



Minimizing the Number of Tardy Jobs in a Two-Machine flowshop problem with Non-Simultaneous Job Entrance

G. Moslehi*, A. hakimian & M. Abouei Ardakan

Ghasem Moslehi, Professor Isfahan University of Technology, moslehi@cc.iut.ac.ir

Ali hakimian, M.Sc student Isfahan University of Technology, a.hakimian@in.iut.ac.ir

Mostafa Abouei Ardakan, PhD student Isfahan University of Technology, m.abouei@in.iut.ac.ir

Keywords

Two machine Flowshop,
Number of tardy, Non-
Simultaneous Job Entrance,
Branch and Bound Algorithm,
Heuristic Algorithm

ABSTRACT

In this paper, minimizing the number of tardy jobs in two-machine flowshop scheduling with non-simultaneous job entrance is discussed. It is proven that the complexity of the problem is NP-hard. Therefore, a heuristic algorithm is proposed to solve the large scale problems. Besides, an exact branch and bound algorithm with utilizing heuristic algorithm as upper bound proposed to achieve optimal solution. Computational results demonstrate that branch and bound method solves problems with 28 jobs in the set High and 20 jobs in the set Low in a reasonable time. Results show the capability of the proposed upper bound, lower bounds and dominance rules. Also, it is shown that the average ratio of optimal solution to the heuristic one with the objective $\sum(I-U_i)$ is at most 1.11 which is smaller in contrast with other researches in the literature. This ratio proves efficacy of the proposed heuristic algorithm. Finally, according to efficiency of the presented approach, sample problems with large dimensions were solved and their results were displayed.

© 2013 IUST Publication, IJIEPM. Vol. 23, No. 4, All Rights Reserved

*
Corresponding author: Ghasem Moslehi
Email: moslehi@cc.iut.ac.ir



حداقل کردن تعداد کارهای دارای دیرکرد در مسئله دو ماشین با ورود غیر همزمان

قاسم مصلحی*، علی حکیمیان و مصطفی ابویی اردکان

چکیده:

در این مقاله مسئله زمان‌بندی فلوشاپ دو ماشین با در نظر گرفتن ورود غیر همزمان و با هدف کمینه‌سازی تعداد کارهای دارای دیرکرد بررسی شده است. در ابتدا پیچیدگی مسئله بررسی و ثابت شده که مسئله NP-hard است. بنابراین برای حل مسئله فوق یک الگوریتم ابتکاری که قابلیت حل مسائل با ابعاد خیلی بزرگ را دارد، ارائه شده است. همچنین به منظور حل بهینه مسئله از روش شاخه و کران با در نظر گرفتن الگوریتم ابتکاری به عنوان حد بالا بهره گرفته شده است. نتایج محاسباتی نشان می‌دهد که رویه شاخه و کران مسائل با ابعاد ۲۸ فعالیت در گروه High و ۲۰ فعالیت در گروه Low را در زمان منطقی و به طور کامل حل می‌کند، که این امر کارآیی حد بالا، حدود پایین و اصول غلبه ارائه شده برای مسئله را نشان می‌دهد. همچنین نشان داده شد که متوسط نسبت جواب بهینه به الگوریتم ابتکاری با هدف $\sum_{i=1}^{11} U_i$ مرتبط با کارهای دارای دیرکرد نسبت کوچکی می‌باشد. این نسبت نشان دهنده کارایی بالای الگوریتم ابتکاری است. با توجه به کارآیی بالای الگوریتم ابتکاری، مسائل نمونه با ابعاد بزرگ نیز حل و نتایج آن ارائه شده است.

کلمات کلیدی

فلوشاب دو ماشین

تعداد کارهای دارای دیرکرد

ورود غیر همزمان

الگوریتم شاخه و کران

الگوریتم ابتکاری

معیارهای ارزیابی زمان‌بندی‌ها مورد استفاده قرار می‌گیرند [۱]. در این مقاله مسئله زمان‌بندی فلوشاپ دو ماشین با زمان‌های ورود غیر همزمان و با هدف حداقل کردن تعداد کارهای دارای دیرکرد در حالت فلوشاپ جایگشتی^۲ مورد بررسی قرار می‌گیرد. در این حالت هر کاری که روی ماشین اول زودتر پردازش شود، روی ماشین دوم نیز زودتر پردازش می‌شود. به عبارت دیگر هیچ کاری مانند Z_j وجود ندارد که پردازش آن روی ماشین اول بعد از Z_i باشد ولی پردازش آن روی ماشین دوم قبل از Z_i صورت پذیرد. کاربردهایی از این مسئله را می‌توان در کارخانه‌های نساجی و یا کارخانه‌های تولید آینه یافت. در فرایند تولید پارچه، پارچه‌ها (کارها) بر روی دو ماشین رنگ زنی و خشک کن و به صورت

۱. مقدمه

در بسیاری از صنایع برآورد کردن زمان‌های تحويل امری حیاتی می‌باشد. لذا معیارهای وابسته به تاخیر همانند تعداد کارهای دارای دیرکرد از اهمیت ویژه‌ای برخوردار هستند و به عنوان

تاریخ وصول: ۹۰/۲/۱۸

تاریخ تصویب: ۹۰/۵/۳۰

*نویسنده مسئول مقاله: قاسم مصلحی، استاد، دانشکده مهندسی صنایع و سیستم‌ها، دانشگاه صنعتی اصفهان moslehi@cc.iut.ac.ir
علی حکیمیان، دانشجوی کارشناسی ارشد، دانشکده مهندسی صنایع و a.hakimian@in.iut.ac.ir
مصطفی ابویی اردکان، دانشجوی دکتری، دانشکده مهندسی صنایع و m.abouei@in.iut.ac.ir

².Permutation Flowshop

садیکف [۱۲] یک روش شاخه و کنترل^۴ را برای حل مسئله حداقل سازی وزنی تعداد کارهای دارای دیرکرد بر روی تک ماشین و با محدودیت زمان در دسترس بودن ارائه کردند. آنها در تحقیق خود از برنامه‌ریزی اعداد صحیح برای حل روش شاخه و کنترل استفاده کردند و جواب‌های امکان ناپذیر را توسط الگوریتم جدا کردند. هلاه و بولفین [۱۳] نیز برای حل همین مسئله یک روش شاخه و کران که در آن یک ساده‌سازی جانشینی که منجر به یک مسئله کوله پشتی چند انتخابه^۵ می‌شود پیشنهاد دادند. آنها از همین روش برای حداقل کردن وزنی تعداد کارهای دارای دیرکرد برروی پردازشگرهای موازی نیز استفاده کردند [۱۴].

هی و همکاران [۱۵] روبکردن دوگانه از الگوریتم حریصانه رویه جلو^۶ را برای حداقل کردن تعداد کارهای دارای دیرکرد در مسئله تک ماشین ارائه کردند. آنها برای هر کار یک ضرب العجل^۷ نیز در نظر گرفتند. الگوریتم ارائه شده از رتبه $O(n\log n)$ می‌باشد. بولفین و هلاه [۱] با استفاده از مسئله کوله پشتی در حدود روش شاخه و کران مسئله حداقل سازی وزنی تعداد کارهای دارای دیرکرد در یک فلوشاپ دو ماشین، توانستند مسائلی تا ۱۰۰ کار حل نمایند. لودری و همکاران [۱۶] الگوریتمی ابتکاری برای حداقل سازی تعداد کارهای دارای دیرکرد در یک فلوشاپ پویا ارائه کردند. آنها فرض کردند که زمان ورود کارها متفاوت باشد. رويز تورس و سنتنو [۱۷] ، مسئله حداقل کردن تعداد کارهای دارای دیرکرد برروی یک فلوشاپ جایگشتی و با منابع ثانویه^۸ را مورد بررسی قرار دادند. آنها برای تخصیص منابع ثانویه و تخصیص کارها به توالی از الگوریتم‌های ابتکاری مانند جستجو همسایگی و SA استفاده کردند. رويز تورس و همکاران [۱۸] در سال ۲۰۱۰ از روشی که در سال ۲۰۰۸ استفاده کرده بودند بار دیگر برای حل مسئله حداقل کردن تعداد کارهای دارای دیرکرد در یک فلوشاپ با منابع و عملیات انعطاف پذیر بهره بردن.

چوی و کیم [۱۹] ، فلوشاپ دو ماشینی را در نظر گرفتند که در آن هر کار دومرتبه به فلوشاپ وارد می‌شود. یعنی هر کار باید یک مرتبه روی ماشین یک و سپس ماشین دو پردازش شود و دوباره تکرار شود. آنها میزان مجموع دیرکرد را در این مسئله با استفاده از روش شاخه و کران حداقل کردند. چوی و لی [۲۰] ، مسئله حداقل کردن تعداد کارهای دارای دیرکرد در یک هیبرید فلوشاپ دو مرحله‌ای را با استفاده از روش شاخه و کران حل کردند. آنها برای حل مسئله در اندازه‌های بزرگ یک الگوریتم ابتکاری دو

متوالی پردازش می‌شوند. در فرایند نفره کاری برای تولید آینه نیز از دو ماشین متوالی استفاده می‌شود. ماشین اول کار پوشش دادن پشت آینه‌ها (کارها) توسط رنگ را بر عهده دارد و ماشین دوم عملیات خشک کردن را انجام می‌دهد.

اکثر پژوهش‌ها در مسائل فلوشاپ برروی حداقل سازی زمان اتمام کل^۹ تمرکز داشته‌اند [۲۰-۲۳] و تاکنون بررسی‌های کمتری برای حداقل سازی تعداد کارهای دارای دیرکرد برروی مسئله فلوشاپ انجام پذیرفته است. پیچیدگی حداقل سازی تعداد کارهای دارای دیرکرد در یک فلوشاپ دو ماشین حتی موقعی که تمام کارهای دارای زمان تحويل یکسان باشند نیز NP-hard است [۴]. حالت بدون وزن و تک ماشین این مسئله، با الگوریتم مور^{۱۰} O(nLogn) قابل حل می‌باشد [۵] و الگوریتم‌های بهینه فقط برای حالت‌های خاص توسعه داده شده‌اند.

لین [۶] یک الگوریتم از $O(n^2)$ را برای حل حالت‌های خاص مسئله فلوشاپ دو ماشین وقتی که همه کارها زمان‌های پردازش یکسان روی ماشین اول داشته و زمان‌های تحويل با زمان‌های پردازش سازگاری دارند ارائه کرد. لاولر و مور [۷] یک الگوریتم بهینه برنامه‌ریزی پویا را برای همین مسئله با محدود تحويل یکسان $D = D(\sum U_j)$ ارائه کردند که از $O(nD^2)$ می‌باشد.

حریری و پتس [۸] مسئله حداقل کردن تعداد کارهای دارای دیرکرد در فلوشاپ جایگشتی با M ماشین را با الگوریتم شاخه و کران حل کردند، در این مسئله کلیه کارها در زمان صفر در دسترس هستند. آنها مسائل با ابعاد حداکثر ۲۵ کار را برای دو و سه ماشین حل کردند. مصلحی و همکاران [۹]، مسئله حداقل کردن مجموع حداکثر دیرکردها و زودکردها^{۱۱} برروی یک فلوشاپ دو ماشین را بررسی کردند. آنها برای حل مسئله از روش شاخه و کران استفاده کردند و برای افزایش کارایی روش شاخه و کران از لم‌هایی برای یافتن جواب‌های غالب استفاده کردند.

وان وین [۱۰]، مسئله حداقل کردن تعداد کارهای دارای دیرکرد و حداقل کردن مجموع زودکردهای وزن دار را در یک ماشین بررسی کردند. آنها برای حل مسئله از روش شاخه و کران استفاده کرده و برای حل مسائل بزرگ از یک الگوریتم ابتکاری با رتبه $O(n^3)$ بهره جستند.

باپتیست و همکاران [۱۱] یک الگوریتم شاخه و کران را برای مسئله حداقل سازی تعداد کارهای دارای دیرکرد در تک ماشین و با محدودیت زمان در دسترس بودن ارائه کردند. در این تحقیق، حد پایین از ساده سازی لاغرانژی به دست آمده و با ارائه یک الگوریتم شاخه و کران قادر به حل مسئله تا اندازه ۲۰۰ کار شدند.

^۱.Makespan

^۲.Moore

^۳.sum of maximum Earliness and Tardiness

مسئله با فرض ورود غیر همزمان کارها تاکنون به صورت بهینه در بیشتر از یک ماشین حل نشده است، در این مقاله این مسئله بر روی فلوشاپ دو ماشین و به صورت بهینه حل و الگوریتمی ابتکاری نیز برای حل مسائل با اندازه بزرگ ارائه می‌شود. این مسئله از این به بعد با نماد $\sum_{j=1}^n P_j t_j$ نمایش داده می‌شود. نمادهای مورد نیاز این مسئله در زیر بیان شده است:

N : مجموعه کارها، $\{1, 2, 3, \dots, n\}$ که n تعداد کل کارها است.

$m=1, 2$: اندیس ماشین

$j=1, 2, 3, \dots, n$: زمان در دسترس کار j

P_{mj} : زمان پردازش کار j روی ماشین m
 $j=1, 2, 3, \dots, n$ & $m=1, 2$

$P_{mj[j]}$: زمان پردازش کار در موقعیت j روی ماشین m
 $j=1, 2, 3, \dots, n$ & $m=1, 2$

d_j : موعد تحویل کار j
 $j=1, 2, 3, \dots, n$

C_{mj} : زمان تکمیل کار j روی ماشین m
 $j=1, 2, 3, \dots, n$ & $m=1, 2$

$C_{mj[j]}$: زمان تکمیل کار در موقعیت j روی ماشین m
 $j=1, 2, 3, \dots, n$ & $m=1, 2$

$$C_{2[j]} = \max(C_{1[j]} + P_{2[j]}, C_{2[j-1]} + P_{2[j]}) \quad (1)$$

$$T_{jj} : \text{میزان دیرکرد}^2 \text{ کار در موقعیت } j \quad j=1, 2, 3, \dots, n$$

$$T_{[j]} = \max\{0, C_{2[j]} - d_{j[j]}\} \quad (2)$$

$j=1, 2, 3, \dots, n$: اگر $C_{2[j]} - d_j$ برابر با 0 و در غیراین صورت
برابر با 1

σ_L : توالی جزئی از کارهای چیده شده در گره L

L : کار چیده شده در موقعیت آن توالی

$N_T(\sigma_L)$: تعداد کارهای دارای دیرکرد در توالی جزئی σ_L که از رابطه زیر به دست می‌آید

مرحله‌ای معرفی کردند. فلوشاپ آنها شامل دو مرحله بود که در مرحله اول M_1 ماشین موازی و یکسان و مرحله دوم M_2 ماشین موازی و یکسان وجود داشت.

موشیف و ساریگ [۲۱]، مسئله فلوشاپ را مطالعه کردند که در آن کارها نیاز به دو عملیات داشتند. عملیات اول بروی تک ماشین مرحله اول (ماشین بحرانی) و عملیات دوم بر روی یکی از M ماشین مرحله دوم صورت می‌پذیرد. آنها برای حل مسئله مورد نظر یک الگوریتم برنامه‌ریزی پویای چند جمله‌ای کاذب و یک الگوریتم ابتکاری موثر ارائه کردند. هدف این مقاله حداقل کردن وزنی تعداد کارهای دارای دیرکرد بود.

همانطور که مشاهده شد، مسئله حداقل سازی تعداد کارهای دارای دیرکرد بر روی فلوشاپ جایگشتی با دو ماشین و زمان‌های ورود غیر همزمان کارها در ادبیات موضوع مشاهد نگردید. لذا در این مقاله این مسئله مورد بررسی قرار می‌گیرد و برای حل سریع آن یک روش ابتکاری و برای پیدا کردن جواب بهینه یک روش شاخه و کران ارائه می‌شود.

ادامه مقاله به این صورت سازماندهی شده است. در بخش دوم مسئله مورد نظر معرفی می‌شود، در بخش سوم رویکرد حل مسئله بیان می‌شود، در بخش چهارم نتایج محاسباتی ارائه می‌شود و در پایان در بخش پنجم نتیجه‌گیری و تحقیقات آینده ارائه می‌شوند.

۲. تعریف و پیچیدگی مسئله

در این بخش، مسئله حداقل سازی تعداد کارهای دارای دیرکرد بر روی فلوشاپ جایگشتی با دو ماشین و ورود غیر همزمان کارها به همراه پارامترها، متغیرها و فرضیات مسئله ارائه می‌شود. همانطور که در مقدمه اشاره شد، این مسئله با تأکید بر روی دو ماشین در ادبیات موضوع مشاهده نشده است، لذا در این مقاله مسئله مذکور مورد بررسی قرار می‌گیرد. مجموعه N شامل n کار مستقل $\{1, 2, 3, \dots, n\}$ می‌باشد. این کارها باید بر روی دو ماشین پردازش شوند.

تمام کارها ابتدا بر روی ماشین اول و سپس بر روی ماشین دوم پردازش می‌شوند. همچنین هر کاری که بر روی ماشین اول زودتر پردازش شود بر روی ماشین دوم هم زودتر پردازش می‌شود. به عبارت دیگر برنامه کاری دو ماشین مشابه هستند.

کارها دارای ورود غیر همزمان هستند و هر کار در زمان خاصی برای انجام پردازش در اختیار می‌باشد که به آن زمان در دسترس 1 گفته می‌شود. هدف پیدا کردن یک توالی از کارها می‌باشد که در آن تعداد کارهایی که دارای دیرکرد هستند حداقل شوند. این

². Tardiness

¹. Release Time

$$\sum_{i=1}^n x_{ij} = 1 \quad j = 1, 2, 3, \dots, n \quad (5)$$

$$C_{1[j]} \geq \sum_{i=1}^n r_i x_{ij} + \sum_{i=1}^n p_{ii} x_{ij} \quad j = 1, 2, 3, \dots, n \quad (6)$$

$$C_{1[j]} \geq C_{1[j-1]} + \sum_{i=1}^n p_{ii} x_{ij} \quad j = 1, 2, 3, \dots, n \quad (7)$$

$$C_{2[j]} \geq C_{1[j]} + \sum_{i=1}^n p_{2i} x_{ij} \quad j = 1, 2, 3, \dots, n \quad (8)$$

$$C_{2[j]} \geq C_{2[j-1]} + \sum_{i=1}^n p_{2i} x_{ij} \quad j = 1, 2, 3, \dots, n \quad (9)$$

$$T_{ij} \geq 0 \quad j = 1, 2, 3, \dots, n \quad (10)$$

$$T_{ij} \geq C_{2[j]} - \sum_{i=1}^n d_i x_{ij} \quad j = 1, 2, 3, \dots, n \quad (11)$$

$$T_{ij} \leq M \cdot U_{ij} \quad j = 1, 2, 3, \dots, n \quad (12)$$

$$x_{ij} \in \{0, 1\} \quad i, j = 1, 2, 3, \dots, n \quad (13)$$

$$U_{ij} \in \{0, 1\} \quad j = 1, 2, 3, \dots, n \quad (14)$$

رابطه (6) تضمین می‌کند که کلیه کارها حتماً یک مرتبه در یکی از n موقعیت قرار بگیرند. رابطه (7) نشان می‌دهد که هر موقعیت مانند j حتماً باید شامل یک کار باشد. محدودیت (8) و (9)، زمان تکمیل کارها را بر روی ماشین اول تعیین می‌کند. محدودیت (10) و (11) معین کننده زمان تکمیل کارها بر روی ماشین دوم می‌باشند. روابط (12) تا (14) حدود متغیر T_{ij} را تعیین می‌نماید. محدودیت‌های (15) تا (16) شرایط متغیرهای مساله را نشان می‌دهند.

از آنجا که اثبات شد این مدل NP-hard است، لذا برای حل مدل از روش شاخه و کران استفاده شده است. در بخش بعدی رویکرد حل مسئله توضیح داده می‌شود.

۳. رویکرد حل

از آنجائی که پیچیدگی مسئله $P2|r_j|\Sigma U_j$ به صورت NP-hard است، لذا منطقی است که برای حل بهینه مسئله از روش شاخه و کران استفاده نمود. در روش شاخه و کران از حدود بالا و پایین

$$NT(\sigma_L) = \sum_{j \in \sigma_L} U_j \quad (1)$$

$NT(\sigma'_L)$: تعداد دیرکرد برای کارهایی که در توالی جزئی σ_L قرار ندارند:

$$NT(\sigma'_L) = \sum_{j \notin \sigma_L} U_j \quad (2)$$

σ' : توالی جزئی حاصل از مجموعه σ'_L با استفاده از الگوریتم مور بر روی ماشین m ، $m=1, 2$ در این مقاله فرض شده است که تمام کارها بدون انقطاع انجام می‌شوند. همچنین بیکاری عمده برای ماشین مجاز می‌باشد، به عبارت دیگر می‌توان انجام کاری را که در دسترس است به صورت عمده به تعویق انداشت و ماشین برای مدتی بیکار بماند تا کار دیگری برای پردازش برسد.

فرض بیکاری عمده موقعی که کارها دارای زمان ورود غیر همزمان هستند می‌تواند باعث کاهش تعداد کارهای دارای دیرکرد شود هرچندکه حل مسئله را دشوارتر می‌کند. باپتیست و همکاران نشان دادند که مسئله $J_1|r_j|\Sigma U_j$ است NP_hard [11]. از آنجا که این مسئله تقلیل یافته مسئله $P2|r_j|\Sigma U_j$ است NP_hard می‌باشد، بنابراین آن هم حداقل دارای پیچیدگی است.

۲-۱. مدل برنامه ریزی خطی

در این بخش مدل برنامه ریزی خطی عدد صحیح مختلط مسئله مورد نظر ارائه می‌شود. در ابتدا متغیرها و پارامتر مورد استفاده در مدل شرح داده می‌شوند. سپس مدل طراحی شده ارائه و تابع هدف و محدودیت‌های مدل تشریح می‌شوند.

اگر کار i در موقعیت j پردازش شود $x_{ij} = 1, 2, 3, \dots, n$ یک و در غیر این صورت صفر است.

M : یک عدد بزرگ

۲-۱-۱. مدل برنامه ریزی عدد صحیح مختلط

$$\text{Minimize } NT = \sum_{j=1}^n U_j \quad (3)$$

s.t.

$$\sum_{j=1}^n x_{ij} = 1 \quad i = 1, 2, 3, \dots, n \quad (4)$$

نسبت به دو الگوریتم دیگر کارایی بهتری داشته، از آن به عنوان الگوریتم برتر و همچنین به عنوان حد بالا استفاده شده است.

۲-۳. الگوریتم شاخه و کران

برای به دست آوردن جواب بهینه مسئله $\sum_{j=1}^m U_j$ از روش شاخه و کران استفاده شده است. در این الگوریتم هر گره از درخت جستجو یک توالی جزئی σ_L را نشان می‌دهد. علاوه بر این هر گره شامل تعداد کارهای زمان‌بندی شده،تابع هدف توالی جزئی چیده شده $NT(\sigma_L)$. یک تخمین برای تابع هدف کارهای چیده نشده (تخمینی از NT کارهای چیده نشده) و یک حد پایین به صورت جمع تابع هدف تخمینی برای کارهای چیده نشده و تابع هدف توالی جزئی چیده شده می‌باشد.

برای شاخه زدن از روش جستجوی عمقی^۳ استفاده شده و در آن از سه حد پایین و از الگوریتم H_1 به عنوان حد بالا استفاده شده است. همچنین چندین اصل غلبه برای افزایش کارائی روش ارائه شده است.

در سطح k ام الگوریتم با توالی جزئی کارهای $\sigma_{[1],...,k}$ شاخه‌های جدید، از طریق اضافه شدن کارهای زمان‌بندی نشده به آخر توالی σ_L به وجود می‌آیند که این کارها می‌بایستی شرایط زمان ورود و اصول غلبه را ارضاء کرده باشند. هنگامی که هیچ کاری برای برنامه‌ریزی وجود نداشته باشد، توالی σ_L یک زمان‌بندی کامل خواهد بود و گره مد نظر به آخرین سطخ خود رسیده است. در طی فرایند جستجو میزان خدمت برای بهترین زمان‌بندی یافت شده تاکنون با حد پایین مقایسه و به هنگام می‌شود.

لم ۱: یک حد پایین برای مسئله $\sum_{j=1}^m U_j$ به صورت زیر است:

$$LB_1 = NT(\sigma_L) + NT(\sigma'_{L-MOORE2}) \quad (15)$$

که در آن $NT(\sigma_L)$ تعداد کارهای دارای دیرکرد در توالی جزئی σ_L و $NT(\sigma'_{L-MOORE2})$ تعداد کارهای دارای دیرکرد در توالی $\sigma'_{L-MOORE2}$ می‌باشد. که این توالی بر اساس الگوریتم مور و از روی ماشین دوم به دست می‌آید.

اثبات: در گره‌ای مثل L ، کارهای چیده شده با σ_L و کارهای چیده نشده با σ'_L نشان داده می‌شوند. حد پایین LB_1 شامل دو قسمت می‌باشد.

قسمت اول تعداد کارهای دارای دیرکرد از توالی جزئی چیده شده σ_L می‌باشد و با $NT(\sigma_L)$ نشان داده می‌شود. قسمت دوم تخمینی

مناسب و اصول غلبه کارایی استفاده شده، تا بتوان با قطع^۱ کردن سریع شاخه‌ها در زمان مناسب به جواب بهینه دست یافت. در ادامه یک الگوریتم ابتکاری معرفی می‌شود که به علت عملکرد خوب آن، به عنوان حد بالا در روش شاخه و کران نیز استفاده شده است.

۱-۳. الگوریتم ابتکاری H_1

اجرای الگوریتم مور- هاجسون برای مسئله $\sum_{j=1}^m U_j$ منجر به جواب بهینه می‌شود [۵]. لذا با الهام از این الگوریتم، برای مسئله $\sum_{j=1}^m U_j$ یک الگوریتم ابتکاری به نام H_1 ارائه شده است. قدم-

های الگوریتم H_1 به صورت زیر است:

(۱) ابتدا کارها به صورت زودترین موعد تحويل (EDD^۲) مرتب شوند.

(۲) سپس کارها براساس ترتیب EDD و با توجه به زمان در دسترس هر کار (r_i)، به ماشین اول و دوم تخصیص داده می‌شوند و زمان تحويل هر کار محاسبه شود.

(۳) اگر اولین کار دیرکرددار در موقعیت ۱ باشد، از ابتدای ترتیب تا موقعیت ۱ هر دفعه یکی از کارها حذف شده و میزان صرفه جویی در زمان ختم کار در موقعیت ۱ به دست آورده شود. کاری که حذف آن منجر به بیشترین کاهش در زمان ختم کار در موقعیت ۱ بر روی ماشین دوم می‌شود در آخر توالی قرار گرفته و این کار دارای دیرکرددار است. هر موقع کاری به آخر توالی برده می‌شود از تعداد کل کارها یکی کم می‌شود. (یعنی اولین کار حذف شده در آخر توالی قرار می‌گیرد و دومین کار حذف شده در مرحله بعدی در موقعیت $n-1$ و به همین ترتیب محل کارهای دارای دیرکرد بعدی مشخص می‌شود).

(۴) برای بقیه کارها مجدداً قدم های ۲ و ۳، مدامی که کار دیرکرددار یافت شود اجرا می‌شوند.

نتایج عددی نشان می‌دهد که این الگوریتم کارائی خوبی دارد. دو الگوریتم ابتکاری دیگر به نامهای H_2 و H_3 نیز مورد بررسی قرار گرفته‌اند. در H_2 ، به جای کار حذف شده در قدم سوم الگوریتم H_1 ، کار با بیشترین زمان پردازش روی ماشین اول و دوم حذف می‌شود.

به عبارت دیگر کاری با بیشترین $P_{1,j} + P_{2,j}$ حذف می‌شود. در الگوریتم H_3 کار با بزرگترین زمان پردازش روی ماشین دوم حذف شده و به آخر توالی برده می‌شود. از آنجا که الگوریتم H_1

¹fatom

²Earliest Due Date

³. Back-Tracking

اثبات: مشابه لم ۱. \square

لم ۳: برای به دست آوردن یک حد پایین در مسئله $P2|rj|\sum U_j$ می توان به صورت زیر عمل کرد:
تعداد کارهای دارای دیرکرد در توالی جزئی چیده شده σ_L همانند لمهای قبل محاسبه شود. برای تخمین تعداد کارهای دارای دیرکرد در مجموعه σ'_L به صورت زیر عمل شود.

$$LB_3 = NT(\sigma_L) + NT(\sigma'_L) \quad (18)$$

$$NT(\sigma'_L) = \sum_{i \in \sigma_L} \delta(T'_{[i]}) \quad (19)$$

که $\delta(T'_{[i]})$ برابر است با،

$$\delta(T'_{[i]}) = \begin{cases} 0 & \text{if } T'_{[i]} = 0 \\ 1 & \text{if } T'_{[i]} > 0 \end{cases} \quad (20)$$

اگر کارهای مجموعه σ'_L یکبار بر اساس زمانهای پردازش ماشین اول و یکبار هم بر اساس ماشین دوم و به ترتیب از زمان شروع A و B چیده شوند برای هر موقعیت دو زمان تکمیل به دست می آید که حداکثر بین هر دو، به عنوان زمان تکمیل کار در موقعیت مربوطه در نظر گرفته می شود و به صورت $C_{[i]}$ نمایش داده می شود. از طرف دیگر اگر کارهای مجموعه σ'_L به ترتیب $d_{[i]}$ EDD چیده شوند و موعد تحويل به دست آمده به صورت $d_{[i]}$ نمایش داده شود می توان دیرکرد هر موقعیت را به صورت زیر محاسبه نمود،

$$T'_{[i]} = MAX \{0, C_{[i]} - Min_{j \geq i} (d_{[j]})\} \quad (21)$$

هرگاه برای یک موقعیت $\delta(T'_{[i]}) = 1$ شود، به تعداد کارهای دارای دیرکرد تخمینی یکی اضافه می شود و زمان ختم آن موقعیت با زمان تحويل بعدی که بزرگتر می باشد مقایسه می شود و این عمل برای تمام موعدهای تحويل ادامه می یابد. \square

لازم به یادآوری است که حد پایین مسئله، یعنی LB با توجه به سه حد پایین گفته شده بر اساس رابطه (۲۴) به دست می آید.

$$LB = MAX \{LB_1, LB_2, LB_3\} \quad (22)$$

لم ۴: اگر در گره L کار i از مجموعه کارهای چیده نشده برای چیدن در انتهای σ_L انتخاب شود و کار i در این موقعیت دارای دیرکرد شود، جواب بهینه‌ای وجود دارد که در آن کار i در آخر توالی قرار می گیرد.

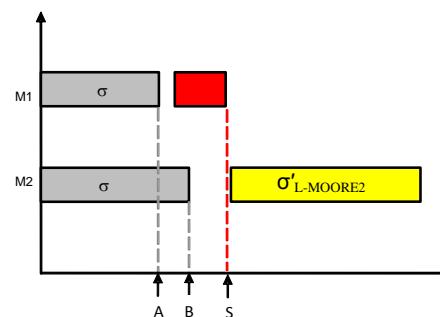
از تعداد کارهای دارای دیرکرد از کارهای چیده نشده σ'_L است که با $NT(\sigma'_{L-MOORE2})$ نشان داده می شود.

این تخمین با استفاده از روش مور-هاجسون در تک ماشین به دست می آید و برای این منظور فقط از زمان پردازش کارها روی ماشین دوم یعنی P_2 استفاده می شود. فرض کنید زمان در دسترس بودن برای شروع مجموعه σ'_L روی ماشین اول A و روی ماشین دوم B باشد. زودترین زمان شروع کارهای مجموعه σ'_L روی ماشین دوم برابر با S است که از رابطه (۱۸) به دست می آید.

$$S = \text{Max} \left\{ \text{Min}_{i \in \sigma'_L} \{ \text{Max}\{A, r_i\} + P_{1i} \}, B \right\} \quad (16)$$

از آنجا که در این روش فقط ماشین دوم در نظر گرفته می شود و اثر ماشین اول در ایجاد بیکاری روی ماشین دوم نادیده گرفته شده است، بنابراین تعداد کارهای دارای دیرکرد به دست آمده از روش مور-هاجسون یک حد پایین برای تعداد کارهای دارای دیرکرد مجموعه σ'_L است. با جمع این مقدار از تعداد کارهای دارای دیرکرد با تعداد کارهای دارای دیرکرد مجموعه σ_L ، یک حد پایین برای مسئله به دست می آید که با LB_1 نمایش داده می شود. \square

شکل ۱ نمایی از این حد پایین را نشان می دهد.

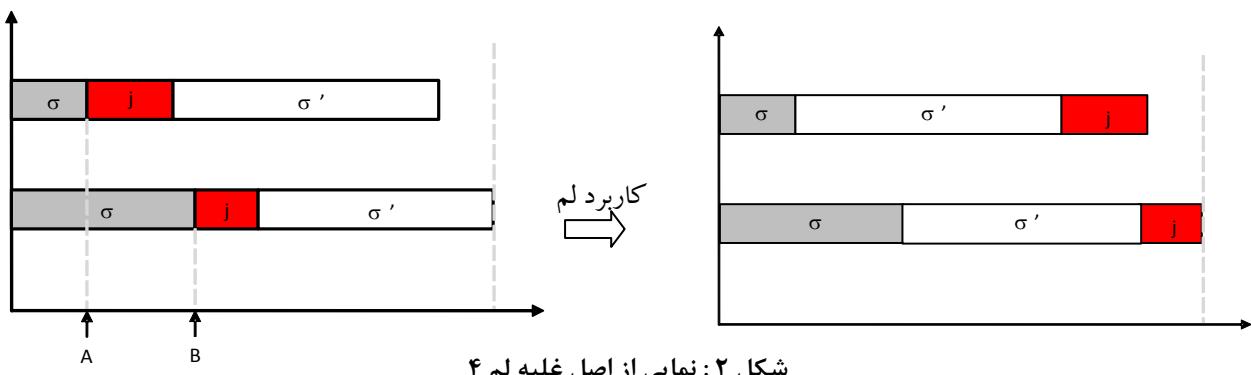


شکل ۱: نمایی از حد پایین لم ۱

لم ۲: یک حد پایین برای مسئله $P2|rj|\sum U_j$ به صورت زیر است:

$$LB_2 = NT(\sigma_L) + NT(\sigma'_{L-MOORE1}) \quad (17)$$

LM مشابه لم ۱ است که از روی ماشین اول به دست می آید. کارهای مجموعه σ'_L بر اساس الگوریتم مور-هاجسون از زمان شروع A چیده می شوند. همچنین برای به دست آوردن آخرین کار چیده شده در مجموعه σ'_L مقدار $\sigma'_{L-MOORE1}$ مدت زمان پردازش روی ماشین دوم از مجموعه σ'_L به آن اضافه می شود.



$$C_1(\sigma_{Lji}) \geq C_1(\sigma_{lij}) \quad (28)$$

$$C_2(\sigma_{Lji}) > C_2(\sigma_{lij}) \quad (29)$$

اثبات: با توجه به شرایط ذکر شده، زمان تکمیل توالی جزئی σ_{Lij} بر روی هر دو ماشین بدتر از زمان تکمیل توالی جزئی σ_{Lji} نمی‌باشد. بنابراین توالی σ_{Lij} سبب به تعویق افتادن شروع کارهای بعدی نمی‌شود و با توجه به این کهتابع هدف حداقل- کردن تعداد کارهای دارای دیرکرد یک تابع منظم است، توالی σ_{Lji} موجب افزایش تعداد کارهای دارای دیرکرد نمی‌شود. \square

۴. نتایج محاسباتی

الگوریتم شاخه و کران و الگوریتم ابتکاری در زبان برنامه نویسی Visual Basic 6 کدنویسی شده‌اند و بر روی یک کامپیوتر با 2 GB of RAM اجرا شده‌اند. برای بررسی کارائی روش ارائه شده مسائلی به صورت تصادفی و با استفاده از روش اشاره شده در مرجع [۱۶] تولید شده‌اند.

زمان‌های پردازش از توزیع یکنواخت گسسته در فاصله [۰,۷۵] تولید شده‌اند. مقادیر موعد تحويل از توزیع یکنواخت گسسته در بازه P_α, P_β (Pa, Pb) به دست می‌آید که مقدار P از رابطه (۳۲) زیر حاصل می‌شود.

$$P = (I/n)^* [\sum_{i=1}^n \sum_{j=1}^2 P_{ij} + (n-1)\max\{\sum_{i=1}^n P_{i1}, \sum_{i=1}^n P_{i2}\}] \quad (30)$$

مقادیر α و β که به ترتیب حد پایین و حد بالای موعدهای تحويل را نشان می‌دهد از دو مجموعه $\{\alpha, \beta\} \in \{0.4, 0.6\}$ به $\alpha \in \{0.8, 1\}$ دست می‌آیند.

زمان‌های در دسترس بودن یعنی r_i از توزیع یکنواخت گسسته (P_δ, P_ϵ) تولید شده که برای δ مقدار صفر و برای ϵ مقادیر $\{0.05, 0.25\}$ در نظر گرفته شده‌اند. برای داده‌ها دو گروه Low و High در نظر گرفته شده‌است. پارامترهای Low به

اثبات: واضح است که با انتقال کار دیرکردار به آخر، زمان ختم بقیه کارها افزایش نمی‌یابد و حتی ممکن است کاهش هم پیدا کند. بنابراین با توجه به این کهتابع هدف حداقل کردن تعداد کارهای دارای دیرکرد یک تابع منظم است، موجب افزایش تعداد کارهای دارای دیرکرد نمی‌شود. \square

لم ۵: با برقراری شروط ۲۵ تا ۲۷ حتماً یک جواب بهینه وجود دارد که کار j باید بعد از کار i پردازش شود.

$$R_i = \min_{k \neq i} \{r_k\} \quad (23)$$

$$r_i \geq A, r_j \geq r_i + P_{li} \quad (24)$$

$$r_i + P_{li} + P_{2i} \leq r_j + P_{lj} \quad (25)$$

اثبات: هنگامی که کارهای مجموعه σ_L چیده شده باشند و دو کار دارای کوچکترین زمان ورود بوده و روابط (۲۵) تا (۲۷) را نیز برآورده کنند مسلماً کار i باید قبل از j انجام شود چون انجام یا عدم انجام کار i در زمان شروع کار j تاثیری ندارد و بنابراین برای کاهش تعداد کارهای دارای دیرکرد، کار i حتماً باید قبل از کار j انجام شود. برای این منظور زمان‌های تکمیل دو توالی جزئی j و i بعد از σ_L به صورت روابط (۲۸) و (۲۹) محاسبه می‌شود.

$$C_2(\sigma_{lij}) = r_j + P_{lj} + P_{2j} \quad (26)$$

$$C_2(\sigma_{lji}) = r_j + P_{lj} + \max(P_{li}, P_{2j}) + P_{2i} \quad (27)$$

به سادگی دیده می‌شود که $C_2(\sigma_{lji}) > C_2(\sigma_{lij})$ ، بنابراین کار i باید قبل از j پردازش شود. \square

لم ۶: اگر دو کار i, j در دو توالی σ_{lij} و σ_{lji} دیرکرد نداشته باشند و روابط (۳۰) و (۳۱) برقرار باشد. آنگاه جواب بهینه‌ای وجود دارد که در آن کار j بعد از i پردازش می‌شود.

عبارت دیگر مسائل گروه Low مسائل سختتری نسبت به مسائل گروه High می‌باشند. جدول ۱ تفاوت‌های بین دو گروه مسئله را نشان می‌دهد.

صورت ($\alpha=0.6$, $\beta=0.8$, $\varepsilon=0.05$) و High ($\alpha=0.4$, $\beta=1$, $\varepsilon=0.25$). مسائل گروه High دارای تولید شده‌اند [۱۶]. مسائل گروه Low دارای شرایط بسیار پویاتری نسبت به مسائل گروه Low می‌باشند. به

جدول ۱. تفاوت‌های بین دو گروه مسئله تولید شده [۱۶]

پارامتر تاثیرگذار	گروه High	گروه Low
r_j	کارها در یک بازه زمانی بزرگ که طولانی‌تر از زمان شروع برنامه‌ریزی شده می‌باشد در دسترس قرار می‌گیرد.	کارها در یک بازه زمانی کوچک و نزدیک به زمان شروع برنامه‌ریزی شده در دسترس قرار می‌گیرند.
d_j	بازه زمان‌های تحویل باز	بازه زمان‌های تحویل فشرده
d_j	زمان‌های تحویل متفاوت	زمان‌های تحویل مشابه
تعداد کارهای دارای دیرکرد	تعداد کارهای دارای دیرکرد بیشتر	تعداد کارهای دارای دیرکرد کمتر

شده‌اند. بولفین و هلاه [۱۵] جهت بررسی عملکرد الگوریتم ابتکاری از شاخص $Z^{\text{opt}} / Z^{\text{heu}}$ استفاده نمودند که در آن Z^{opt} به ترتیب جواب بهینه و جواب الگوریتم ابتکاری با هدف $(\sum_{j=1}^n U_j)$ می‌باشند. با توجه به این که حداقل مقدار این شاخص در تمام دسته‌ها برابر با ۱ می‌باشد بنابراین در ستون "انحراف جواب الگوریتم ابتکاری از جواب بهینه" مقادیر متوسط و حداقل شاخص ارائه شده است.

همچنین برای بررسی کارائی روش‌های ارائه شده مسائلی با تعداد کار ۸ تا ۳۴ و از هر مسئله ۲۰ نمونه تولید شده‌اند و در نهایت در این تحقیق ۳۴۰ نمونه مسئله حل و برای هر مسئله محدودیت زمانی برابر ۳۶۰۰ ثانیه در نظر گرفته شده است و در این مدت زمان اگر جواب بهینه توسط روش شاخه و کران به دست نیاید روند اجرای الگوریتم شاخه و کران قطع می‌شود.

جدول ۲ و ۳ نتایج هر دو گروه را نشان می‌دهد. در جدول ۲ مسائل در گروه High و در جدول ۳ مسائل برای گروه Low حل

جدول ۲. نتایج برای الگوریتم شاخه و کران (B&B) و ابتکاری - گروه High

تعداد کارها	زمان‌های اجرا (ثانیه)	تعداد گره‌های طی شده						تعداد مسائل حل نشده توسعه	تعداد مسائل حل نشده		
		تعداد گره‌های طی شده			B&B	حداکثر	متوسط				
		حداکثر	متوسط	*							
۸	۰,۰۰	۱۰	۴۰	۱۰۰	.			۱,۳۳	۱,۱		
۱۲	۰,۰۹	۱۵	۵۲۲	۶۶۶۱	.			۱,۲۵	۱,۰۶		
۱۶	۰,۱۲	۱,۶۴	۴۳	۸۶۱۹۹	۰,۷۵۵	۵۷۵۵		۱,۱۸	۱,۰۹		
۲۰	۰,۸۵	۱۳,۴۸	۲۱	۵۸۲۵۵۸	۳۶۲۲۳۸			۱,۱۳	۱,۰۸		
۲۴	۰,۰۰	۱۸۶,۲۵	۴۹	۶۴۳۴۲۷۸	۳۲۶۱۹۹			۱,۲۲	۱,۰۹		
۲۶	۰,۰۳	۲۶۰۰,۵۲	۵۴۷	۷۹۰۵۲۹۶۹	۱۱۴۰۴۰۲۹			۱,۲۸	۱,۱۱		
۲۸	۰,۰۱	۲۵۰۷,۰۲	۴۶۶	۸۷۸۲۵۹۴۱	۱۱۷۶۱۰۱۹			۱,۱۸	۱,۰۸		
۳۰	۰,۰۳	۳۱۸۲,۰۶	۱۲۶۶	۱۰۳۵۴۲۲۹۰	۶۳۱۸۹۲۰			۱,۲۲	۱,۱		
۳۲	۰,۰۵	۱۵۲,۰۱	۹۲۱	۳۳۴۰۶۹۳۴	۴۶۲۱۴۲۱			۱,۱۲	۱,۰۸		
۳۴	۰,۰۳	۲۳۰۹,۳۷	۷۴۲	۵۴۴۸۶۶۵۷	۱۵۶۰۴۴۴۱			۱,۱۸	۱,۰۷		

جدول ۳. نتایج برای الگوریتم شاخه و کران (B&B) و ابتکاری - گروه Low

تعداد کارها	حداقل متوسط	زمان های اجرا (ثانیه)	تعداد گره های طی شده				تعداد حداکثر	حل نشده توسط	B&B	تعداد مسائل	انحراف جواب الگوریتم
			حداکثر	متوسط	حداکثر	متوسط					ابتكاري از جواب بهينه
۸	۰.	۰,۰۲	۱۱	۱۶۲	۵۱۶	۵۹۱۰	۱۳	۱۴۰۸	۰.	۱,۰۸	حداکثر
۱۲	۰.	۰,۰۳	۱۳	۱۴۰۸	۴۲۰۸۱۳۱	۳۹۳۹۲۸	۳۱	۹۳,۱۹	۰.	۱,۰۷	B&B
۱۶	۰.	۸,۳۹	۹۳,۱۹	۸,۳۹	۱۰۶۴۴۵۹۱۱	۱۰۶۷۳۹۶۵	۲۵	۳۴۵۰,۹۳	۳۰۶,۲۳	۱,۰۶	حل نشده توسط
۲۰	۰.	۰.	۳۶	۱۴۰۰,۴۷	۴۷۱۰۰,۸۸۵	۱۲۰۷۰۳۹۲	۴۲	۳۷۰,۳۶	۳۷۰,۳۶	۱,۰۴	متوسط
۲۴	۰.	۱۷۶,۶۳	۷۹۶,۱۶	۱۷۶,۶۳	۲۷۲۴۱۳۳۷	۶۱۹۹۶۱۷	۴۰	۴۵۱,۰۹	۸۵,۳۱	۱,۰۷	حداکثر
۲۶	۰.	۰.	۷۹۶,۱۶	۷۹۶,۱۶	۱۴۶۳۲۲۷۵	۲۷۸۴۹۳۸	۱۱	۰,۰۲	۰.	۱,۰۵	ابتكاري از جواب بهينه
۲۸	۰.	۰.	۰.	۰.	۰.	۰.	۰.	۰.	۰.	۱,۱۳	حداکثر

جدول شماره ۴. نتایج برای الگوریتم ابتکاری -

مسائل گروه High

تعداد کارها	زمان های اجرا (ثانیه)	تعداد	حداکثر	متوسط	حداقل	حداکثر
۲۰۰	۰,۰۸	۰,۰۸	۰,۱۳	۰,۱۳	۰,۱۷	۰,۱۷
۴۰۰	۰,۷۰	۰,۷۰	۱,۰۵	۱,۰۵	۱,۳۳	۱,۳۳
۶۰۰	۲,۳۱	۲,۳۱	۳,۰۳	۳,۰۳	۳,۶۴	۳,۶۴
۸۰۰	۵,۵۳	۵,۵۳	۷,۱۰	۷,۱۰	۸,۶۴	۸,۶۴
۱۰۰۰	۱۱,۱۶	۱۱,۱۶	۱۳,۹۸	۱۳,۹۸	۱۶,۳۷	۱۶,۳۷

جدول شماره ۵. نتایج برای الگوریتم ابتکاری -

مسائل گروه Low

تعداد کارها	زمان های اجرا (ثانیه)	تعداد	حداکثر	متوسط	حداقل	حداکثر
۲۰۰	۰,۲۲	۰,۲۲	۰,۲۵	۰,۲۵	۰,۳۰	۰,۳۰
۴۰۰	۱,۶۴	۱,۶۴	۱,۸۵	۱,۸۵	۲,۰۶	۲,۰۶
۶۰۰	۵,۶۰	۵,۶۰	۶,۱۳	۶,۱۳	۶,۵۵	۶,۵۵
۸۰۰	۱۲,۹۷	۱۲,۹۷	۱۴,۴۶	۱۴,۴۶	۱۹,۴۲	۱۹,۴۲
۱۰۰۰	۲۵,۵۵	۲۵,۵۵	۳۲,۰۸	۳۲,۰۸	۴۳,۴۶	۴۳,۴۶

۵. نتیجه‌گیری و پیشنهادات

در این مقاله مسئله $P2/r_j \sum U_j$ یعنی زمان‌بندی فلوشاپ جایگشتی دو ماشین با زمان‌های ورود غیر همزمان و هدف حداقل کردن تعداد کارهای دارای دیرکرد مورد بررسی قرار گرفت. در ابتدا نشان داده شد که این مسئله NP-hard است. از این رو یک الگوریتم شاخه و کران و یک الگوریتم ابتکاری برای حل آن ارائه شد.

در جداول ۲ و ۳ متوسط شاخص Z^{opt} / Z^{heu} حداکثر ۱/۱۱ برابر می‌باشد و در مقایسه با تحقیقات دیگر مربوط به تعداد کارهای دیرکرددار [۲۲]، عدد کوچکی است که نشان دهنده‌ی کارآئی بالای الگوریتم ابتکاری است.

نتایج نشان می‌دهد که الگوریتم شاخه و کران پیشنهادی قادر است مسائلی را تا اندازه ۲۸ کار در گروه High و ۲۰ کار در گروه Low را بطور کامل در زمان منطقی حل نماید و نیز الگوریتم ابتکاری در اندازه‌های کوچک قادر است تا اکثر مسائل را به طور بهینه حل نماید و در اندازه‌های متوسط با تقریب مناسبی می‌تواند مسائل را حل نماید.

علاوه بر این، نتایج نشان می‌دهد که با بزرگ شدن بازه j_1 و j_2 زمان حل و تعداد گره‌های حل شده در روش شاخه و کران کاهش می‌یابد. از آنجا که مسئله حداقل کردن تعداد کارهای دارای دیرکرد بر روی فلوشاپ دوماшиین تا کنون با هیچ فرضی با روش‌های بهینه حل نشده است برای مقایسه نتایج می‌توان به کار حریری و پتس اشاره کرد که آنها همین مسئله را برروی یک فلوشاپ M ماشین و بدون فرض دسترسی حل کردند. آنها توانستند برای فلوشاپ با ۲ ماشین تعدادی از مسائل با ۲۵ کار و برای فلوشاپ با ۳ ماشین تعداد اندکی از مسائل با ۲۵ کار را حل کنند.

در جداول ۴ و ۵ نتایج اجرای الگوریتم ابتکاری برای مسائل با اندازه بزرگ ارائه شده است. در هر مسئله تعداد ۱۰۰ نمونه اجرا شده است.

نتایج نشان می‌دهد که الگوریتم ابتکاری پیشنهادی قادر به حل مسائل بزرگ در زمانی کمتر از ۱۴ ثانیه برای مسائل گروه High و ۳۲ ثانیه برای مسائل گروه Low می‌باشد.

- [8] Hariri, A.M.A., Potts, C.N., "A Branch and Bound Algorithm to Minimize Number of Late Jobs in a Permutation Flow Shop", European Journal of Operational Research, Vol.38, 1989, pp.228–237.
- [9] Moslehi, G., Mirzaee, M., Vasei, M., Modarres, M., Azaron, A., "Two-Machine Flow Shop Scheduling to Minimize the Sum of Maximum Earliness and Tardiness", Int. J. Production Economics, Vol.122, 2009, pp.763–773.
- [10] Wan, G., Yen, B.P.C., "Single Machine Scheduling to Minimize Total Weighted Earliness Subject to Minimal Number of Tardy Jobs", European Journal of Operational Research, Vol.195, 2009, pp. 89–97.
- [11] Baptiste, P., Peridy, L., Pinson, E., "A Branch and Bound to Minimize the Number of Late Jobs on a Single Machine with Release Time Constraints", European Journal of Operational Research, Vol.144, 2003, pp.1–11.
- [12] Sadykov, R., "A Branch-and-Check Algorithm for minimizing the Weighted Number of Late Jobs on a Single Machine with Release Dates", European Journal of Operational Research, Vol. 189, 2008, pp.1284–1304.
- [13] Hallah, R., Bulfin, R.L., "Minimizing the Weighted Number of Tardy Jobs on a Single Machine with Release Dates", European Journal of Operational Research, Vol. 176, 2007, pp.727–744.
- [14] Hallah, R., Bulfin, R.L., "Minimizing the Weighted Number of Tardy Jobs on Parallel Processors", European Journal of Operational Research, Vol. 160, 2005, pp.471–484.
- [15] He, C., Lin, Y., Yuan, J., "A Note on the Single Machine Scheduling to Minimize the Number of Tardy Jobs with Deadlines", European Journal of Operational Research, Vol.201, 2010, pp.966–970.
- [16] Lodree, E., Jang, W., Klein, C.M., "A New Rule for Minimizing the Number of Tardy Jobs in Dynamic flow Shops", European Journal of Operational Research, Vol.159, 2004, pp.258–263.
- [17] Ruiz-Torresa, A.J., Centeno, G., "Minimizing the Number of Late Jobs for the Permutation Flowshop Problem with Secondary Resources", Computers & Operations Research, Vol.35, 2008, pp.1227–1249.
- [18] Ruiz-Torresa, A.J., Ablanedo-Rosas, J.H., Ho, J.C., "Minimizing the Number of Tardy Jobs in the flowshop Problem with Operation and Resource Flexibility", Computers & Operations Research, Vol. 37, 2010, pp.282–291.
- [19] Choi, S.W., Kim, Y.D., "Minimizing Total Tardiness on a Two Machine Re-entrant Flowshop" European

در روش شاخه و کران سه حد پایین و سه اصل غلبه ارائه گردید. کارآیی و عملکرد الگوریتم با حل تعداد ۳۴۰ مسئله نمونه مورد بررسی قرار گرفت. نتایج محاسباتی نشان داد که الگوریتم شاخه و کران مسائل با ابعاد ۲۸ فعالیت در مسائل گروه High و ۲۰ فعالیت در مسائل گروه Low را بطور کامل حل نموده است که این امر کارآیی رویه ارایه شده را نشان می‌دهد. علاوه بر آن نشان داده شد که متوسط نسبت جواب بهینه به ابتکاری با هدف $\sum(1-U_i)$ حداقل ۱/۱۱ برابر می‌باشد که بیان کنندهی عملکرد بالای الگوریتم ابتکاری است.

از آنجا که الگوریتم ابتکاری معرفی شده از قابلیت بالایی برخوردار است و از این الگوریتم به عنوان حد بالا در روش شاخه و کران استفاده شده است، لذا پیشنهاد می‌شود برای بهبود روش شاخه و کران حدود پایین و اصول غلبه جدیدی ارائه شود. در نظر گرفتن فرض زوال پذیری کارها در مسئله مورد بررسی از مسائل جالب برای ادامه تحقیقات به حساب می‌آید. از آنجا که مسئله مورد بررسی تا کنون با هیچ روشی حل نشده است پیشنهاد می‌شود از الگوریتم‌های فراتکاری برای حل مسئله در ابعاد بزرگ استفاده شود.

مراجع

- [1] Bulfin, R.L., M'Hallah, R., "Minimizing the Weighted Number of Tardy Jobs on a Two-Machine Flow Shop", Computers & Operations Research, Vol.30, 2003, pp.1887–1900.
- [2] Lageweg, B.J., Lenstra, J.K., Rinnooy Kan, A., "A General Bounding Scheme for the Permutation Flowshop Problem", Operations Research, Vol.26, 1978, pp.53–67.
- [3] Ignall, E., Schrage, L.E., "Applications of the Branch and Bound Technique to Some Flow Shop Problems", Operations Research, Vol.13, 1965, pp.400–412.
- [4] Lenstra, J.K., Rinnooy Kan, A., Brucker, P., "Complexity of Machine Scheduling Problems", Annals of Discrete Mathematics, Vol.1, 1968, pp.343–362.
- [5] Moore, J.M., "An n Job, One Machine Algorithm for Minimizing the Number of Late Jobs", Management Science, Vol.15, 1968, pp.102–109.
- [6] Lin, B., "Scheduling in the Two-Machine Flow Shop with Due Date Constraints", International Journal of Production Economics, Vol.70, 2001, pp.117–123.
- [7] Lawler, E.L., Moore J.M. "A Functional Equation and its Application to Resource Allocation and Sequencing Problems", Management Science, Vol. 15, 1968, pp.102–109

Journal of Operational Research, Vol. 199, 2009,
pp.375–384.

- [20] Choi, H.S., Lee, D.H., “*Scheduling Algorithms to Minimize the Number of Tardy Jobs in Two-Stage Hybrid Flow Shops*”, Computers & Industrial Engineering, Vol.56, 2009, pp.113–120.
- [21] Mosheiov, G., Sarig, A., “*Minimum Weighted Number of Tardy Jobs on an M-Machine Flow-shop with a Critical Machine*”, European Journal of Operational Research, Vol.201, 2010, pp.404–408.
- [22] Moslehi, G., Jafari, A., “*Minimizing the Number of Tardy Jobs Under Piecewise-Linear Deterioration*” Computers & Industrial Engineering, Vol.59, 2010, pp. 573-584.