diminuée pour passer à un autre palier de température (cf. Mécanisme de refroidissement). Le pseudo code est détaillé dans l'algorithme 1.

6.2 Codage d'une solution

Considérons le graphe mixte $G = (V, A, E)$ contenant n chemins, qu'on note dans cette section Ch_i ($i=1, \dots, n$), et soit U un vecteur d'entiers de taille égale au nombre de

sommets. Dans le codage adopté, chaque numéro de chemin ($i=1, \dots, n$) apparaît dans le vecteur U exactement n_i fois, où $n_i = |Ch_i|$ ($|Ch_i|$ est le nombre de sommets du chemin Ch_i). La $k^{\text{ième}}$ occurrence du numéro d'un chemin, dans le vecteur U , désigne son $k^{\text{ième}}$ sommet.

Une solution (coloration du graphe) est construite, à partir du codage, par l'algorithme 2.

Algorithme 2. Coloration du graphe à partir d'un codage.

Entrée

- 1- U : vecteur d'entiers (constitué d'une permutation de $|V|$ éléments, à partir de n éléments où chaque élément est répété n_i fois et $|V| = \sum_{i=1}^n n_i$).
- 2- V : vecteur de tous les sommets du graphe (avec la précision de la clique maximale et du chemin pour chaque sommet).

Sortie
 C : un vecteur de coloration des sommets du graphe.

Début
 Pour i allant de 1 à $|U|$ faire

début

- 1- Dans le chemin contenant le sommet $U[i]$, déterminer la couleur ($Col1$) du sommet précédent $U[i]$ (si $U[i]$ est sans prédécesseur, alors $Col1=0$);
- 2- Dans la clique maximale contenant le sommet $U[i]$, déterminer la couleur minimale ($Col2$) des sommets appartenant à la même clique que $U[i]$ et qui sont déjà colorés (si aucun sommet de la clique maximale n'est coloré, alors $Col2=0$);
- 3- Déterminer la position j correspondante au sommet $U[i]$ dans le vecteur V ;
- 4- $C[j] = \max(Col1, Col2) + 1$;

fin
Fin

6.3 Génération de la solution initiale.

Dans l'algorithme du recuit simulé proposé, la solution initiale est construite progressivement comme suit :

Nous générons aléatoirement n_1 positions parmi les positions non assignées du vecteur U pour la valeur 1 (représentant le chemin 1), puis n_2 positions parmi les positions non assignées pour la valeur 2 (représentant le chemin 2) et ainsi de suite jusqu'à ce que les n chemins soient assignés dans le vecteur U .

6.4 Fonction de voisinage

Une solution voisine est obtenue par une permutation de deux éléments du vecteur U codant une solution courante. Dans le codage adopté, la permutation devrait concerner uniquement les sommets appartenant aux chemins différents, et par conséquent les éléments à permuter devraient avoir des valeurs différentes.

6.5 Détermination des paramètres T_0 , L , α et K_B

Les paramètres Nb, T_0, L, α et K_B (nombre de paliers de température, température initiale, longueur du palier de température, le taux de décroissance de la température et la constante de Boltzmann) sont déterminés de manière expérimentale en tenant compte de la taille du problème.

6.6 Détermination de la probabilité P

La probabilité P d'accepter une solution courante X^* , quand cette solution est moins bonne que la meilleure solution, décroît avec le temps de l'algorithme. Cette probabilité P est une fonction $Prob(\Delta F, T)$ dépendant de la température T et de la dégradation de la fonction objectif $\Delta F = F(X^*) - F(X)$, X étant la solution courante. Nous rappelons que la loi de probabilité P est dictée par la distribution de Boltzmann $P = Prob(\Delta F, T) = \exp\left(\frac{F_i - F_j}{K_B T}\right)$.

6.7 Mécanisme de refroidissement

Dans le mécanisme de refroidissement, nous avons adopté la procédure de réduction par palier de la température, où la température initiale T_0 est maintenue constante pendant L itérations. Après ces L itérations, on passe à une température $T_1 = \alpha T_0$ ($0 < \alpha < 1$) que l'on réduit à $T_2 = \alpha^2 T_0$ pour L autres itérations, et ainsi de suite, jusqu'à l'itération KL , où la température sera $T_k = \alpha^k T_0$.

7 EXPERIMENTATIONS ET RESULTATS

Dans cette section, nous décrivons les expérimentations réalisées pour l'évaluation des performances du modèle mathématiques et de l'algorithme du recuit simulé proposés.

Nous précisons que le modèle mathématique est résolu par le solveur CPLEX et que le développement informatique pour l'algorithme du recuit simulé est effectué en C++ sous Windows 8.1. Les tests expérimentaux sont réalisés sur un PC équipé d'un processeur Intel (R) Core (TM) i3-2100 à 3.10 GHz et ayant 6 Go de mémoire vive (RAM).

Dans les expérimentations, nous avons utilisé deux types d'instances : des instances générées aléatoirement et des Benchmarks d'OR-Library¹ adaptés au problème $J|p_{ij} = 1|C_{max}$ en substituant la valeur de la durée opératoire de chaque opération de l'instance par une durée unitaire ($p_{ij} = 1$). Notons qu'afin de différencier ces instances de la littérature, à celles que nous avons testées après modification, nous avons rajouté la lettre (m) à l'identifiant de chaque instance.

7.1 Validation du modèle mathématique

Le modèle mathématique a été testé sur deux types de graphes mixtes: dans le premier type, tous les chemins des graphes mixtes possèdent la même longueur égale au nombre de cliques maximales, alors que dans le second type de graphe, les chemins ont la même longueur mais qui est différente au nombre de cliques maximales.

Le premier type comporte alors des instances allant jusqu'à 100 sommets, alors que le second comporte des instances allant jusqu'à 250 sommets. Pour chaque graphe mixte, nous donnons, le nombre de sommets ($|V|$), d'arcs ($|A|$), d'arêtes ($|E|$), de cliques maximales ($|Cl|$), de chemins ($|Ch|$) et la borne inférieure du nombre chromatique mixte (LB).

Les différents résultats des expérimentations issus de la résolution des instances par le modèle mathématique sont donnés dans le tableau 1, où on peut lire (pour chaque type d'instances) : la solution obtenue ' χ ', le nœud où la solution optimale est obtenue ' S_N ', le nombre de nœuds parcourus ' Nb_N ' ainsi que la durée de résolution en seconde ' CPU '. La solution quand elle est optimale est marquée par '*'.

7.1.1 Graphes mixtes avec chemins de longueur égale au nombre de cliques maximales

Dans ce cas, seules les instances de moins de 100 sommets (la10m à la20m, orb01m à orb10m, ft06m, ft20m et abz05m) ont été testées. Les résultats sont donnés dans les 7 premières colonnes du tableau 1.

7.1.2 Graphes mixtes avec chemins de longueur différente au nombre de cliques maximales

Dans cette partie, on s'intéresse à l'expérimentation d'un type particulier d'instances de graphes mixtes possédant des chemins de longueur différente aux nombre de cliques maximales (Rand1 à Rand 35). Ce choix est justifié par le fait qu'il a été utilisé dans la littérature pour la coloration de graphes mixtes à 200 sommets (Al-Anzi et al., 2006). Afin de générer ces instances, nous avons utilisé la méthode de génération utilisée dans les travaux d'Al-Anzi et al. (2006).

Deux valeurs de $|V|$ sont considérés : {200 ; 250}, et trois valeurs de n sont considérés : {10, 20, 25}. Pour chaque instance de graphes mixtes générée aléatoirement, la longueur des chemins étant égale à $|V|/n$, les valeurs de $|V|$ et n sont alors choisies de manière à ce que $|V|/n$ soit entière. Partant de ces données, et pour chaque couple de valeurs ($|V|, n$), les cliques maximales générées ont un nombre de valeur appartenant à {15, 18, 33, 35, 40}. L'ensemble des combinaisons est rapporté dans le tableau 1. Pour chaque combinaison possible (i.e., pour chaque instance de graphes mixtes générée), cinq (05) instances de graphes sont générées aléatoirement. Les résultats sont donnés dans les 7 dernières colonnes du tableau 1.

¹<http://people.brunel.ac.uk/~mastjjb/jeb/orlib/files/jobshop1.txt>

chemins de longueur égale au nombre de clique							chemins de longueur différente au nombre de clique						
Taille	LB	χ	S_N	Nb_N	CPU(s)		Taille	LB	χ	S_N	Nb_N	CPU(s)	
(V _i ,A _i , E _i ,C _i),(Ch)	la01m	10	10*	4813	27436	19.3	(V _i ,A _i , E _i ,C _i),(Ch)	Rand1	23	25*	55027	629177	1051.7
	la02m	11	11*	2516558	2516969	906.9		Rand2	22	26	30871	3157005	5400
	la03m	12	12*	4991	4993	3.0		Rand3	21	25*	28036	317267	608.8
	la04m	11	11*	4958	5063	2.3		Rand4	23	25*	88200	210623	384.1
	la05m	11	11*	4621	140489	65.7		Rand5	22	26*	105117	2848319	4948.7
(V _i ,A _i , E _i ,C _i),(Ch)	la06m	15	15*	5061	5089	5.1	(V _i ,A _i , E _i ,C _i),(Ch)	Rand6	22	25*	2255531	2679960	4219.9
	la07m	16	16*	772620	772941	526.9		Rand7	21	25*	6219	1476574	2172.9
	la08m	15	15*	1497212	1497213	671.9		Rand8	23	25*	29624	139645	191.9
	la09m	15	15*	28204	28204	29.1		Rand9	22	24*	91821	94282	127.0
	la10m	16	16*	28368	28368	28.5		Rand10	21	25	1924	593578	5400
(V _i ,A _i , E _i ,C _i),(Ch)	la11m	21	22	104293	8262350	5400	(V _i ,A _i , E _i ,C _i),(Ch)	Rand11	22	25*	4846	113699	134.5
	la12m	20	20*	33387	33388	49.5		Rand12	22	25*	33484	109638	146.6
	la13m	20	21	7838	7439645	5400		Rand13	21	26*	145740	2766465	3965.3
	la14m	20	20*	3	3	0.5		Rand14	22	25*	12688	619216	777.8
	la15m	20	23	21125	3252889	5400		Rand15	21	25*	10651	1059664	1283.4
(V _i ,A _i , E _i ,C _i),(Ch)	ft20m	22	28	115817	40809595	5400	(V _i ,A _i , E _i ,C _i),(Ch)	Rand16	11	13*	5136	17700	15.2
	la16m	13	15*	813301	817600	712.5		Rand17	12	13*	3367	3806	5.5
	la17m	15	15*	40346	40366	24.5		Rand18	12	13*	633	8744	8.6
	la18m	13	15*	29605	107304	96.5		Rand19	12	13*	4911	5171	6.1
	la19m	14	15*	107658	117546	104.3		Rand20	11	13*	1824	16745	15.8
	la20m	12	15*	156712	197119	172.9		Rand21	12	13*	5190	5610	6.1
	orb1m	16	20	3016894	8043914	5400		Rand22	12	13*	3235	5873	7.8
	orb2m	14	17	28100	9187712	5400		Rand23	11	14*	241	8404	5.5
	orb3m	17	20	704089	9340073	5400		Rand24	11	12*	5825	5875	3.9
	orb4m	17	18*	147484	3436881	585.1		Rand25	12	13*	9043	9517	11.5
(V _i ,A _i , E _i ,C _i),(Ch)	orb5m	16	17	302286	5105263	5400	(V _i ,A _i , E _i ,C _i),(Ch)	Rand26	12	13*	2345	2440	4.0
	orb6m	16	20	3016894	8878386	5400		Rand27	11	12*	4901	4901	4.3
	orb7m	14	17	28100	9187712	5400		Rand28	11	12*	4022	4064	3.5
	orb8m	17	21	352995	4389382	5400		Rand29	11	12*	3	58	2.3
	orb9m	17	18	19053898	31028438	5400		Rand30	11	12*	2128	2270	2.9
	orb10_m	16	17	302286	5374000	5400		Rand31	12	13*	3715383	3716033	1607.1
	abz5m	12	15*	222418	334004	211.4		Rand32	12	14	8607	9299616	5400
	ft10m	16	19	11609929	11711459	5400		Rand33	12	13*	830	10043	14.8
								Rand34	11	13*	293341	295062	135.0
								Rand35	13	13*	6960	6973	13.9

Tableau 1 : Coloration de graphes mixtes par la résolution du modèle mathématique

7.2 Analyse des résultats du modèle

D'après les résultats des expérimentations pour le premier type d'instances de graphes mixtes d'ordre $|V| \leq 100$ sommets et de longueur de chemin égale au nombre de cliques maximales, nous remarquons que le modèle a résolu de manière optimale 19 instances parmi les 33 instances testées. Pour le reste des instances (non résolues de manière optimale), le temps d'exécution du solveur est limité à 5400 secondes.

Signalons que, le pourcentage de déviation d'une solution approchée par rapport à la borne inférieure est obtenue par :

$$PDB = \frac{\text{la solution approchée} - \text{la borne inférieure}}{\text{la borne inférieure}}$$

Notons que le maximum du PDB (resp. la moyenne du PDB) relatif aux instances résolues par le modèle est 0.27 (resp. 0.158).

Pour le second type de graphes mixtes d'ordre $|V| \leq 250$ sommets et de longueur de chaque chemin différente au nombre de cliques maximales, nous constatons que le modèle a résolu de manière optimale 32 instances parmi les 35 instances générées.

Une analyse de la performance de la borne inférieure par rapport à la solution optimale pour le premier type d'instances est ainsi déduite à la base de la moyenne de l'écart de la solution optimale par rapport à la borne inférieure qui est de 0.782 et permet de conclure quant à l'efficacité de la borne inférieure proposée.

(V , A , E , Cl , Ch)	LB	χ_m	CPU(s)	(V , A , E , Cl , Ch)	LB	χ_m	CPU(s)			
(50,40,225,5,10)	la01m	10	10*	7	(100,90,450,10,10)	ft10m	16	18	16	
	la02m	11	11*	7		abz05m	12	15*	16	
	la03m	12	12*	7		abz06m	23	27	75	
	la04m	11	11*	7		abz07m	23	27	75	
	la05m	10	11*	7		abz08m	24	27	75	
(75,60,525,5,15)	la06m	15	15*	10	(300,280,2850,15,20)	abz09m	25	27.4	75	
	la07m	16	16*	10		(100,80,950,5,20)	ft20m	22	24	13
	la08m	15	15*	10		orb01m	16	18.2	16	
	la09m	15	15*	10		orb02m	14	17	16	
	la10m	16	16*	10		orb03m	17	19	16	
(100,80,950,5,20)	la11m	21	21*	13	(100,90,450,10,10)	orb04m	17	18*	16	
	la12m	20	20*	13		orb05m	16	17	16	
	la13m	20	20*	13		orb06m	16	18	16	
	la14m	20	20*	13		orb07m	14	17*	16	
	la15m	20	20*	13		orb08m	17	19	16	
(100,90,450,10,10)	la16m	13	15*	16	(400,380,3800,20,20)	orb09m	17	18	16	
	la17m	15	15*	16		orb10m	16	17	16	
	la18m	13	15*	16		yn1m	23	30	117	
	la19m	14	15*	16		yn2m	23	31	117	
	la20m	12	15*	16		yn3m	24	31.8	117	
(150,135,1050,10,15)	la21m	17	20	25	(200,180,1900,10,20)	yn4m	25	34	117	
	la22m	17	18.6	25		swv01m	26	30	36	
	la23m	16	18	25		swv02m	25	30	36	
	la24m	17	17*	25		swv03m	26	30	36	
	la25m	17	19	25		swv04m	25	30	36	
(200,180,1900,10,20)	la26m	22	22.6	36	(300,280,2850,15,20)	swv05m	25	30	36	
	la27m	22	23	36		swv06m	28	35	71	
	la28m	22	22*	36		swv07m	28	35	71	
	la29m	21	24	36		swv08m	28	35.2	71	
	la30m	21	24	37		swv09m	29	35.2	71	
(300,270,4350,10,30)	la31m	32	32*	68	(500,450,12250,10,50)	swv10m	29	35	71	
	la32m	31	32	69		swv11m	55	69.8	168	
	la33m	30	32	69		swv12m	55	70	168	
	la34m	31	32	69		swv13m	55	69.6	168	
	la35m	30	32	68		swv14m	55	69.2	168	
(225,210,1575,15,15)	la36m	18	23	46	swv15m	55	68.8	168		
	la37m	19	24	46	swv16m	51	51*	168		
	la38m	19	23	46	swv17m	50	50.6	168		
	la39m	19	23	46	swv18m	50	51	168		
	la40m	18	24	46	swv19m	50	50.8	168		
(36,30,90,6,6)	ft06m	8	9*	6	swv20m	50	50*	168		

Tableau 2 : Coloration de graphes mixtes par le recuit simulé : chemins de longueur égale au nombre de cliques

7.3 Performance de l'approche métaheuristique

Dans cette section, nous présentons les résultats des expérimentations de l'algorithme du recuit simulé proposé. Notons que les paramètres de l'algorithme sont fixés comme suit :

Nb=500, $T_0=1$, L=1000, $\alpha=0.97$ et $K_B=0.01$.

Le tableau 2 rapporte les résultats des expérimentations pour toutes les instances modifiées de la librairie 'OR-Library' et représentant des instances de graphes mixtes de chemins de longueur égale au nombre de cliques maximales (Rand1 à Rand 35). Le tableau 3 donne les résultats des expérimentations sur les instances de graphes mixtes possédant des chemins de longueur différente aux nombre de cliques maximales (Rand1 à Rand 35).

Dans les tableaux 2 et 3, on peut lire la valeur $\bar{\chi}$ représentant la moyenne des nombres chromatiques issus des cinq lancements d'un même Benchmark. Pour un même Benchmark, la valeur CPU(s) indique le temps maximum de résolution de ses cinq lancements. Quand la valeur $\bar{\chi}$ est égale à la solution optimale, elle est marquée par '*'.
 Notons que pour tous les Benchmarks testés, l'écart maximum entre les solutions trouvées pour les cinq lancements d'un même Benchmark est d'une unité. Il en découle que pour un benchmark donné, la meilleure (resp. la mauvaise) solution représente la valeur de la partie entière minimale (resp. maximale) de la moyenne calculée.

De manière globale, nous constatons que relativement aux valeurs optimales obtenues par le modèle mathématique, ainsi que les valeurs des bornes inférieures calculées, les différentes expérimentations réalisées à la base de l'approche métaheuristique, permettent de retourner la solu-

tion optimale dans le cas de plusieurs instances, et particulièrement pour la plupart des instances relatives aux graphes mixtes avec chemins de longueur différente au nombre de cliques maximales.

D'autre part, en analysant les expérimentations de l'algorithme du recuit simulé réalisées, nous constatons pour le premier type de graphe mixte avec des chemins de longueur égale au nombre de cliques maximales, que 29 instances sur les 82 instances testées, sont résolues de manière optimale pour les cinq lancements. Pour le cas de graphes mixtes avec des chemins de longueur différente au nombre de cliques maximales, 23 instances sur les 35 instances testées sont résolues de manière optimale pour les cinq lancements.

Enfin, une analyse des résultats relativement à la moyenne de pourcentage de déviation par rapport à la borne inférieure (PDB), permet de déduire que l'algorithme du recuit simulé donne de meilleures performances dans le cas du premier type de graphe (12.45%) comparativement au second type de graphe (14.64%).

(V , A , E , Cl , Ch)	LB	$\bar{\chi}$	CPU(s)	(V , A , E , Cl , Ch)	LB	$\bar{\chi}$	CPU(s)		
(200,190,1235,15,10)	Rand1	23	25*	19	(200,180,507,33,20)	Rand19	12	13*	24
	Rand2	22	25.8	19		Rand20	11	13*	24
	Rand3	21	25*	20		Rand21	12	13*	24
	Rand4	23	25*	19		Rand22	12	13*	24
	Rand5	22	26*	19		Rand23	11	14*	24
(200,190,1078,17,10)	Rand6	22	25.4	19	(200,180,475,35,20)	Rand24	11	12*	25
	Rand7	21	25*	19		Rand25	12	13*	24
	Rand8	23	25*	20		Rand26	12	13*	26
	Rand9	22	24.6	19		Rand27	11	12*	26
	Rand10	21	25	20		Rand28	11	12*	26
(200,190,1012,18, 10)	Rand11	22	25*	20	(200,180,400,40,20)	Rand29	11	12*	26
	Rand12	22	25*	19		Rand30	11	12*	26
	Rand13	21	26*	19		Rand31	12	14	31
	Rand14	22	25*	19		Rand32	12	14	31
	Rand15	21	25*	19		Rand33	12	13.2	31
(200,180,507,33,20)	Rand16	11	13*	24	(250,225,660,40,25)	Rand34	11	13.4	31
	Rand17	12	13*	24		Rand35	13	13.4	31
	Rand18	12	13*	24					

Tableau 3 : Coloration de graphes mixtes par le recuit simulé : chemins de longueur différente au nombre de cliques

8 CONCLUSION

Dans ce papier, nous avons étudié l'ordonnement d'atelier job shop en considérant que toutes les opérations ont des durées unitaires. Ce problème est modélisé par la recherche de la coloration optimale d'une classe spéciale de graphes mixtes, composés d'arcs et d'arêtes, où le graphe partiel engendré par les arcs représente une union de chemins et le graphe partiel engendré par les arêtes représente une union de cliques maximales.

Pour résoudre le problème, nous avons d'abord proposé une formulation mathématique pour la coloration du graphe mixte, modélisant le problème d'ordonnement en utilisant la programmation linéaire en variables mixtes. La formulation proposée regroupe principalement trois

types de contraintes : les contraintes relatives au nombre chromatique mixte, les contraintes liées à la coloration des sommets de chaque chemin, et les contraintes liées à la coloration des sommets de chaque clique maximale. Ensuite, nous avons proposé une borne inférieure pour le nombre chromatique mixte du graphe relatif au problème d'ordonnement étudié.

Vu l'explosion combinatoire du nombre de solutions qui croît exponentiellement avec la taille du problème, l'utilisation de la méthode exacte, par la résolution des modèles mathématiques proposés, est de toute évidence limitée. Le recours à une méthode approchée est nécessaire. Notre choix s'est porté alors sur la proposition d'une approche basée sur l'algorithme du recuit simulé.

D'un autre côté, la borne inférieure proposée a permis de déduire l'optimalité de 12 Benchmarks résolus par l'algorithme du recuit simulé et non résolus par le modèle mathématique à savoir : orb7m, la11m, la13m, la15m, la20m, la24m, la26m, la28m, la31m, ft06m, svw17m et svw20m. De même le modèle mathématique a aidé à déduire l'optimalité de six Benchmarks résolus à la fois par l'algorithme du recuit simulé et par le modèle et dont le coût de la solution est différent de la valeur de la borne inférieure : la16m, la18m, la19m, la20m, orb4m et abz5m.

Nous concluons que les résultats des expérimentations réalisées, montrent l'efficacité du modèle mathématique proposé, et justifient également de l'efficacité de l'algorithme du recuit simulé tout en attestant de la qualité de la borne inférieure.

Comme perspective à ces travaux, nous envisageons de proposer une formulation mathématique basée sur une indexation de la couleur au niveau des variables décisionnelles qui permettrait ainsi de modéliser le problème étudié sans avoir recours aux contraintes disjonctives, ces dernières engendrent en effet une explosion du nombre de cas à traiter séparément, lorsque le nombre de disjonction augmente.

REFERENCES

- Al-Anzi, F. S., Sotskov, Y. N., Allahverdi, A., & Andreev, G. V. (2006). Using mixed graph coloring to minimize total completion time in job shop scheduling. *Applied Mathematics and Computation* 182, 1137–1148.
- Barnier N, and Brisset P. Graph coloring for air traffic flow management. *Annals of Operation Research*, 130(1):163–178, 2004.
- Chaitin, G. J., M. A. Auslander, A. K. Chandra, J. Cocke, M. E. Hopkins, & P. W. Markstein (1981). Register Allocation via Coloring. *Computer Languages* 6:47–57.
- Hanna, F., Adrian, K., & Pawel, Z. (2008). A note on mixed tree coloring. *Information processing letters* 106, 133–135.
- Hansen, Kuplinsky, J., & de Werra, D. (1997). Mixed graph colorings. *Math. Methods Oper. Res.* 45, 145–160.
- Lenstra, J., & Rinnooy Kan, A. (1979). Computational complexity of discrete optimization problems. *Annals of Discrete Mathematics*, 4, 121–140.
- Lim, A., Wang, F., 2005. Robust graph coloring for uncertain supply chain management. In: *Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS'05)-Track 3-Volume 03*, vol. 11(8), IEEE Computer Society, pp. 263–277.
- Maan V. and Sangwan D. “A Genetic Implementation of graph Coloring for cellular networks” *International Journal of electronics and engineering* 2(1), 2010, pp. 183–187.
- Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., & Teller, E. (1953). Equation of state calculations by fast computing machines. *J. Chem. Phys.* 21, 1087–1092.
- Ries, B. (2007). Coloring some classes of mixed graphs. *Discrete Applied Mathematics* 155, 1–6.
- Ries, B., de Werra, D. (2008). On two coloring problems in mixed graphs. *European Journal of Combinatorics* 29, 712–725.
- Rong Qu, Edmund K. Burkea, Barry McCollum (2009) Adaptive automated construction of hybrid heuristics for exam timetabling and graph colouring problems. *European Journal of Operational Research*. 198(2), 392–404.
- Sotskov, Y., Dolgui, A., & Werner, F. (2001). Unit-time Job-shop Scheduling via Mixed Graph Coloring. *International Journal of Mathematical Algorithms* 2, 4, 289–325.
- Sotskov, Y., & Tanaev, V. S. (1976). Chromatic polynomial of a mixed graph. *Vestsi Akademii Navuk BSSR, Seriya Fizika- Matematychnykh Navuk* 6, 20–23.
- Timkovsky, V. G. (1985). On the complexity of scheduling an arbitrary system. *Soviet Journal of Computer and Systems Sciences*, 5, 46–52.