



## Modeling and Analytical Solution of Integrated Scheduling and Capacity Planning Problem: Lower Bounds and Efficient Branch and Bound Algorithm

H. Mokhtari, I. Nakhai Kamal Abadi\* & M.R. Amin-Naseri

*Hadi Mokhtari*, PhD Candidate of Industrial Engineering, Faculty of Engineering, Tarbiat Modares University, Tehran, Iran

*Isa Nakhai Kamal Abadi*, Professor of Industrial Engineering, Faculty of Engineering, Tarbiat Modares University, Tehran, Iran

*Mohammad Reza Amin-Naseri*, Associate Professor of Industrial Engineering, Faculty of Engineering, Tarbiat Modares University, Tehran, Iran

### Keywords

Capacity Planning;  
Order Scheduling;  
Branch and Bound Algorithm;  
Lower Bound;  
Search Strategy

### ABSTRACT

*Production capacity planning and scheduling are parts of decision process in supply chain of many manufacturing and service industries, which have important role in satisfying customer requirements and enhancing customer service level. Since these two processes performs in two different levels of supply chain (capacity planning in tactical level and scheduling in operational one), usually their decision are made independently. In such circumstances, the feasibility of capacity decisions in tactical level, when plan imports into the shop floor, is one of the critical challenges in this decision making approach. Hence in this paper, after formulating the capacity planning and two-machine scheduling problems integrately, an analytical solution approach will be presented. In order to minimize the maximum completion time of orders, concept of order pairs is defined and the optimal orders pairs are established based on the optimal solution of symmetric assignment problem. Afterwards, to determine the optimal sequence of order pairs, a branch and bound procedure is designed which is based on three proposed lower bounds, depth first and best first strategies. To improve the efficiency of algorithm, some mathematical properties are proved and some corresponding dominance rules are derived. Furthermore, to delineate the quantity of resource allocation to the operations, a neighborhood search algorithm is designed and improved based on some structural properties of problem.*

© 2013 IUST Publication, IJIEPM. Vol. 24, No. 2, All Rights Reserved

\*  
Corresponding author. Isa Nakhai Kamalabadi  
Email: [Nakhai@modares.ac.ir](mailto:Nakhai@modares.ac.ir)

## مدل سازی و حل تحلیلی مسأله برنامه ریزی ظرفیت و زمانبندی تولید یکپارچه: استخراج کران پائین و طراحی یک الگوریتم شاخه و کران کارا

هادی مختاری، عیسی نخعی کمال آبادی\* و محمدرضا امین ناصری

### چکیده:

برنامه ریزی ظرفیت تولید و زمانبندی تولید جزئی از فرآیند تصمیم گیری در زنجیره تأمین بسیاری از صنایع ساخت و خدماتی محسوب می شوند که نقش مهمی را در برآورده سازی نیازهای مشتریان و ارتقای سطح خدمت دهی به مشتریان ایفا می نمایند. از آنجائیکه این دو فرآیند در دو سطح متفاوت از زنجیره تأمین عمل می نمایند (برنامه ریزی ظرفیت در سطح تاکتیکی و زمانبندی تولید در سطح عملیاتی)، لذا معمولاً تصمیمات مربوط به این دو حیطة نیز مستقلاً و جدای از هم اتخاذ می شود. در چنین شرایطی شدنی بودن تخصیص ظرفیت انجام شده در سطح تاکتیکی، زمانیکه برنامه بصورت عملیاتی وارد کارگاه می شود، به عنوان یکی از چالش های اصلی این رویکرد تصمیم گیری مجزا، مطرح است. لذا در این تحقیق، پس از مدلسازی یکپارچه مسأله برنامه ریزی ظرفیت تولید و مسأله زمانبندی تولید دو ماشین، به ارائه یک رویکرد حل تحلیلی پرداخته می شود. در راستای کمینه سازی حداکثر زمان تکمیل سفارشات، مفهومی تحت عنوان «زوج سفارش» تعریف، و الگوریتمی جهت تعیین زوج سفارشات بهینه بر مبنای مسأله تخصیص متقارن ارائه می شود. سپس در راستای تعیین توالی بهینه ی زوج سفارشات، الگوریتم شاخه و کرانی بر مبنای سه کران پائین پیشنهادی و همچنین دو استراتژی جستجوی عمق اول و سطح اول، طراحی و در راستای افزایش کارایی آن تعدادی ویژگی ریاضی اثبات، و بر اساس آن ها تعدادی قاعده چیرگی جهت مسأله استخراج شده است. همچنین جهت تعیین مقادیر تخصیص ظرفیت به عملیات، یک الگوریتم جستجوی همسایگی طراحی و بر اساس ویژگی های ساختاری مسأله بهبود داده شد.

### کلمات کلیدی

برنامه ریزی ظرفیت؛  
زمانبندی سفارشات؛  
الگوریتم شاخه و کران؛  
کران پائین؛  
استراتژی جستجو

### ۱. مقدمه

برنامه ریزی ظرفیت تولید یکی از فرآیندهای برنامه ریزی میان مدت در هر واحد صنعتی محسوب می شود. به همان اندازه که

ظرفیت مازاد تولیدی، هزینه های زیادی را در بر خواهد داشت، کمبود ظرفیت جهت برآوردن تقاضای مشتریان جدید و تغییرات پیش بینی نشده ی بازار نیز به مسأله ای اصلی در اقتصاد تولید تبدیل شده است.

ظرفیت قابل تنظیم<sup>۲</sup> و سیاست های تنظیم بهینه ظرفیت می تواند بصورتی هماهنگ عمل نمایند که منجر به افزایش پایداری و ثبات تولید و فروش در یک بنگاه تولیدی شوند (Asl-Farshid Galip Ulsoy 2002).

امروزه برنامه ریزی ظرفیت تولید، بدلیل افزایش مؤثر نقش مشتریان در اقتصاد تولید و همچنین تغییرپذیری بالای تقاضای

تاریخ وصول: ۹۰/۶/۱۶

تاریخ تصویب: ۹۰/۹/۳۰

هادی مختاری، دانشجوی دوره دکتری تخصصی مهندسی صنایع، دانشگاه تربیت مدرس

\*نویسنده مسئول مقاله: دکتر عیسی نخعی کمال آبادی، استاد دانشکده مهندسی صنایع دانشگاه تربیت مدرس، [Nakhai@modares.ac.ir](mailto:Nakhai@modares.ac.ir)

دکتر محمدرضا امین ناصری، دانشیار دانشکده مهندسی صنایع دانشگاه تربیت مدرس

<sup>2</sup> Reconfigurable Capacity

برقراری محدودیت عدم انتظار در این مسأله، به محض اینکه اولین فرآیند هر کار شروع شود، باید کلیه فرآیندهای آن بدون هیچ انتظاری بین ماشین‌ها و هیچ انقطاعی حین فرآیندها، به تمام برسد. در واقع مسأله‌ی زمانبندی در حالت عدم انتظار، حالت خاصی از مسأله‌ی عمومی زمانبندی است که در آن هیچ زمان انتظار و انقطاعی برای کارها مجاز نیست. بعبارت دیگر تفاوت بین زمان تکمیل همه‌ی کارها و زمان شروع آن‌ها باید دقیقاً برابر با مجموع زمان کلیه فرآیندهای آن باشد. در ادبیات، عموماً مسائلی زمانبندی در حالت عدم انتظار با تابع هدف  $C_{max}$  در نظر گرفته شده است و همچنین نشان داده شده است که این مسأله حتی با دو ماشین، یک مسأله‌ی قویاً NP-Hard است.

Hall و Srisankandarajah (۱۹۹۶) ادبیات مسائل در حالت عدم انتظار را مرور کرده و همچنین نشان دادند که مسأله‌ی فوق مخصوصاً برای ابعاد بزرگ مسأله، جزء مسائل سخت بهینه‌سازی محسوب می‌شود. ادبیات مربوط به این مسأله قابل تقسیم به دو حیطه است:

(۱) پیچیدگی محاسباتی مسأله و (۲) رویکردهای حل مسأله. بر اساس اطلاعات نویسنده‌ی این تحقیق، فقط تعداد محدودی تحقیق، مرتبط با روش‌های حل برای این مسأله ارائه شده است. در مورد روش‌های دقیق حل، تنها یک الگوریتم شاخه و کران در ادبیات توسعه داده شده است (Mascis and 2002). علاوه بر این چندین روش فراابتکاری مانند شبیه‌سازی تبرید (Pacciarelli, Raaymakers and Hoogeveen 2000)، جستجوی ممنوعه (Macchiarioli et al. 1996, Schuster 2006)، جستجوی همسایگی متغیر (Schuster and Framinan 2006)، الگوریتم ژنتیک (Brizuela et al. 2001)، جستجوی محلی (Zhu et al. 2009)، ترکیب الگوریتم ژنتیک و گدازش شبیه‌سازی شده (Wang and Zheng 2001)، جستجوی همسایگی (Framinan and Schuster 2006) و الگوریتم ژنتیک ترکیبی (Pan and Huang 2009, Mokhtari et al. 2011) برای مسأله‌ی موضوع این تحقیق ارائه شده است. همچنین Reddi و Ramamoorthy (۱۹۷۳) با ارائه‌ی یک رویکرد تحلیلی مبتنی بر حد پائین، الگوریتم شاخه و کرانی را جهت مسأله زمانبندی تولید در حالت دو مرحله‌ای توسعه دادند. در مدل‌های زمانبندی کلاسیک، فرض می‌شود میزان ظرفیت تخصیص داده شده به کارها پارامتر ثابت و از قبل تعیین‌شده‌ای است (Shabtay and Steiner 2007).

در حالیکه در بسیاری از موارد، زمان‌های پردازش با تخصیص منابع بیشتر، قابل تغییر هستند. نتیجه‌ی رویکرد یکپارچه به دو مسأله از دو سطح برنامه‌ریزی و زمانبندی از زنجیره تأمین نیز به

بازار اهمیت زیادی پیدا کرده است. شرکت‌های تولیدی در راستای بقای خود در بازار رقابتی امروز و امکان رقابت در مقیاس جهانی، نیاز به نوع جدیدی از سیستم‌های تولیدی دارند که دارای قابلیت عکس‌العمل سریع به تقاضای بازار جهانی باشد. چنین سیستم تولیدی می‌بایست دارای ظرفیت تولید کاملاً منعطف در پاسخ-گویی به تغییرات بازار باشد.

یک سیستم تولید تنظیم‌پذیر شامل ماشین‌ها، تکنولوژی‌ها، فرآیندها، منابع تولیدی و نیروی انسانی کاملاً منعطف و تنظیم-پذیر بوده و منجر می‌شود که کل سیستم تولیدی در پاسخ به تغییرات بازار و با کمترین هزینه ظرفیت خود را متناسب با بازار تغییر داده و قادر به پاسخگویی سریع نیازمندی‌های مشتریان باشد. تحقق این هماهنگی نیاز به رویکرد یکپارچه به دو مقوله‌ی برنامه‌ریزی و زمانبندی در دو سطح تاکتیکی و عملیاتی از زنجیره تأمین دارد.

زمانبندی فرآیندی است که در آن تخصیص منابع به منظور انجام مجموعه‌ای از کارها در یک دوره‌ی زمانی انجام می‌گیرد (Baker, 1974). نوعاً دو دسته‌ی کلی از توابع هدف در مسائل زمانبندی قابل بکارگیری است (Baker, 1974). دسته‌ی اول شامل توابع هدف مبتنی بر زمان تکمیل<sup>۱</sup> هستند که در آن‌ها زمان مورد نیاز جهت تکمیل کارها مورد ارزیابی قرار می‌گیرد، در حالیکه دسته‌ی دوم توابع هدف مبتنی بر موعد تحویل<sup>۲</sup> هستند که در این دسته معیار ارزیابی یک زمانبندی، عملکرد آن زمانبندی در برآورده-سازی موعد مقرر تحویل کار به مشتری است.

علاوه بر نوع توابع هدف، عوامل مختلف دیگری نیز همچون پیکربندی منابع تولیدی (ماشین‌آلات، وسائل حمل و نقل، نیروی انسانی و غیره) و ماهیت کارها (سفارشات) در دسته‌بندی مسائل زمانبندی دخیل هستند. یکی از مدل‌های پرکاربرد زمانبندی، مدل‌های زمانبندی در شرایط عدم انتظار<sup>۳</sup> است که مربوط به نحوه‌ی انجام کارها است.

مسأله‌ی زمانبندی در حالت عدم انتظار یک مسأله‌ی زمانبندی را معرفی می‌نماید که در آن یک محدودیت جدید به مسأله‌ی عمومی زمانبندی اضافه شده است (Schuster and Framinan 2003). این محدودیت جدید که در صنایعی همانند صنعت چاپ و نشر مشاهده می‌شود، بیان می‌نماید که هیچ کدام از کارها نمی‌توانند هیچ زمان انتظاری را تحمل کنند مگر روی ماشین اول. یک مجموعه از  $n$  کار  $(J_1, J_2, \dots, J_n)$  و  $m$  ماشین  $(M_1, M_2, \dots, M_m)$  داده شده است، به نحوی که هر کار باید یک مسیر خاص و از پیش تعیین شده‌ای را از میان ماشین‌ها طی نماید. به منظور

<sup>1</sup> Completion Time

<sup>2</sup> Delivery Time

<sup>3</sup> No-wait Jobs Shop

منجر به یک مسأله‌ی تخصیص منابع در فرآیند برنامه‌ریزی ظرفیت و در نهایت تشکیل تابعی از حجم منابع تخصیص داده شده برای زمان‌های پردازش کارها است.

در بسیاری از سیستم‌های تولید واقعی، زمان‌های پردازش را می‌توان از طریق تخصیص منابع بیشتر مانند منابع مالی، انرژی، سوخت، نیروی انسانی و غیره به عملیات، کنترل کرد ( Tseng et al. 2009). اما اکثر مسائل زمانبندی بطور سنتی، چنین فرآیندی را که نتیجه تصمیمات همزمان در دو لایه‌ی برنامه‌ریزی و زمانبندی است، از سیستم تصمیم‌گیری تولید، مجزا نموده و تنها به تصمیمات توالی کارها می‌پردازند. همانطوریکه بحث شد، در ادبیات، مسأله زمانبندی تولید و برنامه‌ریزی ظرفیت تولید که در دو سطح مختلف از تصمیمات زنجیره تأمین مطرح هستند، معمولاً بصورت مستقل و مجزا مدل‌سازی شده و حل می‌شوند، در حالیکه این دو مسأله کاملاً به هم وابسته بوده و ورودی یکی به عنوان خروجی دیگری و خروجی آن، ورودی اولی است. معمولاً در بهترین حالت، مسأله زمانبندی تولید و برنامه‌ریزی ظرفیت بصورت رفت و برگشتی در واقعیت لحاظ می‌شوند ( Maravelias and Sung 2009).

به عبارت دیگر مسأله برنامه‌ریزی ظرفیت یک بار انجام شده و بر اساس نتایج آن، طرح عملیاتی مربوط به زمانبندی تولید پیگیری می‌شود. در صورت وجود مشکلاتی همچون امکان‌پذیر نبودن برنامه‌ریزی ظرفیت انجام شده در مرحله زمانبندی و یا اختصاص ظرفیت بیش از حد مورد نیاز در مرحله برنامه‌ریزی ظرفیت، این موضوع مجدداً بر اساس خروجی فاز زمانبندی تولید به برنامه‌ریزی اعلام می‌شود تا برنامه‌ریزی مجدد انجام شده و خروجی آن جهت زمانبندی مجدد استفاده شود.

همانطوریکه قابل استنباط است، این رویکرد با وجود اینکه منجر به برنامه‌ریزی مناسب و اثر بخش<sup>۱</sup> ظرفیت تولید و همچنین زمانبندی تولید می‌شود ولی به دلیل رفت و برگشت‌های زیاد و انجام محاسبات طولانی در هر بار، کارایی<sup>۲</sup> مناسبی ندارد. اهمیت مدل‌های یکپارچه برنامه‌ریزی ظرفیت تولید و زمانبندی تولید به این لحاظ است که در یک مدل منسجم و برای یک بار پارامترهای برنامه‌ریزی و زمانبندی بطور همزمان در یک فرآیند تصمیم‌گیری لحاظ شده و در نهایت تصمیمات سازگاری در حیطه برنامه‌ریزی و زمانبندی اتخاذ می‌شود.

در اکثر مسائل زمانبندی تولید که در ادبیات ارائه شده‌اند، تنها با در نظر گرفتن تصمیمات مربوط به سطح عملیاتی، و بودن برنامه‌ریزی ظرفیت تولیدی، میزان ظرفیت ماشین‌آلات (ایستگاه‌های کاری) جهت پردازش کارها ثابت فرض شده است ( Kim et al.

چرا که در این مدل‌ها، برنامه‌ریزی تولید که از تصمیمات سطح تاکتیکی از زنجیره است، بصورت مستقل انجام شده و نتایج آن به عنوان ظرفیت ماشین‌آلات به عنوان ورودی به مدل زمانبندی، در نظر گرفته شده است. مسائل زمانبندی کارها و تخصیص منابع می‌بایست بطور کاملاً هماهنگی انجام شود تا به خروجی‌های مؤثرتری دست یافت.

یکی از کاربرد مسأله برنامه‌ریزی ظرفیت و زمانبندی یکپارچه در صنعت فولاد است (Janiak 1989)، جایی که دسته‌ای از شمش‌های فولاد می‌بایست قبل از ورود به فرآیند غلطک داغ، بطور مقدماتی گرم شوند و در این شرایط هم زمان گرمایش مقدماتی و هم زمان غلطک زنی بصورت معکوس مرتبط با چگالی جریان گاز تغذیه هستند).

همچنین کاربرد دیگری از این مسائل را می‌توان در صنایع ساخت و تولید نام برد که در آن زمان پردازش یک فرآیند ماشین‌کاری تابعی از نرخ تغذیه و سرعت ابزار است (Kayan and Akturk, 2005). صنایعی که در بازارهای با رقابت بالا مشغول فعالیت هستند، همچون صنعت چاپ نیز از دیگر نمونه‌های کاربرد مسائل یکپارچه برنامه‌ریزی ظرفیت و زمانبندی تولید است که در آن‌ها یکی از راهکارهای عدم از دست دادن مشتری در شرایطی که تقاضای بازار مازاد بر ظرفیت تولید باشد، برنامه‌ریزی یکپارچه ظرفیت تولی و تنظیم دقیق آن بصورت هماهنگ با زمانبندی تولید است. جهت تحلیل دقیق‌تر، بر اساس پارامترهایی همچون محیط تولیدی، رفتار تابع ظرفیت، نوع منبع تولیدی، معیار زمانبندی، تابع (توابع) هدف، به دسته‌بندی مسائل یکپارچه برنامه‌ریزی ظرفیت و زمانبندی تولید بر اساس جدول ۱ می‌پردازیم.

همانطوریکه در جدول مشخص است، اکثر تحقیقات در زمینه رویکردهای یکپارچه به دو مقوله برنامه‌ریزی ظرفیت و زمانبندی تولید در محیط‌های تولید کارگاهی، با منابع محدود ناپذیر و یک منبع ارائه شده‌اند.

بر اساس کاربردهای از ادبیات که مطرح شد و همچنین خلأ تحقیقاتی موجود در ادبیات، اهمیت رویکرد یکپارچه به مسائل برنامه‌ریزی ظرفیت و زمانبندی مشخص می‌شود. لذا در این تحقیق به بررسی مدلی از این مسأله در شرایط عدم انتظار که از جمله اقتضات تولید در صنایعی استراتژیک همچون فولاد است، پرداخته می‌شود. بر اساس بهترین اطلاعات ما، مدلی که چنین مسأله‌ای را در محیط تولید کارگاهی بررسی نماید، ارائه نشده است. لذا مدل ارائه شده در فضای تولید کارگاهی است که پس از مدل‌سازی ریاضی مسأله به ارائه رویکرد حلی مبتنی بر الگوریتم جستجوی همسایگی و رویکرد شاخه و کران پرداخته می‌شود.

<sup>1</sup> Effective

<sup>2</sup> Efficient

جدول ۱. دسته بندی ادبیات مسائل یکپارچه برنامه ریزی ظرفیت و زمان بندی تولید

ردیف	مقاله (نویسندگان)	محیط تولیدی	رفتار تابع ظرفیت تولید	نوع منبع	تعداد منبع	معیار زمان بندی	تابع هدف
۱	Yedidsion et al. 2007	تک ماشین	تابع محدب	تجدید ناپذیر	تک منبع	Lmax	چند هدفه با معیار زمان بندی و هزینه منابع
۲	Yedidsion et al. 2009	تک ماشین	تابع محدب	تجدید ناپذیر	تک منبع	تعداد کارهای با تأخیر	چند هدفه با معیار زمان بندی و هزینه منابع
۳	Xu et al. In Press	تک ماشین	تابع محدب	تجدید ناپذیر	تک منبع	تحقق موعد مقرر تکمیل	تک هدفه با هزینه منابع
۴	Koulamas et al. 2010	تک ماشین	تابع خطی و محدب	تجدید ناپذیر	تک منبع	مجموع زمان تکمیل کارها	تک هدفه با معیار زمان بندی و منابع
۵	Gupta et al. 2002	تولید جریانی	تابع خطی	تجدید ناپذیر	تک منبع	زود کرد + دیر کرد	تک هدفه با معیار زمان بندی
۶	Grigoriev et al. 2007	ماشین موازی	تابع گسسته چند حالت	تجدید پذیر	تک منبع	Cmax	تک هدفه با معیار زمان بندی
۷	Chuang et al. 2009	ماشین موازی	تابع گسسته چند حالت	تجدید پذیر	تک منبع	مجموع زمان های دیر کرد	تک هدفه با معیار زمان بندی
۸	Ventura and Kim, 2003	ماشین موازی	تابع خطی	تجدید پذیر	تک منبعی / چندمنبعی	زود کرد + دیر کرد	تک هدفه با معیار زمان بندی
۹	Shabtay and Kaspi, 2006	ماشین موازی	تابع محدب	تجدید ناپذیر	تک منبع	Cmax	تک هدفه با معیار زمان بندی
۱۰	Choi et al. 2010	تک ماشین	تابع خطی	تجدید ناپذیر	تک منبع	مجموع وزنی زمان تکمیل کارها	تک هدفه با معیار زمان بندی
۱۱	Tseng et al. 2009	تک ماشین	تابع خطی	تجدید ناپذیر	تک منبع	مجموع زمان تأخیر کارها	تک هدفه با معیار زمان بندی و هزینه منابع
۱۲	Cheng et al. 2006a	تک ماشین	تابع خطی	تجدید ناپذیر	تک منبع	Cmax	تک هدفه با معیار زمان بندی و هزینه منابع
۱۳	Cheng et al. 2006b	تک ماشین	تابع خطی	تجدید ناپذیر	تک منبع	مجموع زمان تکمیل کارها	تک هدفه با معیار زمان بندی و هزینه منابع
۱۴	Wang and Xia, 2007	تک ماشین	تابع خطی	تجدید ناپذیر	تک منبع	مجموع زمان تکمیل کارها، مجموع زمان انتظار	تک هدفه با معیار زمان بندی و هزینه منابع
۱۵	Wang, 2006	تک ماشین	تابع خطی	تجدید ناپذیر	تک منبع	مجموع زمان تأخیر کارها	تک هدفه با معیار زمان بندی و هزینه منابع

زمان بندی سفارشات، و برنامه ریزی ظرفیت توصیف می شود. مفهوم زوج سفارشات و نحوه محاسبه زوج سفارشات بهینه بحث شده و سه کران پائین جهت زمان بیکاری ماشین ها ارائه می شود. سپس با بکارگیری کران های پائین، الگوریتم شاخه و کرانی جهت تعیین زمان بندی بهینه ارائه توالی بهینه طراحی می شود. در نهایت در بخش (۷) الگوریتم جستجوی همسایگی جهت تعیین تخصیص ظرفیت به سفارشات، ارائه می شود. در نهایت یک مثال عددی و نتایج محاسباتی ارزیابی الگوریتم های پیشنهادی در بخش (۸) و نتایج حاصل از تحقیق در بخش آخر ارائه می شود.

## ۲. تعریف مسأله تحقیق و مدل سازی ریاضی

یک تعریف کلی از مسأله یکپارچگی برنامه ریزی ظرفیت و زمان بندی تولید، را می توان بصورت زیر ارائه نمود: به تعداد  $n$  کار مختلف  $I = \{J_1, J_2, \dots, J_n\}$  می بایست روی  $m$  ماشین (ایستگاه کاری) پردازش شوند  $M = \{M_1, M_2, \dots, M_m\}$ ، بطوریکه  $O_{ij}$  فرآیند  $j$ -ام کار (سفارش)  $i$ -ام را نمایش می دهد. زمان در دسترس بودن  $T_i$  مقطع زمانی را که کارها وارد می شوند را نمایش می دهد. زمان

همانطوریکه در جدول مشخص است، اکثر تحقیقات در زمینه رویکردهای یکپارچه به دو مقوله برنامه ریزی ظرفیت و زمان بندی تولید در محیط های تولید کارگاهی، با منابع محدود ناپذیر و یک منبع ارائه شده اند. بر اساس کاربردهای از ادبیات که مطرح شد و همچنین خلأ تحقیقاتی موجود در ادبیات، اهمیت رویکرد یکپارچه به مسائل برنامه ریزی ظرفیت و زمان بندی مشخص می شود. لذا در این تحقیق به بررسی مدلی از این مسأله در شرایط عدم انتظار که از جمله اقتضات تولید در صنایعی استراتژیک همچون فولاد است، پرداخته می شود. بر اساس بهترین اطلاعات ما، مدلی که چنین مسأله ای را در محیط تولید کارگاهی بررسی نماید، ارائه نشده است. لذا مدل ارائه شده در فضای تولید کارگاهی است که پس از مدل سازی ریاضی مسأله به ارائه رویکرد حلی مبتنی بر الگوریتم جستجوی همسایگی و رویکرد شاخه و کران پرداخته می شود.

ادامه این مقاله به این شیوه ارائه می شود. در بخش (۲) مدلی یکپارچه از دو مقوله برنامه ریزی ظرفیت و زمان بندی تولید، ارائه می شود. سپس رویکرد حل مبتنی بر تجزیه مسأله در دو فاز

(۳) کمینه سازی میزان مصرف منابع تحت محدودیت حداکثر

مقدار معیار زمانبندی

(۴) موازنه بین میزان مصرف منابع و معیار زمانبندی (رویکرد

چند هدفه)

بر این اساس، مفروضات مسأله پیشنهادی بصورت زیر لحاظ می شود.

- جریان حرکت سفارشات بصورت، تولید کارگاهی<sup>۷</sup> است.
  - قطع شدگی<sup>۸</sup> کارها مجاز نیست.
  - زمان انجام کارها مستقل از توالی کارهاست.
  - ماشینها (ایستگاههای کاری) مشابه نبوده و بیکاری آنها مجاز است.
  - لغو کارها مجاز نیست.
  - هر ماشینی در هر زمان تنها می تواند به یک کار خدمتدهی نماید.
  - ماشینها در تمام افق برنامه ریزی در دسترس هستند.
  - پارامترهای مدل قطعی هستند.
  - سایر تنضیمات زنجیره مانند تعداد تسهیلات، و پیکربندی زنجیره تأمین از قبل انجام شده اند.
  - زمان انجام کارها قطعی است.
  - زمان انجام کارها تابعی گسسته (دارای دو حالت حدی) از منابع تخصیص داده شده است.
  - یک نوع منبع تجدید ناپذیر در اختیار است.
  - انتظار مابین دو عملیات از یک سفارش مجاز نیست.
- بر اساس مفروضات و تعاریف ارائه شده در بالا، در این قسمت به مدلسازی زمانهای تولید وابسته به منابع، که نتیجه رویکرد یکپارچه به فرآیند برنامه ریزی ظرفیت تولید و زمانبندی تولید است می پردازیم. به این منظور، ابتدا پارامترها را تعریف نموده و سپس مدلسازی ریاضی آن را ارائه می دهیم.

#### مجموعه ها و پارامترها:

$i$	شماره سفارش
$j$	شماره ماشین
$I$	مجموعه سفارشات دریافت شده در ابتدای دوره برنامه ریزی
$J$	مجموعه ماشینها
$n$	تعداد سفارشات دریافت شده در ابتدای دوره برنامه ریزی
$m$	تعداد ماشینها
$O_i$	سفارش شماره $i$ ام
$d_i$	موعد تحویل سفارش $O_i$
$O_{ij}$	آزمین عملیات سفارش $O_i$

پردازش برای هر عملیات از کارها، به عنوان تابعی از میزان منابع تخصیص داده شده به آن عملیات روی ماشین خاص و به عبارتی تابعی از ظرفیت ماشین آلات در نظر گرفته می شود. منابع تخصیص داده شده در فاز برنامه ریزی (سطح تاکتیکی از زنجیره) می تواند بصورت گسسته و یا پیوسته باشد. در حالت پیوسته، زمان پردازش کارها از طریق مقداری از منابع قابل تقسیم همانند سوخت گاز و الکتریسیته که به آن فرآیند تخصیص داده شده است، تعیین می شود و در نتیجه ظرفیت تولید می تواند بصورت پیوسته تغییر نماید. در حالتیکه منابع گسسته همانند نیروی انسانی و تجهیزات، غیر قابل تقسیم بوده و به همین جهت زمان پردازش کارها فقط در مقادیر مشخص و محدودی قابل تغییر هستند. همچنین دسته بندی دیگری روی منابع مورد استفاده می توان انجام داد که به در دسترس بودن منابع در دوره برنامه ریزی بستگی دارد. از طرف دیگر منابع تجدید پذیر<sup>۱</sup> به منابعی گفته می شود که در کل دوره برنامه ریزی در دسترس هستند همانند نیروی انسانی در صورتیکه منابع تجدید ناپذیر<sup>۲</sup> در کل دوره در دسترس نبوده و پس از مصرف دیگر در دسترس نیستند مثل سوخت، منابع مالی، الکتریسیته و غیره. تعداد کمی از تحقیقات در زمینه منابع تجدید پذیر در زمانبندی تولید در ادبیات ارائه شده اند. در این زمینه می توان به تحقیقات ارائه شده توسط Jozefowska و Weglarz (۲۰۰۴) و Blazewicz و همکاران (۲۰۰۴) اشاره نمود. اکثر تحقیقاتی که در این زمینه انجام شده است، در محیط های تک ماشینه<sup>۳</sup> به بررسی مسأله پرداخته اند که زمینه تحقیقاتی بالقوه برای مدل های یکپارچه برنامه ریزی ظرفیت ماشین آلات و زمانبندی تولید در محیط هایی همچون تولید گارگاهی<sup>۴</sup>، تولید جریانی<sup>۵</sup> و ماشین های موازی<sup>۶</sup> را نشان می دهد. مسأله برنامه ریزی ظرفیت و زمانبندی یکپارچه را با توابع هدف مختلفی می توان در نظر گرفت. اساساً زمانبندی و برنامه ریزی با زمان های پردازش قابل کنترل، یک رویکرد دو هدفه است که در آن تعادلی بین یک یا چند معیار زمانبندی از یک طرف و میزان منابع مورد نیاز از طرف دیگر صورت می گیرد. با این تعریف، می توان چهار رویکرد زیر را برای تابع هدف این مسأله در نظر گرفت (Yedidsion et al., 2007):

- (۱) کمینه سازی تابع هزینه شامل هزینه معیار زمانبندی و هزینه منابع مصرفی
- (۲) کمینه سازی معیار زمانبندی برای مقدار محدودی از منابع موجود

<sup>1</sup> Renewable  
<sup>2</sup> Nonrenewable  
<sup>3</sup> Single Machine  
<sup>4</sup> Job Shop  
<sup>5</sup> Flow Shop  
<sup>6</sup> Parallel Machine

<sup>7</sup> Job Shop  
<sup>8</sup> Preemption

در این مدل، تابع هدف (۱)، کمینه‌سازی حداکثر زمان تکمیل سفارشات است. محدودیت سری (۱-۱) تضمین می‌نماید که در هر زمان تنها یک سفارش روی ماشین‌ها در حال پردازش باشد. به عبارت دیگر این محدودیت تضمین می‌نماید که زوج عملیات  $(l, k)$  از دو سفارش  $(l, k)$  که می‌بایست بر روی یک ماشین پردازش شوند، تداخلی نداشته باشند. محدودیت سری (۲-۱) شرط عدم انتظار میانی عملیات یک سفارش را نمایش می‌دهد. محدودیت‌های (۳-۱) حداکثر زمان تکمیل سفارشات را محاسبه می‌نماید. محدودیت سری (۴-۱) الزام می‌نماید که زمان تکمیل تمامی سفارشات از موعد تحویل سفارشات (تعیین شده توسط مشتریان) نباید بیشتر باشد. اما از آنجائیکه  $C_{max}$  یک معیار عملکرد عادی<sup>۱</sup> است، لذا می‌توان نتیجه گرفت که با برآورده شدن محدودیت (۴-۱) مسلماً مقدار  $C_{max}$  افزایش خواهد یافت. لذا می‌توان نتیجه گرفت که در راستای کمینه‌سازی تابع هدف  $C_{max}$ ، در صورت وجود جواب موجه برای مسأله، همزمان این محدودیت نیز ارضاء خواهد شد. لذا بدون از دست دادن کلیت می‌توان این محدودیت را از مدل حذف نمود. معادله (۵-۱) محدودیت مالی مصرف منابع را حداکثر تا مقدار  $B$  واحد نشان می‌دهد. محدودیت سری (۶-۱) میزان منبع تخصیص داده شده به عملیات  $O_{ij}$  را تعیین می‌نماید، و در نهایت محدودیت‌های سری آخر (۷-۱) شرط غیر منفی بودن متغیرهای  $t_{ij}$ ،  $C_{max}$ ،  $f_{ij}(r_{ij})$  و همچنین ماهیت صفر و یکی متغیرهای  $\delta_{ij}$  و  $Z_{ijkl}$  را اعمال می‌نمایند. مدل در نظر گرفته شده در این مقاله، حالت خاصی از مسأله با  $m = 2$  است.

### ۳. رویکرد حل تحلیلی جهت زیرمسأله زمانبندی

#### سفارشات

در این قسمت تعدادی از ویژگی‌های ساختاری مسأله و همچنین حالات مختلفی از آن را در قالب روابط ریاضی استخراج می‌نماییم. در ابتدا تعدادی از نمادهای مسأله‌ی دو ماشین را تعریف می‌نماییم. مسأله‌ی پیشنهادی، کمینه‌سازی زمان تکمیل کل کارها در حالتی است که تولیدکننده در ابتدای افق برنامه‌ریزی تعدادی سفارش از مشتریان دریافت می‌نماید. هر کدام از سفارشات دارای یک زمان تحویل مطلوب بوده و جهت تکمیل شدن نیاز به دو عملیات متفاوت دارند. به دلیل ساختار بعضی از سفارشات در صنعت چاپ، انتظار بین دو عمل از هر سفارش مجاز نیست. جهت ساده‌سازی، زمان پردازش اولین عمل از هر سفارش  $O_i$  را با  $a_i$  و دومین عمل را با  $b_i$  نمایش می‌دهیم. در ادامه در راستای در نظر گرفتن محدودیت عدم انتظار سفارشات،

تابع زمان پردازش عملیات  $O_{ij}$  وابسته به حجم تخصیص داده شده از منابع ۱ تا  $p$

ضریب زاویه هزینه تغییر در زمان پردازش عملیات  $O_{ij}$  از طریق تخصیص منبع بیشتر

مجموعه زوج عملیاتی از دو سفارش  $O_i$  و  $O_j$  که باید بر روی یک ماشین پردازش شوند

حداکثر مقدار منبع موجود جهت تخصیص به عملیات  $O_{ij}$

حداقل مقدار منبع موجود جهت تخصیص به عملیات  $O_{ij}$

حداکثر مقدار بودجه در دسترس

یک عدد مثبت بزرگ

#### متغیرهای تصمیم مدل:

حداکثر زمان تکمیل سفارشات  $C_{max}$

زمان آغاز پردازش عملیات  $O_{ij}$   $t_{ij}$

حجم منابع تخصیص داده شده به عملیات  $O_{ij}$   $r_{ij}$

متغیر صفر و یک که نشان‌دهنده تخصیص و یا عدم تخصیص منبع بیشتر به عملیات  $O_{ij}$  است.  $\delta_{ij}$

متغیر ۰-۱ که بیان می‌نماید کدام یک از دو عملیات  $O_{jk}$  و  $O_{il}$  زودتر انجام شود (بر اساس مسیر تولید سفارشات، این دو عملیات بر روی ماشین مشترک باید پردازش شوند، یعنی  $O_{il}, O_{jk} \in E_{ij}$ )  $Z_{ijkl}$

مدل ریاضی مسأله با توجه به پارامترهای تعریف شده و به منظور کمینه کردن  $C_{max}$  بصورت زیر فرمول‌بندی می‌شود:

$$P1: \quad \min C_{max} \quad (1)$$

Subjected to:

$$\begin{cases} t_{il} + f_{il}(r_{il}) \leq t_{jk} + MZ_{ijkl} \\ t_{jk} + f_{jk}(r_{jk}) \leq t_{il} + M(1 - Z_{ijkl}) \end{cases} \quad (2)$$

$$\text{for } i < j, \quad \text{all } (l, k) \in E_{ij}$$

$$t_{ij} + f_{ij}(r_{ij}) = t_{i(j+1)} \quad \text{for } i = 1, 2, \dots, n, \quad j = 1, 2, \dots, m - 1 \quad (3)$$

$$t_{im} + f_{im}(r_{im}) \leq C_{max} \quad \text{for } i = 1, 2, \dots, n \quad (4)$$

$$t_{im} + f_{im}(r_{im}) \leq d_i \quad \text{for } i = 1, 2, \dots, n \quad (5)$$

$$\sum_{i=1}^n \sum_{j=1}^m \beta_{ij} r_{ij} \leq B \quad (6)$$

$$r_{ij} = \delta_{ij} U_{ij} + (1 - \delta_{ij}) L_{ij} \quad \text{for } i = 1, 2, \dots, n, \quad j = 1, 2, \dots, m \quad (7)$$

$$C_{max} \geq 0, \quad t_{ij} \geq 0, \quad f_{ij}(r_{ij}) \geq 0, \quad \delta_{ij} = 0, 1 \quad Z_{ijkl} = 0, 1 \quad (8)$$

<sup>1</sup> Regular Measure

با کمترین عضو اضافه می‌شود. بعد از افزودن سفارشات مجاری، تعداد اعضای دو مجموعه  $P_1$  و  $P_2$  برابر خواهد شد. جهت تعیین بهترین زوج سفارشات که دارای کمترین سهم از زمان بیکاری ماشین‌ها هستند، مسأله تعیین زوج سفارشات را بصورت یک مسأله تخصیص فرمول بندی می‌کنیم.

$$Z = \sum_{i \in P_1} \sum_{j \in P_2} D_{ij} X_{ij} \quad (9)$$

$P_2$ : Minimize

Subject to:

$$\sum_{i \in P_1} X_{ij} = 1 \quad \text{for all } j \in P_2 \quad (10)$$

$$\sum_{j \in P_2} X_{ij} = 1 \quad \text{for all } i \in P_1 \quad (11)$$

$$X_{ij} = 0, 1 \quad \text{for } i \in P_1, j \in P_2 \quad (12)$$

تابع هدف (۲) مقدار کل سهم زوج سفارشات تشکیل شده را از زمان بیکاری ماشین‌ها تعیین می‌کند و در آن  $X_{ij}$  متغیر تصمیم صفر و یک است و تعیین می‌نماید که آیا زوج سفارش  $[O_i, O_j]$  تشکیل شود یا خیر. محدودیت سری (۲-۱) الزام می‌نماید که هر سفارش عضو مجموعه  $P_2$  تنها با یک سفارش از مجموعه  $P_1$  زوج شود. همچنین محدودیت سری (۲-۲) تضمین می‌نماید که هر سفارش عضو  $P_1$  نیز تنها با یک سفارش از مجموعه  $P_2$  زوج شود.

#### ۴. توسعه کران پائین جهت زیرمسأله زمانبندی زوج سفارشات

در این قسمت سه کران پائین بر اساس سه رویکرد مختلف به مسأله تحقیق، ارائه می‌شود. با استفاده از کران‌های پائین ارائه شده می‌توان تخمین بهتری نسبت به پاسخ بهینه مسأله به دست آورد. هر اندازه کران پائین به مقدار بهینه واقعی نزدیک‌تر باشد، ارزیابی پاسخ‌های بدست آمده از الگوریتم‌های مختلف با دقت بیشتری انجام می‌شود. جهت محاسبه کران پائین مسأله  $P_3$ ، فرض می‌کنیم  $\pi$  زیرمجموعه‌ای از  $k$  زوج سفارشات برنامه‌ریزی شده را نمایش می‌دهد، بطوریکه  $\pi = \{OP_{[1]}, OP_{[2]}, \dots, OP_{[k]}\}$ ،  $k < h$  و  $OP_{[u]}$  زوج سفارش برنامه‌ریزی شده در موقعیت  $u$ -ام را نمایش می‌دهد. در این صورت بدیهی است که مسأله‌ی باقیمانده، مسأله‌ای مشابه با  $P_3$  است با این تفاوت که در آن تنها در مورد توالی  $(h-k)$  زوج سفارش باقیمانده تصمیم‌گیری می‌شود. این زمانبندی جزئی<sup>۲</sup> از مسأله‌ی  $P_3$  را  $P_{\pi}^k$  می‌نامیم و

از مفهومی تحت عنوان «زوج سفارش» را تعریف می‌نمائیم. و سپس بر اساس برنامه‌ریزی ریاضی زوج سفارشات بهینه را تعیین نموده و مقداری اولیه جهت کران بالای تابع هدف تعیین می‌نمائیم. سپس حالات مختلف توالی زوج سفارشات را بصورت ریاضی استخراج می‌نمائیم.

#### ۳-۱. تعیین زوج سفارشات بهینه

در این قسمت مفهومی تحت عنوان زوج سفارش و همچنین نقش آن در الگوریتم پیشنهادی توصیف می‌شود. مجموعه‌ی  $[O_i, O_j]$  را که از دو سفارش  $O_i$  و  $O_j$  تشکیل شده است، یک زوج سفارش تعریف می‌شوند، اگر و تنها اگر:

(۱) توالی تکنولوژیکی انجام کار برای دو سفارش عکس

هم باشند ( $M_1 \rightarrow M_2, M_2 \rightarrow M_1$ ).

(۲)  $O_{i2}$  بلافاصله بعد از  $O_{j1}$  و همچنین  $O_{j2}$  بلافاصله

بعد از  $O_{i1}$  آغاز شوند.

همانطوریکه قابل استنباط است، تشکیل زوج سفارشات منجر می‌شود که محدودیت عدم انتظار تضمین شود. در ادامه رویکردی را جهت تشکیل مناسب‌ترین زوج سفارشات توسعه می‌دهیم. جهت اختصار زوج سفارش  $k$ ام را با  $OP_k$  نمایش می‌دهیم. بهترین ترکیب جهت تشکیل زوج سفارشات، ترکیبی است که در آن تابع هدف مسأله که همان زمان تکمیل کارها است بهینه شود. بر اساس مفهوم برابری معیارها، تعیین ترکیب بهینه‌ی زوج سفارشات جهت کمینه‌سازی معیار  $I$ ، مسلماً معیار  $C_{max}$  را نیز بهینه خواهد نمود. در همین راستا مفهوم «سهم زوج سفارش  $[O_i, O_j]$  از زمان بیکاری ماشین‌ها» را در ادامه تعریف می‌نمائیم. با تعریف  $s_{ij} = |a_i - a_j|$  و  $l_{ij} = |b_i - b_j|$  در این صورت سهم زوج سفارش  $[O_i, O_j]$  از زمان بیکاری ماشین‌ها بصورت زیر محاسبه می‌شود.

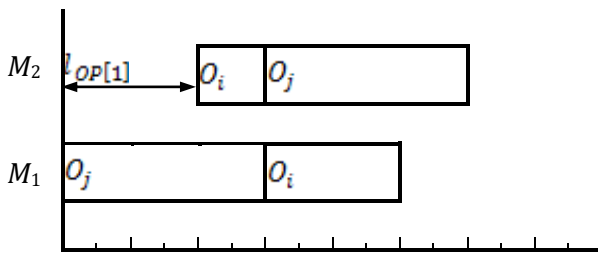
$$D_{ij} = s_{ij} + l_{ij} = |a_i - a_j| + |b_i - b_j|$$

زوج سفارشات بهینه، ترکیبی از سفارشات خواهند بود که منجر به کمترین مقدار  $D_{ij}$  شوند. در راستای تعیین این ترکیب، الگوریتم زیر بر اساس مدل مسأله‌ی تخصیص متقارن<sup>۱</sup> ارائه می‌شود. ابتدا مقدار سهم هر زوج سفارش بالقوه  $[O_i, O_k]$  از زمان بیکاری ماشین‌ها  $D_{ik}$  محاسبه می‌شود. سپس مجموعه‌ی  $P_1$  را مجموعه‌ی سفارشات که از ماشین  $M_1$  شروع می‌شوند و مجموعه‌ی  $P_2$  را مجموعه سفارشات که از ماشین  $M_2$  آغاز می‌شوند، قرا می‌دهیم. در این حالت بدیهی است که  $|P_1| + |P_2| = n$ . اگر  $|P_1| \neq |P_2|$ ، به تعداد  $||P_1| - |P_2||$  سفارش مجازی  $O'_d$  با  $a_d = b_d = 0$  تولید نموده و به مجموعه

<sup>2</sup> Partial Schedule

<sup>1</sup> Symmetric Assignment Problem



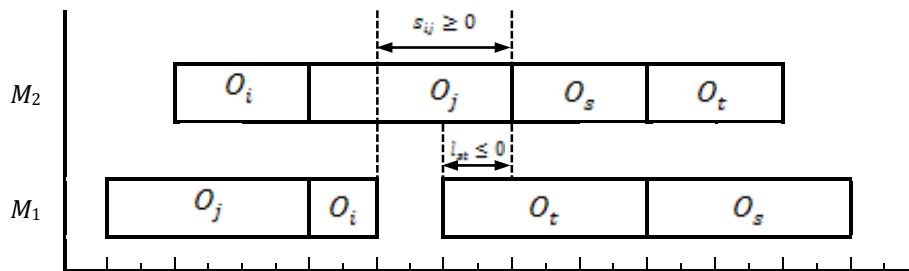


شکل ۱. نمایش زمان بیکاری زیرمسأله  $P_{\pi}^1$

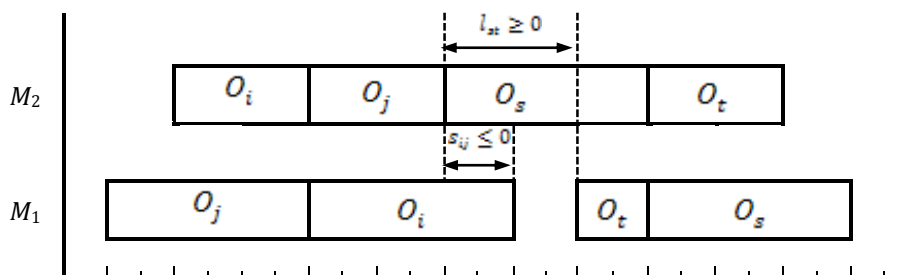
در حالتیکه  $k$  زوج سفارش برنامه ریزی شده باشند، زمان بیکاری ماشین‌ها مربوط به زیرمسأله  $P_{\pi}^k$  که با ورود زوج سفارش  $OP_{k+1}$  تغییر نخواهد کرد، بصورت زیر محاسبه می‌شود.

$$\mu_{\pi} = \left( \sum_{i \in \pi} (|l_{OP[i]}| + |s_{OP[i]}|) - s_{OP[i]} \right) - \sum_{i=1}^{k-1} (OP_{[i]} \text{ و } OP_{[i+1]})$$

بیکاری ماشین‌ها، علاوه بر  $\mu_{\pi}$  (زمان بیکاری محقق شده ناشی از زوج سفارشات برنامه ریزی شده)، می‌بایست حد پائینی برای زمان بیکاری بالقوه‌ی ناشی از زوج سفارشات که هنوز برنامه ریزی نشده و در مراحل بعدی افزوده می‌شوند  $s_{\pi}$ ، محاسبه شود. جهت تعیین این حد پائین، از امکان پوشش داده شدن زمان بیکاری زوج سفارشات  $[O_i, O_j]$  و  $[O_s, O_t]$  که به ترتیب دارای  $l_{st} \leq 0$  و  $s_{ij} \geq 0$  یا  $l_{st} \geq 0$  و  $s_{ij} \leq 0$  استفاده می‌نماییم. شکل‌های (۲) و (۳) زیر این دو حالت را نمایش می‌دهد.



شکل ۲. نمایش شرط  $s_{ij} \geq 0$  و  $l_{st} \leq 0$  جهت کاهش زمان بیکاری ناشی از توالی دو زوج سفارش



شکل ۳. نمایش شرط  $s_{ij} \leq 0$  و  $l_{st} \geq 0$  جهت کاهش زمان بیکاری ناشی از توالی دو زوج سفارش

در ادامه ۳ کران پائین جهت جواب بهینه‌ی حاصل از  $P_{\pi}^k$  ارائه می‌دهیم.

۴-۱. کران پائین بر مبنای سهم زوج سفارشات ( $LB_1$ )

فرض می‌کنیم که اولین زوج سفارش برنامه ریزی شده است و بنابراین مجموعه  $\pi$  تنها دارای یک عضو است. در این صورت زمان بیکاری ماشین‌ها متناظر با زمانبندی جزئی  $P_{\pi}^1$  که در این مرحله ایجاد شده و با افزودن زوج سفارشات جدید تغییر نخواهد کرد برابر با  $|l_{OP[1]}| = |a_{i[1]} - a_{j[1]}|$  که در شکل (۱) نمایش داده شده است.

عبارت اول در  $\mu_{\pi}$  مجموع زمان بیکاری زوج سفارشات در حالتی که مستقلاً در نظر گرفته شوند را محاسبه می‌نماید، در حالیکه عبارت دوم میزان بهبود زوجی ناشی از توالی زوج سفارشات را، زمانی که زوج سفارشات در کنار هم یک زمانبندی جزئی را تشکیل می‌دهند، از عبارت اول کم می‌کند تا زمان بیکاری  $P_{\pi}^k$  را محاسبه نماید. عبارت دوم با استفاده از میزان بهبود در زمان بیکاری ماشین‌ها ناشی از توالی زوج سفارشات محاسبه می‌شود. جهت تعیین کران پائین مبتنی بر سهم زوج سفارشات از زمان

که در آن  $\sum_{[i,j] \in \pi'} l_{ij} + \sum_{[i,j] \in \pi'} s_{ij}$  جمع جبری ناشی از پوشش زمانی بیکاری زوج سفارشات است. این جمع جبری در صورتیکه دقیقاً برای زوج سفارشات مجاور صورت گیرد، مقدار دقیق تابع هدف را برای توالی داده شده از زوج سفارشات محاسبه می‌نماید. اما از آنجائیکه توالی بهینه مشخص نیست لذا جمع جبری تمامی زوج سفارشات (بدون در نظر گرفتن توالی) کران پائینی جهت زمان بیکاری بهینه ماشین‌ها ناشی از ورود زوج سفارشات برنامه‌ریزی نشده خواهد بود. همچنین مقدار  $S_{OP[k]}$  ناشی از زمان بیکاری زوج سفارش آخر ( $k$ -ام) برنامه‌ریزی شده است که ممکن است در اثر هم‌پوشانی با اولین زوج سفارش ورودی پوشش داده شده و کاهش یابد. لذا مقدار نهایی کران پائین برای مسأله  $P_{\pi}^k$  برابر با  $\mu_{\pi} + \sigma_{\pi}$  و بصورت زیر خواهد بود.

$$LB_1 = \left( \sum_{i \in \pi} (|l_{OP[i]}| + |s_{OP[i]}|) - |s_{OP[k]}| \right) - \sum_{i=1}^{k-1} (OP_{[i]} \text{ و } OP_{[i+1]}) \text{ (بهینود UB برای } OP_{[i]} \text{ و } OP_{[i+1]}) \\ + \left| \sum_{[i,j] \in \pi'} l_{ij} + \sum_{[i,j] \in \pi'} s_{ij} + s_{OP[k]} \right|$$

۳-۴. کران پائین بر مبنای آزادسازی متغیرهای صفر و یک<sup>۱</sup> ( $LB_3$ )

آزادسازی متغیرهای ۰-۱ یک برنامه‌ریزی عدد صحیح مختلط<sup>۲</sup>، یکی از روش‌های متداول در محاسبه کران پائین است. برای زیرمسأله  $P_{\pi}^k$ ، مسأله برنامه‌ریزی خطی ناشی از آزادسازی متغیرهای صفر و یک را  $P_4$  می‌نامیم. جواب بهینه برنامه‌ریزی خطی  $P_4$  کران پائینی برای جواب بهینه مسأله اصلی  $P_{\pi}^k$  خواهد بود.

$$P_4: \quad \min C_{max} \quad (17)$$

$$\begin{cases} t_{ii} + f_{ii}(\bar{r}_{ii}) \leq t_{jk} + MZ_{ijkl} \\ t_{jk} + f_{jk}(\bar{r}_{ii}) \leq t_{ii} + M(1 - Z_{ijkl}) \\ \forall i, j \in V(\pi'), i < j, \text{ all } (l, k) \in E_{ij} \end{cases} \quad (18)$$

$$t_{i1} + f_{i1}(\bar{r}_{i1}) = t_{i2} \quad \forall i \in V(\pi') \quad (19)$$

$$t_{i2} + f_{i2}(\bar{r}_{i2}) \leq C_{max} \quad \forall i \in V(\pi') \quad (20)$$

$$C_{max} \geq 0, \quad t_{ij} \geq 0, \quad Z_{ij} \geq 0 \quad (21)$$

### ۵. الگوریتم شاخه و کران پیشنهادی

تکنیک شاخه و کران یکی از رویکردهای متداولی است که جهت حل بهینه‌ی انواع مختلفی از مسائل بهینه‌سازی پیچیده مورد استفاده قرار می‌گیرد. همانطوریکه توسط Pinedo (۱۹۹۵)

همانطوریکه در شکل‌ها نمایش داده شده است، کاهش زمان بیکاری ناشی از توالی زوج سفارشات تنها در دو حالت با شرط‌های بالا امکان‌پذیر است.

همچنین همانطوریکه در شکل مشخص است، می‌توان نتیجه گرفت که میزان بیکاری حاصل از توالی زوج سفارشات در هر دو حالت برابر است با جمع جبری  $s_{ij}$  و  $l_{st}$  زوج سفارشات مجاور یعنی  $|l_{st} + s_{ij}|$  است. از این ویژگی زوج سفارشات استفاده می‌نمائیم تا کران پائینی برای زمان بیکاری ناشی از ورود زوج سفارشات برنامه‌ریزی نشده، بصورت زیر استخراج شود.

$$\sigma_{\pi} = \left| \sum_{[i,j] \in \pi'} l_{ij} + \sum_{[i,j] \in \pi'} s_{ij} + s_{OP[k]} \right|$$

۲-۴. کران پائین بر مبنای آزادسازی محدودیت ( $LB_2$ )

در این بخش کران پائینی بر مبنای آزادسازی محدودیت عدم انتظار برای مسأله  $P_{\pi}^k$  پیشنهاد می‌شود. به عبارت دیگر زمانیکه تعداد  $k$  زوج سفارش برنامه‌ریزی شده و  $(h - k)$  زوج سفارش باقی مانده‌اند، مسأله آزادسازی شده یعنی  $P_3$  بصورت زیر در نظر گرفته می‌شود.

$$P_3: \quad \min C_{max} \quad (13)$$

Subject to:

$$\begin{cases} t_{ii} + f_{ii}(\bar{r}_{ii}) \leq t_{jk} + MZ_{ijkl} \\ t_{jk} + f_{jk}(\bar{r}_{ii}) \leq t_{ii} + M(1 - Z_{ijkl}) \end{cases} \quad (14)$$

$$\forall i, j \in V(\pi'), i < j, \text{ all } (l, k) \in E_{ij}$$

$$t_{i2} + f_{i2}(\bar{r}_{i2}) \leq C_{max} \quad \forall i \in V(\pi') \quad (15)$$

$$C_{max} \geq 0, \quad t_{ij} \geq 0, \quad Z_{ijkl} = 0, 1 \quad (16)$$

که در آن  $V(\pi')$  مجموعه سفارشات است که مجموعه‌ی  $\pi'$  را تشکیل می‌دهند و  $|V(\pi')| = 2(h - k)$ . مقدار کران پائین  $LB_2$  مجموع زمان بیکاری بدست آمده از جواب حاصل از مدل  $P_4$  خواهد بود. جهت محاسبه این کران از بسته نرم‌افزاری TOMLAB که از Solverهایی همچون CPLEX جهت حل مدل‌های بهینه‌سازی عدد صحیح و عدد صحیح مختلط استفاده می‌نماید، استفاده شد.

<sup>1</sup> LP Relaxation

<sup>2</sup> Mixed Integer Programming (MIP)

توسعه داده شده‌اند. در همین راستا، نزدیک‌ترین کران پائین به جواب بهینه در هر گره از درخت تصمیم، به عنوان کران نهایی محاسبه می‌شود.

### ۵-۳. استراتژی جست و جو

در هر الگوریتم شاخه و کران، استراتژی جست و جو، نحوه پوشش گره‌های درخت تصمیم در راستای رسیدن به پایان الگوریتم را تعیین می‌نماید. بطور کلی استراتژی‌های مورد استفاده در پیمایش درخت تصمیم، شامل دو روش عمق اول<sup>۶</sup> و سطح اول<sup>۷</sup> است، که در ادامه به بررسی بیشتر آن‌ها پرداخته می‌شود.

### ۵-۳-۱. روش جست و جوی عمق اول DFS

در این روش شاخه‌ای که جهت بررسی انتخاب می‌شود آنقدر ادامه پیدا می‌کند تا یا به عمق برسد (کران پائین ناشی از زیرمسائل تولید شده از جواب مرجع بدتر باشد)، و یا جواب کامل در لایه آخر حاصل شود. در حالت دوم نیز دو حالت ممکن است، اگر مقدار ارزیابی جواب حاصل از مقدار کران بالا بدتر باشد، جواب حاصل حذف می‌شود و در غیر این صورت، جایگزین کران بالا جهت ادامه‌ی جست و جو می‌شود. این کران بالای جدید امکان حذف شاخه‌هایی که منجر به جواب بهینه نمی‌شوند را افزایش داده و منجر به کاهش محاسبات و افزایش کارایی الگوریتم می‌شود.

### ۵-۳-۲. روش جست و جوی سطح اول BFS

همانطوریکه از عنوان این روش جست و جو مشخص است، فرآیند شاخه‌زنی در این روش تنها برای گرهی انجام می‌گیرد که دارای کمترین کران پائین (در بین کلیه‌ی گره‌های لایه‌ی آخر) باشد. در مرحله‌ی صفر، تمام شاخه‌های ممکن از مسأله‌ی اصلی رسم می‌شود، و در هر مرحله گره‌های دارای کران پائین بیشتر از کران بالا از بررسی بیشتر حذف شده و در بین گره‌های باقیمانده، گره با کمترین کران شاخه‌زنی می‌شود.

### ۶. اثبات چند قاعده چیرگی<sup>۸</sup> جهت بهبود کارایی

#### الگوریتم شاخه و کران

در این بخش به اثبات چند قاعده چیرگی که در فرآیند شاخه‌زنی و شاخه و کران، منجر به کاهش فضای جست و جو و افزایش کارایی الگوریتم خواهند شد.

توصیف شده است، شاخه و کران یک الگوریتم شمارش ضمنی<sup>۱</sup> است که در آن جواب‌های یک مسأله بهینه‌سازی توسط گره‌هایی در یک درخت تصمیم نمایش داده می‌شوند. یک الگوریتم شاخه و کران تعداد گره‌های موجود در درخت را که می‌بایست بطور صریح بررسی شوند، از طریق استفاده از یک کران پائین کاهش می‌دهد. در حالت کلی، شاخه و کران با جواب‌های جزئی آغاز می‌شود و از طریق افزودن بخش‌هایی از جواب (شاخه‌زنی<sup>۲</sup>)، در هر مرحله جواب حاصل را تکمیل می‌نماید، تا در نهایت به جواب کامل در لایه‌ی آخر درخت برسد. در هر گره، یک کران پائین که نشان‌دهنده‌ی بهترین جوابی است که می‌تواند توسط جواب جزئی‌ای که توسط آن گره ارائه شده است، محاسبه شده و با مقدار کران بالا مقایسه می‌شود. اگر کران پائین محاسبه شده در یک گره، از مقدار جواب مرجع بزرگتر باشد (برای مسائل با تابع هدف کمینه‌سازی) به این معنی است که جواب‌هایی که در اثر ادامه شاخه‌زنی گره‌ی فعلی حاصل خواهند شد، مطمئناً از بهترین جواب شناخته شده تا به حال بهتر نخواهند بود. لذا شمارش ضمنی کلیه‌ی جواب‌های حاصل از ادامه‌ی این گره نشان می‌دهد که می‌توان شاخه‌زنی بیشتر این گره را متوقف نمود. در چنین شرایطی اصطلاحاً گفته می‌شود که شاخه‌ی متناظر با گره به عمق رسیده<sup>۳</sup> و هرس شده<sup>۴</sup> است. الگوریتم تا زمانیکه تمامی گره‌های حاصل منجر به جواب کامل شده باشند، یا به عمق رسیده باشند، و یا اینکه کران پائین یک جواب کامل برابر با کران بالا شود، ادامه پیدا می‌کند.

### ۵-۱. استراتژی شاخه‌زنی

شاخه‌زنی فرآیند تفکیک<sup>۵</sup> یک مسأله به چندین زیرمسأله است. در فرآیند شاخه‌زنی، مسأله‌ی اصلی  $P_0$  به تعدادی زیرمسأله کوچکتر  $P_{\pi}^k$  تجزیه می‌شود به نحویکه کل فرآیند شاخه‌زنی توسط درخت تصمیم نمایش داده می‌شود. در هر لایه از درخت تصمیم که توسط  $k$  نشان داده می‌شود، هر زیرمسأله به عنوان مبنایی جهت شاخه‌زنی لایه‌ی پائین‌تر قرار می‌گیرد بطوریکه  $U_{i \in \pi'} P_{\pi}^k = P_{\pi}^{k-1}$ . استراتژی شاخه‌زنی پیشنهادی در این تحقیق، مبتنی بر توالی زوج سفارشات تشکیل شده در بخش قبلی است بطوریکه در هر لایه از درخت، یک زوج سفارش بیشتر نسبت به لایه‌ی بالاتر برنامه‌ریزی شده است.

### ۵-۲. استراتژی کران پائین

استراتژی کران پائین بکارگرفته شده در الگوریتم شاخه و کران پیشنهادی بر اساس ۳ رویکرد حد پائینی است که در بخش (۴)

<sup>6</sup> Depth First Search DFS

<sup>7</sup> Best First Search BFS / Frontier Search / Branch-from-Lowest-Bound

<sup>8</sup> Dominance Rule

<sup>1</sup> Implicit Enumeration Algorithm

<sup>2</sup> Branching

<sup>3</sup> Fathomed

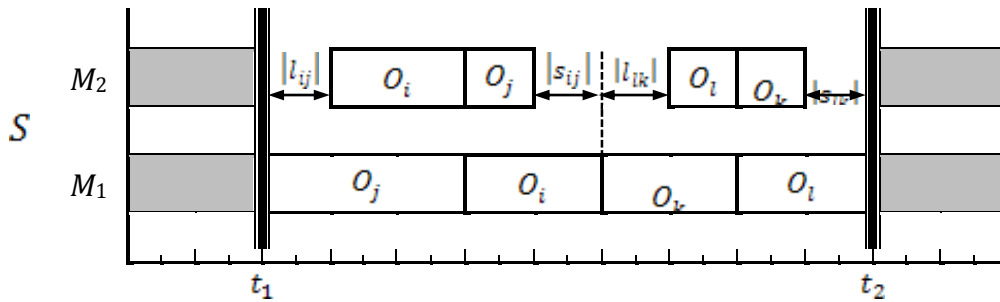
<sup>4</sup> Pruned

<sup>5</sup> Partitioning

**اثبات:**

ابتدا با استفاده از تعویض جفتی نشان می‌دهیم که هر توالی دلخواه از زوج سفارشات، جواب بهینه مسأله  $P_{\pi}^k$  خواهد بود. فرض می‌کنیم که  $S$  یک توالی دلخواه از زوج سفارشات است که در آن دو زوج سفارش  $[O_i, O_j]$  و  $[O_l, O_k]$  با شرط  $s_k < 0$ ،  $l_l < 0$  و  $s_l < 0$  و  $l_k < 0$  بصورت متوالی برنامه‌ریزی شده- اند (شکل ۴).

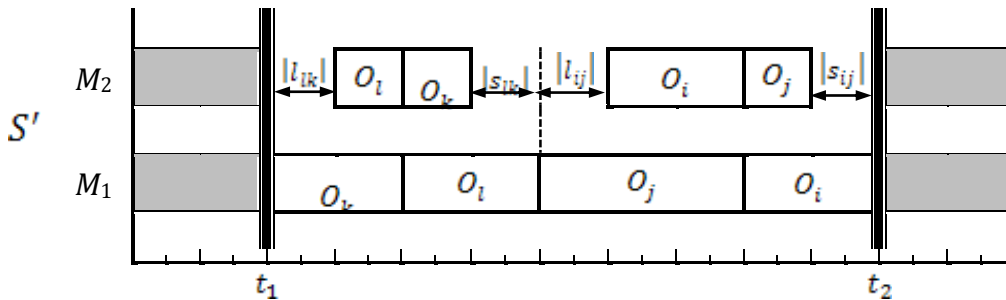
**قضیه ۱:** برای زیرمسأله  $P_{\pi}^k$  که در آن مجموعه  $\pi$  مجموعه زوج سفارشات برنامه‌ریزی شده است، اگر  $s_k$  و  $l_k$  به ازای تمامی زوج سفارشات برنامه‌ریزی نشده  $k \in (\pi)'$  هم علامت باشند، آنگاه هر توالی دلخواه از زوج سفارشات، یک جواب بهینه برای زیر مسأله  $P_{\pi}^k$  بوده و مقدار جواب بهینه برابر است با  $|\sum_{[i,j] \in \pi'} l_{ij}| + |\sum_{[i,j] \in \pi} s_{ij}|$ .



شکل ۴. زمانبندی اولیه  $S$  در تعویض جفتی

زمانبندی  $S'$  که در آن دو زوج سفارش جابجا شده‌اند، بصورت شکل (۵) خواهد بود.

زمان بیکاری متناظر با زمانبندی  $S$  برابر با  $(|l_{ij}| + |s_{ij}|) + (|l_{lk}| + |s_{lk}|)$  است. در این صورت



شکل ۵. زمانبندی  $S'$  حاصل از تعویض جفتی

ماشین‌ها نداشته و مقدار آن به ازای هر توالی دلخواه برابر  $|\sum_{[i,j] \in \pi'} l_{ij}| + |\sum_{[i,j] \in \pi} s_{ij}|$  خواهد بود.

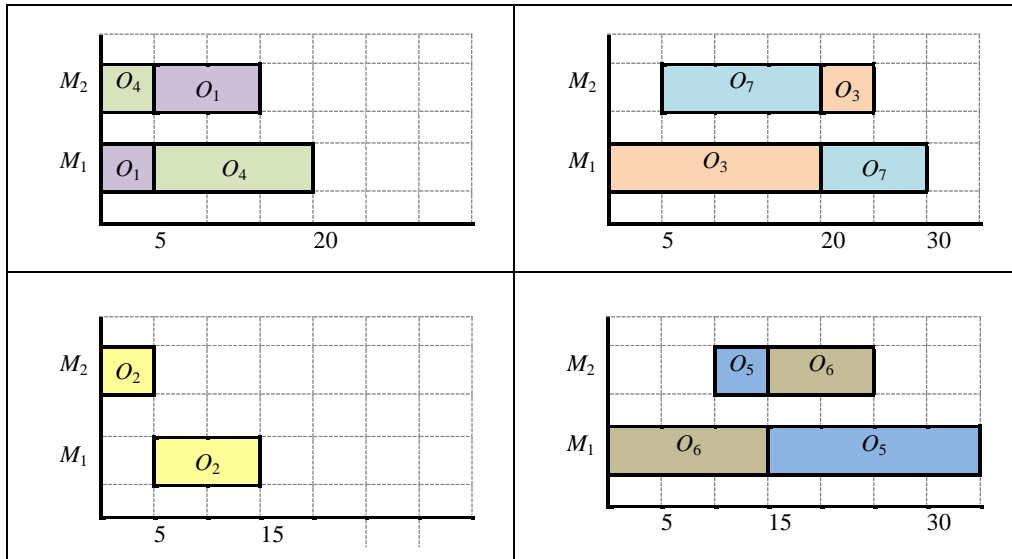
به طریق مشابه می‌توان در حالتیکه  $s_k > 0$ ،  $l_k > 0$ ،  $s_l > 0$  و  $l_l > 0$  می‌توان نشان داد که ماشین  $M_2$  گلوگاه بوده و قضیه را اثبات نمود. مثالی را در نظر می‌گیریم که در آن زوج سفارشات تشکیل شده از مدل تخصیص متقارن  $P_0$  بصورت شکل (۶) است. در این مسأله، به تعداد  $n = 7$  سفارش توسط سازنده در ابتدای افق برنامه‌ریزی دریافت شده است و نتیجه‌ی آن تشکیل  $h = 4$  زوج سفارش  $OP_1 = [O_4, O_1]$ ،  $OP_2 = [O_7, O_3]$ ،  $OP_3 = [O_2, O_8]$  و  $OP_4 = [O_5, O_6]$  بوده

مانطوریکه ملاحظه می‌شود میزان بیکاری ناشی از تعویض جفتی در زمانبندی جدید  $S'$  نیز برابر با  $(|l_{ij}| + |s_{ij}|) + (|l_{lk}| + |s_{lk}|)$  است. از آنجائیکه ترتیب سفارشات قبل از زمان  $t_1$  و همچنین بعد از زمان  $t_2$  برای دو زمانبندی  $S$  و  $S'$  ثابت است، لذا می‌توان نتیجه گرفت که هر دو زمانبندی منجر مقدار ثابتی از تابع هدف می‌شوند.

در این حالت ماشین  $M_1$  گلوگاه<sup>۱</sup> بوده و به همین دلیل زمان بیکاری آن برابر صفر است. همانطوریکه ملاحظه شد، تغییر در توالی زوج سفارشات در این حالت، تأثیری بر مقدار بیکاری

<sup>۱</sup> Bottleneck

است. همانطوریکه مشاهده می شود زوج سفارش  $OP_3$  از سفارش  $O_2$  و سفارش مجازی  $D_8$  تشکیل شده است.



شکل ۶. زوج سفارشات تشکیل شده حاصل از مدل تخصیص متقارن  $P_0$

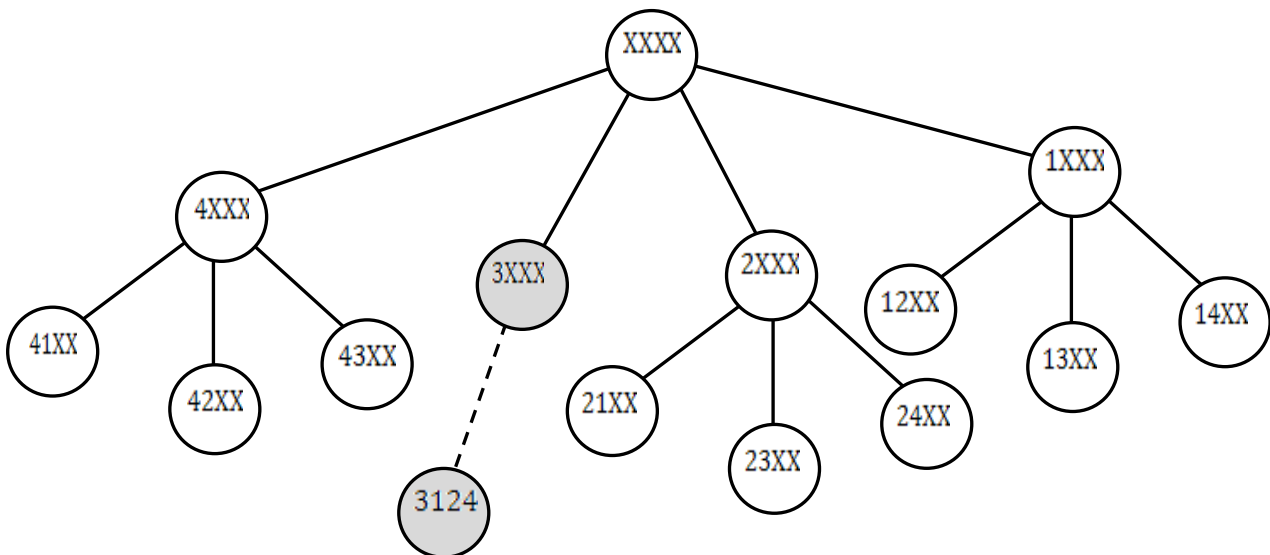
$$OP_2: \begin{cases} s_{73} = a_7 - a_3 = -5 \\ l_{73} = b_3 - b_7 = -5 \end{cases}$$

$$OP_4: \begin{cases} s_{56} = a_5 - a_6 = -10 \\ l_{56} = b_6 - b_5 = -10 \end{cases}$$

همانطوریکه مشاهده می شود، مقادیر  $s_k$  و  $l_k$  به ازای تمامی زوج سفارشات برنامه ریزی نشده در گره  $2XXX$  غیر مثبت بوده و شرط قضیه (۱) برقرار است. لذا نیازی به بررسی تمامی زیرشاخه های این گره نبوده و فقط کافی است یکی از زیرشاخه ها را تا به انتها ادامه داده (به دلخواه) تا به جواب موجه برسیم (2134).

در این مرحله، حالتی از فرآیند شاخه و کران را که در شکل (۷) مشخص شده است در نظر می گیریم که در آن زوج تنها سفارش برنامه ریزی شده است (گره  $3XXX$ ). در این مرحله مجموعه ی زوج سفارشات برنامه ریزی شده  $\pi = \{OP_3\}$  و در نتیجه مجموعه ی زوج سفارشات برنامه ریزی نشده  $\pi' = \{OP_1, OP_2, OP_4\}$  خواهد بود. در این حالت شرط برقراری قضیه (۱) را برای زوج سفارشات مجموعه  $\pi'$  تست می نماییم.

$$OP_1: \begin{cases} s_{41} = a_4 - a_1 = 0 \\ l_{41} = b_1 - b_4 = -5 \end{cases}$$

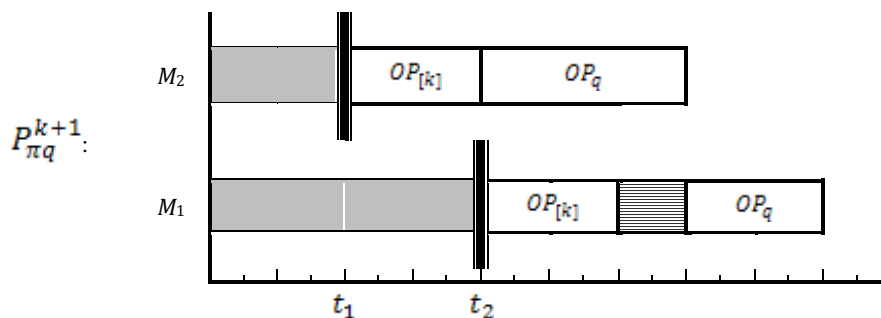


شکل ۷. نمایش قضیه (۱) برای گره  $3XXX$  از درخت تصمیم در الگوریتم شاخه و کران

## قضیه ۲:

برای زیرمسأله  $P_{\pi}^k$  که در آن مجموعه  $\pi$  مجموعه زوج سفارشات برنامه ریزی شده است، فرض کنید تمامی  $(h - k)$  زوج سفارش باقیمانده که در موقعیت  $k + 1$  می توانند برنامه ریزی بشوند، زوج سفارشات مجازی هستند. در این صورت اگر برای زوج سفارش  $OP_m$  که  $m \in \pi'$  ویژگی زیر برقرار باشد، جواب بهینه مسأله  $P_{\pi}^k$  حالتی است که در آن زوج سفارش  $OP_m$  در موقعیت  $k + 1$  برنامه ریزی شود.

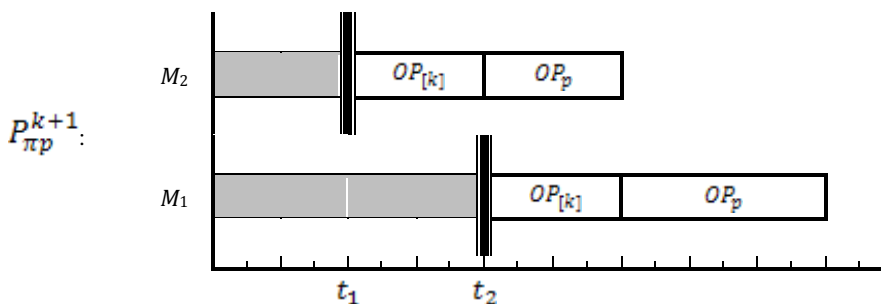
$$l_m = s[k]$$



شکل ۸. زمانبندی ناشی از برنامه ریزی  $P_{\pi q}^{k+1}$

اما اگر زمانبندی  $P_{\pi p}^{k+1}$  را در نظر بگیریم، همانطوریکه در شکل (۹) مشخص است، مقدار بیکاری حاصل از برنامه ریزی موقعیت  $k + 1$  صفر خواهد بود که منجر به بهبود مقدار کل بیکاری در زمانبندی نهایی می شود.

همانطوریکه در شکل ملاحظه می شود، برنامه ریزی زوج سفارش  $OP_q$  در موقعیت  $k + 1$  منجر به ایجاد زمان بیکاری برای ماشین  $M_1$  و به اندازه  $I_2 = l_q - s[k]$  می شود. همچنین داریم که زمان بیکاری  $I_2$  قابل جبران نبوده و در زمانبندی نهایی منجر به افزایش مقدار کل زمان بیکاری ماشین ها خواهد شد.



شکل ۹. زمانبندی ناشی از برنامه ریزی  $P_{\pi p}^{k+1}$

مبتنی بر تجزیه مسأله است. در همین راستا در بخش های قبل به ارائه یک رویکرد حل تحلیلی و دقیق جهت زیرمسأله ی زمانبندی سفارشات، با فرض مشخص بودن برنامه ظرفیت تولید، پرداختیم. هر جواب مسأله برنامه ریزی ظرفیت به عنوان ورودی به مسأله زمانبندی سفارشات بوده و زمان پردازش سفارشات را تعیین می نماید.

سپس رویکرد شاخه و کران اجرا شده و تابع هدف بدست آمده به عنوان ارزیابی جواب، به مسأله برنامه ریزی ظرفیت برگردانده

به طریقی مشابه می توان برقراری قضیه فوق را در حالت  $M_1 \rightarrow M_2$  و  $l_q < s[k]$  اثبات کرد.

۷. الگوریتم جستجوی همسایگی<sup>۱</sup> جهت زیر مسأله

## برنامه ریزی ظرفیت

همانطوریکه قبلاً مطرح شد، یکی از رویکردهای موفق در حل کارای مسائل هماهنگی در زمانبندی و زنجیره تأمین، روش های

<sup>۱</sup> Neighborhood Search Algorithm (NSA)

$$\begin{cases} r_{d_1, d_2} = 1 & \text{if } r_{d_1, d_2} = 0 \\ r_{d_1, d_2} = 0 & \text{if } r_{d_1, d_2} = 1 \end{cases} \quad (17)$$

همچنین در راستای بهبود توان جستجوی محلی در همسایگی ماتریس تخصیص  $R$ ، ساختار  $S_2$  بر اساس ایجاد تشدید بکار گرفته شده است. در این ساختار، دو بردار تصادفی  $r_1$  و  $r_2$  مشابه بردار  $r$  در ساختار  $S_1$  انتخاب شده بطوریکه  $r_1 \neq r_2$  و سپس درایه‌های ماتریس  $R$  متناظر با دو بردار  $r_1$  و  $r_2$  جابجا می‌شوند.

## ۲-۷. اثبات ویژگی ریاضی برای زیر مسأله برنامه‌ریزی ظرفیت

از طریق استخراج ویژگی‌های ریاضی مسأله‌ی مورد بررسی، می‌توان به یک سری قواعد چیرگی دست یافت که از طریق آن‌ها، حجم جستجوها و در نتیجه حجم محاسبات کاهش یافته و کارایی الگوریتم بهبود یابد. در ادامه یک ویژگی ریاضی متناظر با مسأله‌ی برنامه‌ریزی ظرفیت جهت بکارگیری در ساختار تولید همسایگی بیان و اثبات می‌شود.

### ویژگی ۱:

اگر  $R = [r_{ij}]$  یک ماتریس جواب شدنی برای زیرمسأله برنامه‌ریزی ظرفیت باشد و جواب بهینه مدل تخصیص  $P_0$  منجر به تولید زوج سفارشات با ویژگی  $r_{i1} \cdot s_i > 0$  شود، در اینصورت تخصیص ظرفیت بیشتر به عملیاتی که روی ماشین  $M_m$  انجام می‌شوند، هیچ بهبودی در مسأله ایجاد نمی‌کند.

$$M_m = \begin{cases} M_1 & \text{if } r_{i1} \cdot s_i > 0 \\ M_2 & \text{if } r_{i1} \cdot s_i < 0 \end{cases} \quad (18)$$

### اثبات:

براساس قضیه (۱) که قبلاً اثبات شد، جواب بهینه‌ی زیرمسأله زمانبندی سفارشات ناشی از ماتریس تخصیص  $R = [r_{ij}]$ ، هر ترتیب دلخواه از زوج سفارشات خواهد بود. در این صورت ترتیب دلخواه  $S$  از زوج سفارشات را در حالت  $r_{i1} \cdot s_i < 0$  در نظر می‌گیریم. شکل (۱۰) زوج سفارش  $[O_i, O_j]$  و  $[O_i, O_k]$  از زمانبندی  $S$  را نمایش می‌دهد.

همانطوریکه در شکل مشخص است، تخصیص ظرفیت بیشتر به عملیات دوم سفارش  $O_i$  نه تنها کاهشی در زمان بیکاری ماشین-ها نخواهد داشت، بلکه منجر به افزایش آن خواهد شد. رابطه  $I_S$  بیکاری متناظر با دو زوج سفارش در زمانبندی  $S$  را نشان می‌دهد.

می‌شود. این رویه تا تحقق شرط توقف ادامه خواهد یافت. در این بخش رویکردی مبتنی بر جستجوی همسایگی جهت حل کارای زیرمسأله تخصیص منابع خواهیم پرداخت. این زیرمسأله، با داشتن یک مسأله‌ی تصمیم‌گیری توأم برنامه‌ریزی ظرفیت تولید و زمانبندی تولید، یک تخصیص موجه از ظرفیت تولید به عملیات انجام می‌دهد.

سپس این بردار تخصیص به عنوان ورودی به زیر مسأله زمانبندی سفارشات وارد شده و فرآیند الگوریتم شاخه و کران اجرا می‌شود. در نهایت، میزان کیفیت جواب تولید شده، بر اساس ارزیابی انجام گرفته توسط الگوریتم شاخه و کران، به زیرمسأله برنامه‌ریزی ظرفیت بازگردانده می‌شود. این ارزیابی به عنوان بازخوری جهت هدایت و اصلاح جستجو در لایه‌ی بالا مورد استفاده قرار گرفته و تخصیص‌های جدید بر اساس این بازخور تولید می‌شوند و مجدد این فرآیند ادامه پیدا می‌کند. فرآیند تکاملی الگوریتم منجر به هدایت الگوریتم به جواب‌های بهینه (نزدیک بهینه) خواهد شد. در ادامه مراحل مختلف الگوریتم جستجوی همسایگی پیشنهادی بررسی می‌شود.

## ۱-۷. ساختار تولید همسایگی

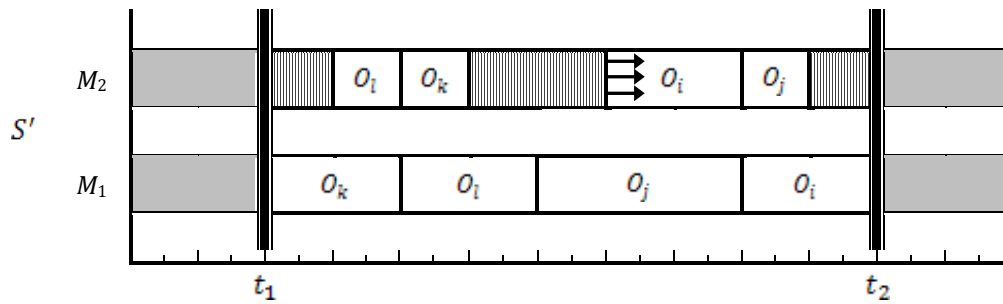
در این قسمت از یک ماتریس صفر و یک  $R = [r_{ij}]$  با ابعاد  $n \times 2$  جهت نمایش مقادیر تخصیص ظرفیت به سفارشات در زیر مسأله برنامه‌ریزی ظرفیت استفاده می‌شود. هر عضو از ماتریس  $r_{ij}$  میزان تخصیص منابع به عملیات  $O_{ij}$  را نمایش می‌دهد.

اگر مقدار  $r_{ij} = 1$  نشان‌دهنده تخصیص کران بالای مقادیر ظرفیت ممکن به  $O_{ij}$  و در اینصورت، مقدار کران پائین ظرفیت به  $O_{ij}$  تخصیص داده می‌شود.

با داشتن بردار اولیه تخصیص  $R_0 = [r_{ij}^0]$  با استفاده از اجرای مدل  $P_5$ ، دو ساختار تولید همسایگی در راستای اعمال دو کارکرد تشدید<sup>۱</sup> و تنوع<sup>۲</sup> بکار گرفته می‌شود. در ساختار اول  $S_1$  که مبتنی بر ایجاد تنوع در ماتریس  $R$  است، یک درایه تصادفی  $r = (d_1, d_2)$  که در آن  $d_1$  عدد صحیح تصادفی  $d_1 = 1 + (n - 1) \times \text{rand}()$  (بین ۱ و  $n - 1$ ) و  $d_2$  بطور تصادفی از بردار  $(M_1, M_2)$  انتخاب می‌شوند. سپس  $S_1$  بردار  $r$  را به عنوان ورودی دریافت نموده و تابع زیر را در راستای ایجاد تنوع در ماتریس  $R$  اجرا می‌نماید.

<sup>1</sup> Intensification

<sup>2</sup> Diversification



شکل ۱۰. تخصیص ظرفیت به زوج سفارش  $[O_i, O_j]$  از زمانبندی  $S$

جریمه در میزان نقض محدودیت‌ها، یک تابع جریمه به تابع هدف اضافه شده و مسأله به فرم زیر تبدیل می‌شود.

$$\text{Minimize } F(X) = f(X) + P(X), \quad X = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n \quad (21)$$

که در آن  $P(X)$  تابع جریمه است. در حالت عمومی، یک فرم متداول از تابع جریمه بصورت زیر است.

$$P(X) = K \cdot \sum_{i=1}^q (v_i(X))^\beta \quad (22)$$

که در آن  $K$  نشان‌دهنده ضریب جریمه است،  $\beta$  پارامتر جریمه بوده که معمولاً برابر با یک فرض می‌شود، و

$$v_i(X) = \max \{0, g_i(X)\} \quad (23)$$

اما یکی از تصمیمات مهم در طراحی هر تابع جریمه‌ای، تعیین میزان بزرگی ضریب جریمه است، که تأثیر بسزایی بر عملکرد الگوریتم اصلی خواهد داشت. یکی از روش‌های متداول تعیین مقدار ثابت برای ضریب جریمه در طول کل فرآیند تکامل است که تحت عنوان جریمه استاتیک<sup>۲</sup> در ادبیات مطرح است. در این روش، یک مقدار ثابت از جریمه به تابع هدف تمامی جواب‌هایی که ناحیه شدنی را به هر طریقی نقض می‌نمایند، اضافه می‌شود (در مسائل با تابع کمینه‌سازی). یکی از انواع تابع جریمه استاتیک بر اساس تعداد محدودیت‌های نقض شده و بصورت زیر است.

$$F(X_k) = f(X_k) + \sum_{i=1}^m \beta_i \rho_i \quad (24)$$

که در آن  $F(X_k)$  یک تابع جریمه استاتیک را نمایش می‌دهد،  $f(X_k)$  میزان تابع هدف اصلی (تابع ارزیابی<sup>۳</sup>) متناظر با جواب

$$I_S = |l_{ij}| + |s_{ij}| + |l_{ik}| + |s_{ik}| \quad (19)$$

تخصیص ظرفیت بیشتر به عملیات دوم سفارش  $O_i$  منجر به کاهش زمان پردازش آن و در نتیجه افزایش  $|l_{ij}|$  می‌شود. لذا مقدار بیکاری متناظر با این افزایش ظرفیت  $I_S$  افزایش خواهد داشت. همین حالت برای کلیه عملیاتی که روی ماشین  $M_2$  انجام می‌شوند برقرار است. به طریقی مشابه می‌توان برای حالت  $r_i, s_i > 0$  نیز برقراری ویژگی (۱) را نشان داد.

#### ۴-۷. ارزیابی جواب‌ها

همانطوریکه قبلاً مطرح شد، تحویل به موقع و قابل اطمینان محصولات به مشتریان یکی از پارامترهای مهم و تأثیرگذار بر عملکرد شرکت‌های تولیدی در بازارهای رقابتی محسوب می‌شود. به همین منظور در مدل پیشنهادی از رویکرد یکپارچه به مقوله برنامه‌ریزی ظرفیت تولید و زمانبندی تولید، موعد تحویل سفارش به مشتری به عنوان محدودیت‌های محکم و غیرقابل عدول، در نظر گرفته شده‌اند. علاوه بر این، میزان بودجه در دسترس جهت برنامه‌ریزی منابع مورد نیاز به عملیات تولیدی نیز جزء محدودیت‌های مدل در نظر گرفته شده است. در این بخش به ارائه مکانیزمی جهت ارضای این محدودیت‌ها در فرآیند تکاملی الگوریتم جستجوی همسایگی پرداخته می‌شود. تا به حال رویکردهای متفاوتی در ادبیات در رابطه با ارضای محدودیت‌ها<sup>۱</sup> در فرآیندهای تکاملی جستجوی جواب ارائه شده است. موفق‌ترین این رویکردها شامل تکنیک تابع جریمه است که در آن جواب‌های غیرشدنی از طریق افزایش در مقدار تابع ارزیابی جریمه می‌شوند. بطور کلی مسائل محدودیت‌های غیرمساوی بصورت زیر فرمولبندی می‌شوند.

$$\text{Minimize } f(X), \quad X = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n \quad (20)$$

$$\text{s.t. } g_i(X) \leq 0 \quad \text{for } i = 1, 2, \dots, m$$

که در آن،  $f(X)$  تابع هدف بوده و ناحیه شدنی از طریق مجموعه-ای از  $m$  محدودیت خطی یا غیرخطی تعیین می‌شود. به منظور

<sup>2</sup> Static Penalty  
<sup>3</sup> Fitness Function

<sup>1</sup> Constraints Handling



## ۷-۵. فرآیند الگوریتم جستجوی همسایگی

بر اساس ساختار تولید همسایگی  $NS_1$  و  $NS_2$  و همچنین ویژگی ریاضی اثبات شده جهت کاهش در فضای جستجوی الگوریتم و افزایش کارایی آن، مراحل الگوریتم جستجوی همسایگی

$$F(X_k) = f(X_k) + \sum_{i=1}^m \beta_i P_i$$

پیشنهادی بصورت زیر نمایش داده می شود.

$X_k$  بوده و  $\beta_i$  یک مقدار ثابت از ضریب جریمه‌ی نقض محدودیت  $i$ -ام را نشان می‌دهد. همچنین  $P_i$  یک متغیر عدد صحیح صفر و یک است که نشان می‌دهد آیا محدودیت  $i$ -ام نقض شده است یا خیر.

$$P_i = \begin{cases} 1 & \text{if constraint } i \text{ is violated} \\ 0 & \text{if constraint } i \text{ is met} \end{cases} \quad (25)$$

آماده‌سازی:

- (۱) ماتریس تخصیص اولیه  $R_0 = [r_{ij}^0]$  را بر اساس مدل  $P_5$  محاسبه و به عنوان ورودی دریافت کنید.
- (۲) مقدار تابع هدف  $R_0$  را از طریق اجرای فرآیند شاخه و کران پیشنهادی در بخش قبل محاسبه و  $f^0$  بنامید؛
- (۳) قرار دهید مقدار اولیه شمارشگر تکرار الگوریتم را  $k = 1$ ؛
- (۴) مجموعه  $E$  را شامل زوج عملیات  $O_{k2}$  و  $O_{k1}$  از زوج سفارشات بالقوه  $[O_i, O_j]$  و  $[O_i, O_k]$  که دارای ویژگی (۱) هستند، تشکیل دهید.

قدم‌های اصلی:

تا زمانی که  $k \leq k_{max}$  مراحل زیر را تکرار کنید:

- (۱) بردار تصادفی  $r = (d_1, d_2)$  را از مجموعه  $E'$  تولید نمائید. سپس ساختار تولید همسایگی  $NS_1$  را روی ماتریس  $R$  اعمال نمائید تا تخصیص  $R^k$  حاصل شود. سپس مجموعه  $E$  را بروز نمائید.
- (۲) مقدار تابع هدف ماتریس تخصیص جدید را برابر  $f^k$  قرار دهید.
- (۳) قرار دهید  $p = 1$  و  $F = f^k$ ،  $k = k + 1$
- (۴) تا زمانی که  $p \leq p_{max}$  مراحل زیر را تکرار نمائید.

(۱-۴) بردار تصادفی  $r_1$  و  $r_2$  را از مجموعه  $E'$  تولید نمائید. ساختار تولید همسایگی  $NS_2$  را روی ماتریس  $R^k$  اعمال نمائید تا ماتریس

تخصیص  $T$  حاصل شود.

(۲-۴) مقدار تابع هدف ماتریس  $T$  را محاسبه نموده و قرار دهید در  $F$ ؛

(۳-۴) قرار دهید  $p = p + 1$

(۴-۴) اگر  $F < f^k$ ، ماتریس تخصیص  $R^k$  را با  $T$  و مقدار  $f^k$  را با  $F$  جایگزین نمائید.

یکی از پارامترهای اساسی در طراحی هر الگوریتم شاخه و کران، میزان عملکرد کران‌های بکار گرفته شده در آن الگوریتم است. در همین راستا در این بخش به ارزیابی عملکرد سه کران پائین پیشنهادی در بخش (۴) می‌پردازیم. مهم‌ترین پارامتری که در ارزیابی عملکرد هر کران مورد بررسی قرار می‌گیرد، میزان محکم بودن<sup>۱</sup> آن کران است. محکم بودن شاخصی است که میزان توانایی کران را در برآورد جواب‌های هر چه نزدیکتر به جواب بهینه نمایش می‌دهد. هر چه این فاصله کمتر باشد، منجر به کاهش تعداد گره‌هایی که در فرآیند شاخه و کران بررسی (پیمایش) می‌شوند شده و در نتیجه منجر به افزایش کارایی الگوریتم می‌گردد. اما پارامتر دیگری که در ارزیابی عملکرد یک کران مورد استفاده قرار می‌گیرد و مربوط به میزان کارایی کران است، زمان محاسبات (اجرا<sup>۲</sup>) است. ممکن است یک کران از نظر دقت (محکم بودن) کران مناسبی بوده و منجر به کاهش تعداد

همانطوریکه در الگوریتم مشخص است، بکارگیری ویژگی (۱) که در بخش قبل اثبات شد، منجر به کاهش فضای جواب از مجموعه‌ی کل فضای جواب به مجموعه  $E$  شده و در نتیجه زمان حل و کارایی الگوریتم افزایش می‌یابد. ویژگی بکار گرفته شده منجر به حذف قسمتی از فضای جواب که منجر به جواب بهینه نخواهد شد، شده و از این طریق عملکرد الگوریتم بهبود می‌یابد.

## ۸. نتایج محاسباتی

در این بخش، تعدادی آزمایش به منظور ارزیابی طرح‌های مختلف الگوریتم شاخه و کران پیشنهادی و همچنین بررسی دقت کران-های پائین و کران بالای توسعه داده شده، طراحی می‌شود. به این منظور الگوریتم‌های طراحی شده، در محیط برنامه‌نویسی MATLAB, 2008a کدنویسی شده و با استفاده از یک کامپیوتر شخصی اجرا شدند. در ادامه به بررسی بیشتر نتایج و تحلیل آن‌ها می‌پردازیم.

## ۸-۱. مقایسه عملکرد کران‌های پائین

<sup>1</sup> Tightness<sup>2</sup> Computational/Implementation Time

متغیرهای ۱-۰ (آزادسازی برنامه ریزی خطی) هستند، می پردازیم. به این منظور تعدادی مثال عددی با ابعاد در نظر گرفته شده و میزان و زمان محاسبات کرانها (متوسط و کمترین و بیشترین) با استراتژی DFS در ۵ نمونه تولید شده از هر مثال در جداول (۲) تا (۷) گزارش شده اند.

جستجوهای انجام شده در فرآیند شاخه و کران شود ولی زمان اجرای طولانی آن منجر شود که جستجوی همان تعداد کم گره های ناشی از محکم بودن کران، زمانبر شده و در نهایت کارایی الگوریتم شاخه و کران را کاهش دهد. در همین راستا، در ادامه به بررسی کرانهای  $LB_1$ ،  $LB_2$  و  $LB_3$  که به ترتیب بر اساس زمان بیکاری زوج سفارشات، آزادسازی محدودیت، و آزادسازی

جدول ۲. متوسط کرانهای پائین محاسبه شده در دو حالت  $|seq| = 0.4n$  و  $|seq| = 0.6n$

شماره (ID)	تعداد سفارش (n)	$ seq  = 0.4n$			$ seq  = 0.6n$		
		$LB_3$	$LB_2$	$LB_1$	$LB_3$	$LB_2$	$LB_1$
۱	۵	۱۱۷,۸	۸۶,۴	۶۵,۰۵۷	۱۳۱,۴	۱۰۷	۹۹,۹۶۵۰۶
۲	۱۰	۱۱۸,۳۶۵۸۸	۱۱۶,۹۴	۱۰۲,۸۴۸۵۲	۱۳۰,۶۴۷۱۶	۱۸۱,۲۰	۱۳۴,۲۵۵۲۴
۳	۲۰	۱۰۱,۱۷۲۲۸	۴۵,۲۰	۵۸,۴۶۶۶	۱۴۵,۷۲۱۱۶	۱۱۱,۶۷	۱۲۴,۱۴۰۹۴
۴	۳۰	۲۵۴,۳۳۵۵۸	۱۵۷,۲۳	۹۹,۲۹۰۰۴	۲۹۹,۵۲۱۸۸	۲۰۷,۱۰	۲۲۴,۷۱۷۹۸
۵	۵۰	۱۹۷,۴۹۷۸۸	۱۷۶,۹۹	۱۵۸,۰۳۴۱۶	۲۹۰,۳۲۰۲	۲۵۷,۱۹	۲۶۲,۷۷۷۶۲
متوسط		۱۵۷,۸۳۴	۱۱۶,۵۵۲	۹۶,۷۳۹	۱۹۹,۵۲۲	۱۷۲,۸۳۲	۱۶۹,۱۷۱

جدول ۳. کمترین مقدار کرانهای پائین محاسبه شده در دو حالت  $|seq| = 0.4n$  و  $|seq| = 0.6n$

شماره (ID)	تعداد سفارش (n)	$ seq  = 0.4n$			$ seq  = 0.6n$		
		$LB_3$	$LB_2$	$LB_1$	$LB_3$	$LB_2$	$LB_1$
۱	۵	۲۸	۲,۰۰	۶,۳۲۲	۹۲	۳۷,۰۰	۳۹,۹۲۸۱
۲	۱۰	۲۴,۳۴۲۹	۱۶,۹۰	۲۰,۷۳۷۲	۳۴,۸۵۱۸	۳۸,۶۲	۳۴,۹۱۱۸
۳	۲۰	۵۷,۱۰۷۲	۳۰,۵۰	۵۰,۹۵۸۲	۶۷,۷۷۴۶	۶۷,۷۷	۹۶,۰۲۴۵
۴	۳۰	۱۷۱,۴۲۳۷	۷۸,۷۴	۵۰,۳۸۲	۱۸۱,۴۸۳۹	۱۱۳,۲۴	۱۰۵,۰۹۵
۵	۵۰	۱۲۵,۲۹۶۹	۷۹,۵۲	۷۹,۸۱۶۱	۲۰۴,۲۸۶	۱۸۱,۱۸	۱۸۲,۷۷۸
متوسط		۸۱,۲۳۴	۴۱,۵۳۲	۴۱,۶۴۳	۱۱۶,۰۷۹	۸۷,۵۶۲	۹۱,۷۴۸

جدول ۴. بیشترین مقدار کرانهای پائین محاسبه شده در دو حالت  $|seq| = 0.4n$  و  $|seq| = 0.6n$

شماره (ID)	تعداد سفارش (n)	$ seq  = 0.4n$			$ seq  = 0.6n$		
		$LB_3$	$LB_2$	$LB_1$	$LB_3$	$LB_2$	$LB_1$
۱	۵	۱۹۶	۲۶۲,۰۰	۸۶,۳۷۴۷	۱۹۶	۱۴۸,۰۰	۱۴۸,۰۲۰۸
۲	۱۰	۱۷۷,۶۷۸۵	۱۵۲,۳۸	۱۴۰,۳۰۱۵	۱۷۷,۶۷۸۵	۲۳۵,۱۴	۱۵۸,۶۱۶۵
۳	۲۰	۱۲۴,۴۸۹۹	۵۵,۶۱	۷۵,۹۵۶۷	۲۴۲,۸۷۹۸	۱۵۰,۶۹	۱۴۶,۲۲۱۶
۴	۳۰	۳۲۳,۷۵۳۴	۳۱۳,۲۰	۱۳۴,۳۸۴۳	۳۸۰,۵۹۳۲	۲۹۲,۹۴	۴۱۸,۱۶۷۱
۵	۵۰	۳۸۰,۲۴۲۳	۲۶۴,۳۹	۲۷۹,۲۲۰۹	۴۰۶,۰۲۱۲	۴۰۴,۷۶	۴۱۸,۵۸۲۲
متوسط		۲۴۰,۴۳۳	۲۰۹,۵۱۶	۱۴۳,۲۴۸	۲۸۰,۶۳۵	۲۴۶,۳۰۶	۲۵۷,۹۲۲

جدول ۵. متوسط زمان اجرای کران‌های پائین در دو حالت  $|seq| = 0.4n$  و  $|seq| = 0.6n$ 

$ seq  = 0.6n$			$ seq  = 0.4n$			تعداد سفارش (n)	شماره (ID)
$LB_3$	$LB_2$	$LB_1$	$LB_3$	$LB_2$	$LB_1$		
۰,۲۶۵۲	۰,۰۳	۰,۰۰۰۱۲۴	۰,۰۸۳۱۲	۰,۳۸	۰,۰۰۰۰۸	۵	۱
۰,۰۸۷۴	۰,۰۲	۰,۰۰۰۱۵۶	۰,۷۸۳۸۲	۰,۱۴	۰,۰۰۰۰۸	۱۰	۲
۰,۳۶۰۵	۱۲۹,۸۵	۰,۰۰۰۱۰۸	۲,۵۵۹۰۸	۱۰۶۵,۵۵	۰,۰۰۰۰۹۸	۲۰	۳
۲,۷۷۷۵	۹۰۸,۸۴	۰,۰۰۰۰۹۲	۲۸,۰۱۳۴۴	۶۱۰,۱۹	۰,۰۰۰۰۱	۳۰	۴
۲۴,۳۳۸	۴۸۴,۰۰	۰,۰۰۰۰۱۴	۱۹۱,۹۲۶۱	۴۷۹,۹۳	۰,۰۰۰۰۱۲	۵۰	۵
۵,۵۶۵۷	۳۰۴,۵۴۸	۰,۰۰۰۰۱۲۴	۴۴,۶۷۳	۴۳۱,۲۴	۰,۰۰۰۰۹۵		متوسط

جدول ۶. کمترین زمان اجرای کران‌های پائین در دو حالت  $|seq| = 0.4n$  و  $|seq| = 0.6n$ 

$ seq  = 0.6n$			$ seq  = 0.4n$			تعداد سفارش (n)	شماره (ID)
$LB_3$	$LB_2$	$LB_1$	$LB_3$	$LB_2$	$LB_1$		
۰,۰۴۳۱	۰,۰۱	۰,۰۰۰۰۰۸	۰,۰۶۲۸	۰,۰۸	۰,۰۰۰۰۰۷	۵	۱
۰,۰۴۱۵	۰,۰۱	۰,۰۰۰۰۰۸	۰,۰۵۴۱	۰,۱۲	۰,۰۰۰۰۰۸	۱۰	۲
۰,۲۹۱۲	۰,۴۳	۰,۰۰۰۰۰۸	۱,۶۶۳۶	۱۹۰,۱۹	۰,۰۰۰۰۰۹	۲۰	۳
۱,۹۲۷۴	۰,۹۹	۰,۰۰۰۰۰۹	۱۴,۲۰۸۲	۲۶۳,۲۴	۰,۰۰۰۰۰۱	۳۰	۴
۱۳,۵۱۴	۴۰۵,۲۱	۰,۰۰۰۰۰۱	۱۳۸,۷۶۰۲	۴۰۷,۰۱	۰,۰۰۰۰۰۱	۵۰	۵
۳,۱۶۳	۸۱,۳۳۰	۰,۰۰۰۰۰۸۶	۳۰,۹۵۰	۲۱۰,۱۶۶	۰,۰۰۰۰۰۸۸		متوسط

جدول ۷. بیشترین زمان اجرای کران‌های پائین در دو حالت  $|seq| = 0.4n$  و  $|seq| = 0.6n$ 

$ seq  = 0.6n$			$ seq  = 0.4n$			تعداد سفارش (n)	شماره (ID)
$LB_3$	$LB_2$	$LB_1$	$LB_3$	$LB_2$	$LB_1$		
۰,۹۰۰۴	۰,۰۷	۰,۰۰۰۰۳	۰,۱۳۴۸	۱,۲۱	۰,۰۰۰۰۰۹	۵	۱
۰,۱۰۵۱	۰,۰۳	۰,۰۰۰۰۲	۰,۹۸۶۹	۰,۱۵	۰,۰۰۰۰۱	۱۰	۲
۰,۵۳۴۳	۶۴۴,۱۷	۰,۰۰۰۰۲	۳,۳۷۸۴	۱۸۳۴,۳۵	۰,۰۰۰۰۱	۲۰	۳
۵,۹۵۷۷	۲۶۰۶,۲۸	۰,۰۰۰۰۱	۸۲,۳۲۶۷	۸۸۵,۶۵	۰,۰۰۰۰۱	۳۰	۴
۶۳,۷۶۱	۵۵۰,۹۷	۰,۰۰۰۰۲	۳۵۶,۴۹۶۳	۵۷۲,۱۶	۰,۰۰۰۰۲	۵۰	۵
۱۴,۲۵۲	۷۶۰,۳۰۴	۰,۰۰۰۰۲	۸۸,۶۶۵	۶۵۸,۷۰۴	۰,۰۰۰۰۱۲		متوسط

به  $LB_2$  و  $LB_3$  داشته و همچنین  $LB_2$  نسبت به  $LB_3$  از زمان محاسباتی کمتری برخوردار است. در نتیجه مسلماً بکارگیری  $LB_1$  در فرآیند شاخه و کران منجر به کاهش تعداد گره‌های پیمایش شده و همچنین کاهش زمان اجرای الگوریتم و افزایش کارایی آن خواهد شد.

همانطوریکه مشاهده می‌شود، از لحاظ محکم بودن،  $LB_1$  دقت بیشتری نسبت به  $LB_2$  و  $LB_3$  داشته و همچنین  $LB_2$  نسبت به  $LB_3$  از دقت بیشتری برخوردار است (شاخص دقت برای کران پائین، میزان نزدیک بودن کران به مقدار بهینه، یعنی بیشتر بودن کران است). علاوه بر این  $LB_1$  زمان اجرای بسیار کمتری نسبت

$$R_i = \begin{cases} M_2 \rightarrow M_1 & \text{if } R(O_{i1}) > R(O_{i2}) \\ M_1 \rightarrow M_2 & \text{if } R(O_{i1}) < R(O_{i2}) \end{cases}$$

که در آن،  $R_i$  توالی سفارش  $i$ -ام در مثال اصلاح شده، و  $R$  ماتریس توالی پردازش سفارشات در مثال‌های استاندارد La01 تا La05 است.

برای هر کدام از مثال‌ها، فرآیند تعیین زوج سفارشات با استفاده از مدل تخصیص متقارن ( $P_2$ )، اجرا شده و تعداد بهینه‌ی زوج سفارشات  $|OP|$  مشخص گردید. جداول (۸) تا (۱۰) تعداد کل گره‌ها در صورت شمارش کامل فضای جواب  $T_T$ ، زمان اجرای الگوریتم شاخه و کران  $CPU Time$  (بر حسب ثانیه)، تعداد گره‌های پیمایش شده در الگوریتم  $T_G$ ، و تعداد گره‌های هرس شده (به عمق رسیده) در آن  $T_p$  را به ترتیب برای الگوریتم شاخه و کرانی که از کران‌های پائین  $LB_1$ ،  $LB_2$  و  $LB_3$  استفاده می‌نماید، نمایش می‌دهند.

## ۲-۸. مقایسه عملکرد استراتژی‌های DFS و BFS بر الگوریتم شاخه و کران

همانطوریکه قبلاً بیان شد، یکی از پارامترهای اساسی که بر عملکرد الگوریتم شاخه و کران تأثیرگذار است، توالی پیمایش گره‌ها در فرآیند شاخه‌زنی است، که تحت عنوان استراتژی جستجو مطرح می‌شود. در این بخش، به بررسی و تحلیل عملکرد دو استراتژی عمق اول و سطح اول جهت جستجو در فرآیند شاخه و کران می‌پردازیم. در این راستا، تعدادی مثال از طریق اعمال تغییراتی در مسائل استاندارد موجود در ادبیات بکار گرفته شده‌اند. مسائل La01 تا La05 جزء مسائل استاندارد مسأله زمانبندی تولید در حالت  $m > 2$  ماشین هستند. در راستای سفارشی‌سازی این نوع از مسائل جهت مسأله حاضر، تنها دو ماشین اول از مسائل استاندارد در نظر گرفته شده تا زمان‌های پردازش تعیین شوند، و جهت تعیین مسیر تولید سفارشات، از قاعده‌ی زیر استفاده شده است.

### جدول ۸. نتایج اجرای الگوریتم شاخه و کران بر اساس استراتژی جستجوی عمق اول و سطح اول با استفاده از $LB_1$

مسأله استاندارد	$ OP $	$T_T$	استراتژی عمق اول (DFS)			استراتژی سطح اول (BFS)		
			$T_p$	$T_G$	$CPU Time$	$T_p$	$T_G$	$CPU Time$
La01	۵	۲۰۶	۲۶	۴۰	۰,۰۰۱۴	۲۴	۳۹	۰,۰۰۱۳
La02	۸	۶۹۲۸۱	۲۸۳	۴۰۶	۰,۰۱۲۶	۲۷۱	۵۶۲	۰,۰۱۹۶
La03	۸	۶۹۲۸۱	۱۹۵۴	۲۷۹۵	۰,۰۹۵۲	۲۱۸۷	۳۱۲۷	۰,۱۰۹۹
La04	۷	۸۶۶۰	۱۱۰۶	۱۷۵۲	۰,۰۳۵۲	۱۲۸۳	۲۰۵۵	۰,۰۴۶۴
La05	۵	۲۰۶	۵	۵	۰,۰۰۵۴	۵	۶	۰,۰۳۳۲
متوسط		۲۹۵۲۶,۸	۶۷۴,۸	۹۹۹,۶	۰,۰۲۹۹۶	۷۵۴	۱۱۵۷,۸	۰,۰۴۲۰۸

### جدول ۹. نتایج اجرای الگوریتم شاخه و کران بر اساس استراتژی جستجوی عمق اول و سطح اول با استفاده از $LB_2$

مسأله استاندارد	$ OP $	$T_T$	استراتژی عمق اول (DFS)			استراتژی سطح اول (BFS)		
			$T_p$	$T_G$	$CPU Time$	$T_p$	$T_G$	$CPU Time$
La01	۵	۲۰۶	۱۵	۲۱	۱۷,۱۰۸۵	۲۳	۳۲	۱۳,۲۹۴۲
La02	۸	۶۹۲۸۱	۵۷۷	۷۴۰	۶۸,۰۰۰۷	۲۷۱	۴۸۸	۱۱۱,۶۲۸۰
La03	۸	۶۹۲۸۱	۳۱۱	۴۱۷	۱۰۸,۲۰۲۴	۲۶۲	۴۱۵	۱۲۰,۴۸۳۵
La04	۷	۸۶۶۰	۵۴۸	۸۵۹	۱۵,۸۷۷۶	۶۲۱	۹۹۲	۵۰۱,۷۱۸۷
La05	۵	۲۰۶	۲۷۸۳	۴۴۷۷	۲۹,۲۴۸۰	۳۲۷۲	۵۲۸۱	۳۴,۰۱۷۷
متوسط		۲۹۵۲۶,۸	۸۴۶,۸	۱۳۰۲,۸	۴۷,۶۸۷۴	۸۸۹,۸	۱۴۴۱,۶	۱۵۶,۲۲۸

جدول ۱۰. نتایج اجرای الگوریتم شاخه و کران بر اساس استراتژی جستجوی عمق اول و سطح اول با استفاده از  $LB_3$ 

مسأله استاندارد	OPI	$T_t$	استراتژی عمق اول (DFS)			استراتژی سطح اول (BFS)		
			$T_p$	$T_g$	CPU Time	$T_p$	$T_g$	CPU Time
La01	۵	۲۰۶	۲۵	۳۹	۰,۱۷۲۳	۳۵	۵۴	۰,۰۰۱۳
La02	۸	۶۹۲۸۱	۱۰۴۲	۱۴۲۷	۵۷,۸۰۲۶	۶۳۹	۱۰۲۷	۵۷,۷۱۴۳
La03	۸	۶۹۲۸۱	۵۴۶۴	۷۷۵۷	۲۶۷,۶۶۵۱	۵۷۷۰	۸۳۴۷	۲۹۶,۵۶۷۰
La04	۷	۸۶۶۰	۲۰۱۲	۳۲۰۴	۷۳,۸۸۰۳	۲۲۴۷	۳۵۹۹	۸۹,۱۱۳۶
La05	۵	۲۰۶	۹۶	۱۶۵	۴,۰۴۷۹	۱۲۰	۲۰۶	۴,۱۳۸۲
متوسط		۲۹۵۲۶,۸	۱۷۲۷,۸	۲۵۱۸,۴	۸۰,۷۱۳۶	۱۷۶۲,۲	۲۶۴۶,۶	۸۹,۵۰۶۹

روش پیشنهادی، از جواب حاصل از نرم افزار تجاری LINGO8 که یک از نرم افزارهای بهینه ساز تجاری محسوب می شود استفاده شود. در این راستا، مثال هایی با ابعاد ۵، ۱۰، ۱۵، ۲۰ و ۳۰ که در بخش های قبلی مورد استفاده قرار گرفتند مورد اصلاح قرار گرفتند. پارامترهایی همچون ضریب زاویه هزینه  $\beta_{ij}$ ، کران بالا و پائین زمان پردازش سفارشات  $U_{ij}$  و  $L_{ij}$ ، موعد مقرر تحویل سفارشات  $d_i$ ، حداکثر میزان بودجه در دسترس  $B$  بطور تصادفی تولید و به مسائل قبلی اضافه شدند. لازم به ذکر است که در این قسمت از بهترین طراحی الگوریتم شاخه و کران یعنی  $LB = LB_1$  و استراتژی جستجوی DFS که در قسمت های قبلی نشان داده شد که دارای بیشترین کارایی هستند، در فرآیند جستجوی همسایگی استفاده شده است.

همانطوریکه نتایج نشان می دهد، استراتژی جستجوی عمق اول، در هر سه کران پائین، چه از لحاظ زمان محاسبات و چه از لحاظ تعداد گره های درخت شاخه و کران، کارایی بهتری نسبت به استراتژی سطح اول دارد.

### ۸-۳. ارزیابی عملکرد الگوریتم جستجوی همسایگی

در این قسمت، نتیجه حاصل از اجرای الگوریتم پیشنهادی جهت حل مدل مسأله برنامه ریزی ظرفیت و زمان بندی تولید یکپارچه ( $P_2$ )، مورد ارزیابی و مقایسه قرار می گیرد. همانطوریکه قبلاً مطرح شد، بر اساس اطلاعات نویسنده، مدل پیشنهادی برای اولین بار در ادبیات مطرح شده و عدم وجود نمونه مشابه جهت تعیین دقت و کارایی آن در ادبیات منجر شد که جهت ارزیابی

جدول ۱۱. نتایج محاسباتی عملکرد الگوریتم NSA

NSA			LINGO		تعداد سفارش (n)	شماره (ID)
CPU Time (s)	Ave.	Best	CPU Time	Optimal		
0.3524	40.6	35	00:06:52	33	۵	۱
2.2706	46.6	40	00:16:09	39	۱۰	۲
41.21	125.2	116	00:45:31	113	۱۵	۳
119.54	256.8	241	01:35:54	239	۲۰	۴
267.91	309.1	291	02:49:27	287	۳۰	۵

به جواب رسیده است. همچنین همانطوریکه ملاحظه می شود اختلاف جواب حاصل از NSA با جواب بهینه ناچیز است، در صورتیکه زمان اجرای LINGO به مراتب بیشتر بوده که استفاده از آن را در مسائل زمان بندی غیرممکن می نماید.

همانطوریکه ملاحظه می شود، LINGO تنها در دو مورد در زمان کمتر از ۲۰ دقیقه به جواب رسیده است که در این دو مورد الگوریتم جستجوی همسایگی NSA در زمان کمتر از ۵ ثانیه جواب پایان یافته در حالیکه LINGO بطور متوسط در ۱۱ دقیقه

## ۹ نتیجه گیری و پیشنهادات

در این تحقیق، دو مسأله از دو سطح تاکتیکی و عملیاتی زنجیره تأمین، یعنی مسأله برنامه ریزی ظرفیت تولید و مسأله زمانبندی سفارشات تولیدی در یک محیط دو ماشینه، بطور یکپارچه مورد توجه قرار گرفتند. در راستای تحقق نیازمندی های مشتریان و ارتقای سطح خدمت دهی به آنان، هدف مدل برآورده سازی موعده تحویل تمامی سفارشات است، به نحوی که میزان ظرفیت تخصیص داده شده از میزان در دسترس تجاوز نکرده، و حداکثر زمان تکمیل سفارشات کمینه شود. امکان تخصیص ظرفیت اضافی به عملیات منجر به کاهش زمان پردازش آن ها شده و در نتیجه مدل پیشنهادی تحت عنوان مدل زمانبندی با «زمان های پردازش وابسته به منابع»، زمان های پردازش را به عنوان متغیر تصمیم در نظر می گیرد.

پس از مدل سازی رویکرد پیشنهادی در غالب مسأله برنامه ریزی عدد صحیح مختلط، نشان داده شد، که مدل با تابع هدف حداکثر زمان تکمیل سفارشات با مدل با تابع هدف کل زمان بیکاری ماشین ها برابر (معادل) است. لذا یک رویکرد حل تحلیلی مبتنی بر تجزیه مسأله جهت کمینه سازی کل زمان بیکاری ماشین ها پیشنهاد شد. در این رویکرد، جهت زمانبندی سفارشات، مفهومی تحت عنوان زوج سفارشات تعریف و الگوریتمی جهت تعیین زوج سفارشات بهینه با حداقل سهم زوج سفارشات از زمان بیکاری ماشین ها، پیشنهاد شد. سپس یک الگوریتم شاخه و کران جهت تعیین توالی بهینه زوج سفارشات طراحی شد که در آن از یک کران پائین محکم پیشنهادی جهت ارزیابی گره های (زیرمسائل) درخت تصمیم استفاده شد. علاوه بر این از دو استراتژی جستجوی عمق اول (DFS) و سطح اول (BFS) جهت پیمایش درخت تصمیم استفاده شد و عملکرد هر کدام بطور جداگانه ارزیابی شد.

همچنین چندین ویژگی ریاضی جهت کاهش فضای مسأله و افزایش کارایی الگوریتم شاخه و کران اثبات شد و به عنوان قواعد چیرگی جواب ها در فرآیند جستجو بکار گرفته شد. برای زیرمسأله برنامه ریزی ظرفیت نیز، یک الگوریتم جستجوی همسایگی (NSA) طراحی و با استفاده از ویژگی ساختاری این زیرمسأله که اثبات شد، عملکرد آن بهبود داده شد.

به عنوان جهت گیری های آتی جهت ادامه تحقیق می توان به ارائه ی رویکردهای حل دقیق تری در بخش برنامه ریزی ظرفیت اشاره کرد، چرا که رویکرد پیشنهادی در بخش زمانبندی (شاخه و کران) بهینه یاب بوده و لیکن رویکرد کلی مبتنی بر جستجوی همسایگی قابلیت بهبود دارد. همچنین اثبات ویژگی های ریاضی جهت بهبود بیشتر در کارایی رویکرد شاخه و کران نیز از دیگر زمینه های تحقیقات آتی محسوب می شود.

## منابع و مراجع

- [1] Asl Farshid, M., Galip Ulsoy, A., *Capacity Management Via Feedback Control in Reconfigurable Manufacturing Systems*. In Proceeding of Japan-USA Symposium on Flexible Automation, Hiroshima, Japan, 2002.
- [2] Baker, K.R., *Introduction to sequencing and scheduling*. NY: Wiley, 1974.
- [3] Brizuela, C.A., Zhao, Y., Sannomiya, N., "No-Wait and Blocking Job-Shops: Challenging Problems for GA's". In IEEE international conference on systems, man, and cybernetics, Tucson, Arizona, USA, 2001.
- [4] Chao-Tang Tseng, Ching-Jong Liaob, *Kuo-Ling Huangb Minimizing Total Tardiness on a Single Machine with Controllable Processing Times*, *Computers & Operations Research* 36, 2009, pp. 1852 – 1858.
- [5] Cheng, T.C.E., Kovalyov, M.Y., Shakhlevich, N.V., *Scheduling with Controllable Release Dates and Processing times: Makespan Minimization*. *European Journal of Operational Research*, 175, 2006a, pp. 751–768.
- [6] Cheng, T.C.E., Kovalyov, M.Y., Shakhlevich, N.V., *Scheduling with Controllable Release Dates and Processing Times: Total Completion Time Minimization*. *European Journal of Operational Research*, 175, 2006b, pp. 769–781.
- [7] Choi, B.-C., Leung, J.Y.-T., Pinedo, M.L., *Complexity of a Scheduling Problem with Controllable Processing Times*. *Operations Research Letters*, 38, 2010, pp. 123–126.
- [8] Christos, T., Maravelias, *Charles Sung, Integration of Production Planning and Scheduling: Overview, challenges and opportunities*, *Computers and Chemical Engineering* 33, 2009, pp. 1919–1930.
- [9] Chuang, S.-P., Hsu, T.-S.H., & Yang, C.-L., *Parallel Work Center Scheduling with Release Dates, Due Dates, and resource-dependent processing times*. *Int J Adv Manuf Technol*, 40, 2009, pp. 193–202.
- [10] Framinan, J.M., Schuster, C., *An Enhanced Timetabling Procedure for the No-Wait Job Shop Problem: A Complete Local Search Approach*. *Computers & Operations Research*, 331, 2006, pp. 1200–1213.
- [11] Gupta, J.N.D., Kruger, K., Lauff, V., Werner, F., Stoskov, Y.N., *Heuristics for Hybrid Job Shops with Controllable Processing Times and Assignable Due Dates*, *Computers & Operations Research*, 29, 2002, 1417–1439.
- [12] Grigoriev, A., Sviridenko, M., Utez, M., *Machine Scheduling with Resource Dependent Processing Times*. *Math. Program., Ser. B*, 110, 2007, pp. 209–228.

- [27] Tseng, C-T.T., Liao, C-J.L., Huang, K-L., *Minimizing Total Tardiness on a Single Machine with Controllable Processing Times*. Computers & Operations Research, 36, 2009, pp. 1852--1858.
- [28] Ventura, J.A., Kim, D., *Parallel Machine Scheduling with Earliness-Tardiness Penalties and Additional Resource Constraints*. Computers & Operations Research, 30, 2003, pp. 1945-1958.
- [29] Wang, L., Zheng, D., *An Effective Hybrid Optimization Strategy for Job-Shop Scheduling Problems*. Computers & Operations Research, 28, 2001, pp. 585-96.
- [30] Wang, J-B., Xia, Z-Q., *Single Machine Scheduling Problems with Controllable Processing Times and Total Absolute Differences Penalties*. European Journal of Operational Research, 177, 638-645.
- [31] Wang, J-B., *Single Machine Scheduling with Common Due Date and Controllable Processing Times*. Applied Mathematics and Computation, 174, 2006, pp. 1245-1254.
- [32] Xu, K., Feng, Z., Jun, K. (In press) *A Tabu-Search Algorithm for Scheduling Jobs with Controllable Processing Times on a Single Machine to Meet Due-Dates*. Computers & Operations Research.
- [33] Yedidsion, L., Shabtay, D., Kaspi, M., *A Bicriteria Approach to Minimize Maximal Lateness and Resource Consumption for Scheduling a Single Machine*. J. Sched., 10, 2007, pp. 341-352.
- [34] Yedidsion, L., Shabtay, D., Korach, E., Kaspi, M., *A Bicriteria Approach to Minimize Number of Tardy Jobs and Resource Consumption in Scheduling a Single machine*, Int. J. Production Economics, 119, 2009, pp. 298-307.
- [35] Zhu, J., Li, X., Wang, Q., *Complete Local Search with Limited Memory Algorithm for No-Wait Job Shops to Minimize Makespan*. European Journal of Operational Research, 198, 2009, pp. 378-386.
- [13] Haejoong Kim, Han-Il Jeong, Jinwoo Park. *Integrated Model for Production Planning and Scheduling in a Supply Chain using Benchmarked Genetic Algorithm* Int J Adv Manuf Technol, 2008, 39:1207-1226.
- [14] Hall, N. G., & Sriskandarajah, C., *A Survey of machine Scheduling Problems with Blocking and no-Wait in Process*. Operations Research, 44, 1996, pp. 510-525.
- [15] Janiak, A., *Minimization of the Blooming mill Standstills-Mathematical Model*, suboptimal algorithms, Mechanika, 8(2), 1989, pp. 37-49.
- [16] Kayan, R.K., Akturk, M.S., *A New Bounding Mechanism for the CNC Machine Scheduling Problem with Controllable Processing Times*, Eur J Oper Res, 167, 2005, pp. 624-643.
- [17] Koulamas, C., Gupta, S., Kyparisis, G.J., *A Unified Analysis for the Single-Machine Scheduling Problem with Controllable and Non-Controllable Variable Job Processing Times*. European Journal of Operational Research, 205, 2010, pp. 479-482.
- [18] Mascis, A., & Pacciarelli, D., *Job-Shop Scheduling with Blocking and No-Wait Constraints*. European Journal of Operational Research, 143, 2002, pp. 498-517.
- [19] Mokhtari, H., Nakhai Kamal Abadi, I., Zegordi, S.H., *Production Capacity Planning and Scheduling in a No-Wait Environment with Controllable Processing Times: An integrated modeling approach*, Expert Systems with Applications, 38, 2011, 12630-12642.
- [20] Pan, J. C.-H., Huang H.C., *A Hybrid Genetic Algorithm for No-Wait Job Shop Scheduling Problems*. Expert Systems with Applications, 36, 2009, pp. 5800-5806.
- [21] Raaymakers, W. H. M., Hoogeveen, J.A., *Scheduling Multipurpose Batch Process Industries with No-Wait Restrictions by Simulated Annealing*. European Journal of Operational Research, 126, 2000, pp. 131-151.
- [22] Reddi, S., Ramamoorthy, C., *A Scheduling Problem*. Operational Research Quarterly, 24, 1973, pp. 441-446.
- [23] Schuster, C., Framinan, J.M., *Approximative Procedures for No-Wait Job Shop Scheduling*. Operations Research Letters, 31, 2003, pp. 308-18.
- [24] Schuster, C., *No-Wait Job Shop Scheduling: Tabu Search and Complexity of Subproblems*. Mathematical Methods of Operations Research, 63, 2006, pp. 473-491.
- [25] Shabtay, D., & Kaspi, M., *Parallel Machine Scheduling with a Convex Resource Consumption Function*. European Journal of Operational Research, 173, 2006, pp. 92-107.
- [26] Shabtay, D., Steiner, G., *A Survey of Scheduling with Controllable Processing Times*, Discrete Applied Mathematics, 155, 2007, pp. 1643-1666.

