

## Designing a New Mathematical Model for Dynamic Cell Formation Problem

Ehsan Karimi, Farid Khoshalhan\* & S.Reza Hejazi

*Ehsan Karimi*, Msc student of Industrial Engineering, Najaf-Abad Free University

*\*Farid Khoshalhan*, Assistance professor of Industrial Engineering, K. N. Toosi University of Technology

*S.Reza Hejazi*, Associate professor of Industrial Engineering, Isfahan University of Technology.

### Keywords

*Dynamic cell formation –  
Intra-cell movement –  
Inter-cell movement –  
Branch and cut –  
Local search*

### ABSTRACT

*In this paper a new nonlinear integer programming model is presented for dynamic cell formation problem (DCF) in cellular manufacturing system (CMS). idea “higher material flow in shorter distance” for formation of cells is introduced. Since the linearized model is NP-hard, a novel approach called local search embedded branch and cut (LSEBC) is introduced. Computational results show the efficiency of the LSEBC in comparison with standard comparable methods.*

© 2011 IUST Publication, IJIEPM. Vol. 22, No. 2, All Rights Reserved

\*  
Corresponding author. Farid Khoshalhan  
Email: [khoshalhan@kntu.ac.ir](mailto:khoshalhan@kntu.ac.ir)



## طراحی یک مدل ریاضی جدید برای مسئله تشکیل سلول پویا بر مبنای تعداد و متوسط طول حرکات درون و بین سلولی

احسان کریمی، فرید خوش الحان\* و سید رضا حجازی

### کلمات کلیدی

تشکیل سلول پویا،  
حرکت درون سلولی،  
حرکت بین سلولی،  
شاخه و برش،  
جستجوی محلی

### چکیده:

در این مقاله یک مدل برنامه ریزی عدد صحیح غیر خطی جدید را برای مسئله تشکیل سلول پویا در سیستم تولید سلولی ارائه می شود. تفاوت بین این مدل و دیگر مدل هایی که پیشتر توسعه یافته اند، در پیاده سازی ایده «جریان مواد بیشتر در فاصله کوتاه تر» در شکل گیری سلول ها است. مدل ارائه شده NP-hard است، یک رهیافت نوین به نام «جستجوی محلی تعبیه شده در شاخه و برش» برای حل آن توسعه داده شده است. نتایج محاسباتی به دست آمده از حل مسائل مختلف، سرعت و کارایی «جستجوی محلی تعبیه شده در شاخه و برش» را در مقایسه با «شاخه و برش استاندارد» نشان می دهد.

### ۱. مقدمه

سیستم تولید سلولی<sup>۲</sup> یک کاربرد مهم از فلسفه تکنولوژی گروهی<sup>۳</sup> است که به منظور استفاده از مشابهت های هندسی و عملیاتی قطعات در طراحی و ساخت، آن ها را در خانواده های قطعات و ماشین های مرتبط را در سلول های ماشینی گروه بندی می کند. سپس هر گروه از ماشین آلات، کار تولید یک خانواده از قطعات را به عهده می گیرد.

گروه بندی قطعات و ماشین آلات باید طوری انجام شود که هزینه حمل و نقل درون و بین سلولی مواد کمینه گردد. اگر دو عملیات

متوالی از یک قطعه درون یک سلول انجام شوند، یک حرکت درون سلولی<sup>۴</sup> و اگر درون دو سلول مختلف انجام شوند، یک حرکت بین سلولی<sup>۵</sup> رخ می دهد. پیاده سازی موفق سیستم تولید سلولی مزایایی همچون کاهش در زمان آماده سازی و ابزار مورد نیاز، تسهیل در تولید و کنترل موجودی، کاهش هزینه حمل و نقل مواد، کاهش کار درون فرآیند، کاهش زمان تدارک، بهبود کیفیت محصول و بهبود انعطاف پذیری کار را به دنبال خواهد داشت [۱، ۲]. طراحی یک سیستم تولید سلولی از چهار فاز تشکیل سلول<sup>۶</sup>، لی اوت گروهی<sup>۷</sup>، زمان بندی گروهی<sup>۸</sup> و تخصیص منابع<sup>۹</sup> تشکیل می شود [۳].

اولین و مهم ترین فاز در طراحی این سیستم ها تشکیل سلول می باشد که در سه دهه اخیر روش های مختلفی برای انجام آن پیشنهاد شده است. از این میان، روش های مبتنی بر تولید بیشتر از بقیه مورد توجه قرار گرفته اند. آن ها را می توان در شش بخش به صورت زیر طبقه بندی کرد:

تاریخ وصول: ۸۹/۳/۲۲

تاریخ تصویب: ۸۹/۸/۵

احسان کریمی، کارشناس ارشد، دانشکده مهندسی صنایع، دانشگاه آزاد اسلامی واحد نجف آباد، [karimi\\_e@iaun.ac.ir](mailto:karimi_e@iaun.ac.ir)

\*نویسنده مسئول مقاله: دکتر فرید خوش الحان، استادیار، دانشکده مهندسی صنایع، دانشگاه صنعتی خواجه نصیرالدین طوسی، [khoshalhan@kntu.ac.ir](mailto:khoshalhan@kntu.ac.ir)

دکتر سیدرضا حجازی، دانشیار، دانشکده مهندسی صنایع، دانشگاه صنعتی اصفهان [rehejazi@cc.iut.ac.ir](mailto:rehejazi@cc.iut.ac.ir)

<sup>۲</sup>. Cellular Manufacturing System

<sup>۳</sup>. Group Technology

<sup>۴</sup>. Intra Cell Movement

<sup>۵</sup>. Inter Cell Movement

<sup>۶</sup>. Cell Formation

<sup>۷</sup>. Group Layout

<sup>۸</sup>. Group Scheduling

<sup>۹</sup>. Resource Allocation

۱- تحلیل خوشه‌ای  
۲- تصمیم‌گیری چند شاخصه  
۳- روش‌های هوش مصنوعی  
۴- روش‌های افراز گراف  
۵- روش‌های ابتکاری و فرا ابتکاری

۶- روش‌های برنامه‌ریزی ریاضی [۴]. روش‌های برنامه‌ریزی ریاضی ابزارهای قدرتمندی هستند که به سادگی می‌توانند پارامترهای طراحی مختلفی را در قالب تابع هدف و محدودیت‌ها فرمول‌بندی کنند، لذا در ادبیات تحقیق مربوط به مسئله تشکیل سلول به طور گسترده‌ای به کار رفته‌اند [۵]. اما به دلیل NP-hard بودن این مسئله، نمی‌توان آن را در ابعاد واقعی در مدت زمانی معقول به صورت بهینه حل کرد.

بنابراین در اغلب موارد از روش‌های فرا ابتکاری نظیر الگوریتم ژنتیک<sup>۱</sup>، شبیه‌سازی تبرید<sup>۲</sup>، جستجوی ممنوعه<sup>۳</sup> و... برای حل آن‌ها استفاده می‌شود. اگر چه در تعداد قابل توجهی از مقالات منتشره، مسئله تشکیل سلول در قالب یک دوره زمانی منفرد با تقاضای قطعی مورد بررسی قرار گرفته است، اما در عمل کاهش چرخه عمر محصولات و تغییرات سریع در طراحی، تقاضا و ترکیب آن‌ها، محیط‌های تولیدی را با شرایط پویا<sup>۴</sup> مواجه کرده است [۶]. ترکیب تولید<sup>۵</sup> به مجموعه‌ای از قطعات گفته می‌شود که باید در هر دوره تولید شوند.

در محیط‌های پویا، می‌توان افق برنامه‌ریزی را به دوره‌های کوچکتری تقسیم کرد که ترکیب و تقاضای محصولات در هر دوره متفاوت است. در نتیجه امکان دارد سلول‌های تشکیل شده در دوره فعلی، برای دوره بعدی بهینه نبوده و پیکره‌بندی مجدد<sup>۶</sup> اجتناب‌ناپذیر باشد. پیکره‌بندی مجدد شامل افزودن ماشین‌های جدید به سلول‌ها، مکان‌یابی مجدد ماشین‌ها و حذف ماشین‌های موجود در سلول‌ها است [۷]. در سال‌های اخیر تحقیقات زیادی در زمینه تولید سلولی پویا<sup>۷</sup> انجام شده است.

رولت و همکارانش (Rheault et al, 1995) مفهوم تولید سلولی پویا را برای مقابله با تغییرات تقاضا در طول افق برنامه‌ریزی مطرح کردند. بدین منظور در هر دوره سلول‌های تولیدی باید مجدداً پیکره‌بندی شوند [۶].

مارکوس و همکارانش (Marcoux et al, 1997) با انجام یک مطالعه ده هفته‌ای نشان دادند که تولید سلولی پویا که توسط رولت و همکارانش پیشنهاد شده، به طور معنی‌داری از تولید سلولی سنتی کم‌هزینه‌تر است [۸]. ویلهلم و همکارانش (Wilhelm et

۱- Genetic Algorithm  
۲- Simulated Annealing  
۳- Tabu Search  
۴- Dynamic Condition  
۵- Product Mix  
۶- Reconfiguration  
۷- Dynamic Cellular Manufacturing

۸- Dynamic Cell Formation Problem  
۹- Decomposition Method  
۱۰- Dynamic Programming  
۱۱- Neural Network  
۱۲- Multi Objective Scatter Search  
۱۳- Mean Field Annealing-Simulated Annealing

مونگ واتانا (Mungvattana, 2000) مسئله تشکیل سلول پویا را با فرض امکان انجام یک عملیات روی بیش از یک ماشین فرمول‌بندی کرد. در ادامه از الگوریتم شبیه‌سازی تبرید برای حل این مدل استفاده شد [۱۳]. توکلی مقدم و همکارانش مدل ارائه شده توسط مونگ واتانا را اصلاح کردند. سپس مدل پیشنهادی به کمک سه روش فرا ابتکاری الگوریتم ژنتیک، شبیه‌سازی تبرید و جستجوی ممنوعه و نیز نرم افزار LINGO حل شد. نتایج به دست آمده، برتری شبیه‌سازی تبرید را نسبت به دیگر روش‌ها ثابت کرد [۱۴].

سعیدی مهرآباد و صفایی مدل مقاله [۱۴] را برای تعیین تعداد بهینه سلول‌ها در هر دوره، توسعه دادند. به دلیل NP-hard بودن این مدل، از رهیافت شبکه عصبی<sup>۱۱</sup> برای حل آن استفاده شد [۱۵]. آرمون باجستانی و همکارانش با افزودن هدف حداقل کردن کل تغییرات بار سلول به مدل مقاله [۱۴]، یک مدل دو هدفه جدید توسعه دادند.

با توجه به NP-hard بودن مدل، روش جستجوی تراکنشی چند هدفه<sup>۱۲</sup> برای یافتن مرز بهینه پارتو به کار رفت [۱۶]. صفایی و همکارانش مدل تشکیل سلول پویا را برای حالتی که هزینه حرکات درون و بین سلولی توأمان در تابع هدف لحاظ شده‌اند، تعمیم دادند و آن را به کمک رهیافت ترکیبی (MFA-SA)<sup>۱۳</sup> حل کردند. نتایج محاسباتی نشان داد که MFA به طور

از آن جایی که در مقالات موجود، هزینه حمل و نقل مواد فقط به تعداد حرکات بین سلولی بستگی دارد، انجام این حرکت بین سلول های (۱ و ۲) یا (۱ و ۳) هزینه یکسانی را به سیستم تولید تحمیل می کند، زیرا در هر دو حالت یک حرکت بین سلولی رخ می دهد. اما در عمل ممکن است فاصله بین سلول های (۱ و ۲) از فاصله بین سلول های (۱ و ۳) کمتر باشد. در نتیجه انجام این حرکت بین دو سلول (۱ و ۲) هزینه حمل و نقل کمتری خواهد داشت.

با این مثال ساده علاوه بر مشخص شدن نقطه ضعف مقالات مرور شده، اهمیت طول حرکات بین سلولی در تشکیل سلول ها نیز مشخص می شود.

این مقاله سعی دارد با دخیل کردن متوسط طول حرکات درون و بین سلولی در هزینه حمل و نقل مواد و نتیجتاً در شکل گیری سلول ها، این نقطه ضعف را برطرف کند. همچنین یک روش نوین برای حل مدل پیشنهادی ارائه می شود. ایده اصلی این روش کاهش پیچیدگی مسئله اصلی به کمک تثبیت برخی از متغیرها در هر تکرار و در نتیجه کاهش زمان حل است. در ادامه این مقاله، بخش دوم به فرمول بندی مسئله می پردازد. در بخش سوم روش حل معرفی می شود. بخش چهارم یک مثال عددی را توضیح می دهد.

نتایج محاسباتی مربوط به حل تست مسئله ها در بخش پنجم آورده شده است. بخش ششم به نتیجه گیری اختصاص دارد. نهایتاً بخش هفتم پیشنهاداتی برای تحقیقات آتی را مطرح کرده است.

## ۲. فرمول بندی مسئله

در این قسمت با در نظر گرفتن فرض های زیر، یک مدل برنامه ریزی عدد صحیح غیر خطی<sup>۱</sup> جدید بر مبنای تعداد و متوسط طول حرکات درون و بین سلولی برای مسئله تشکیل سلول پویا ارائه می شود.

### ۲-۱. فرض های مسئله

- ۱- هر قطعه تعدادی عملیات دارد که باید به ترتیب انجام شوند.
- ۲- هر عملیات از هر قطعه می تواند روی بیش از یک ماشین انجام شود. زمان انجام این عملیات روی ماشین های مختلف، معلوم و قطعی است.
- ۳- تقاضای هر قطعه در هر دوره معلوم و قطعی است.
- ۴- توانایی و ظرفیت زمانی هر ماشین در کل افق برنامه ریزی معلوم و ثابت است.

محسوسی سرعت همگرایی SA را افزایش داده و موجب رسیدن به جواب های نسبتاً بهتری خواهد شد [۷].

صفایی و توکلی مقدم یک مدل ریاضی برای یکپارچه سازی مسئله تشکیل سلول چند دوره ای و برنامه ریزی تولید ارائه دادند. عملکرد این مدل با حل دو مثال عددی کوچک مقیاس اعتبار سنجی شد [۱۷].

دفرشا و چن (Defersha & Chen, 2006) یک مدل ریاضی جامع برای مسئله تشکیل سلول پویا ارائه دادند که در آن امکان پردازش یک عملیات روی یک ماشین با توجه به ابزار مورد نیاز جهت پردازش آن عملیات و ابزارهای در دسترس روی آن ماشین، مشخص می شود.

برای اعتبار سنجی این مدل، یک تست مسئله متوسط مقیاس به کمک نرم افزار LINGO حل و بررسی شد [۱۸]. دفرشا و چن (Defersha & Chen, 2008) یک مدل ریاضی جدید پیشنهاد کردند که یکپارچه سازی تشکیل سلول و اندازه انباشته را با در نظر گرفتن کیفیت محصول بررسی می کند. با توجه به NP-complete بودن مدل پیشنهادی، یک روش جدید تحت عنوان برنامه ریزی خطی تعبیه شده در الگوریتم ژنتیک برای حل آن توسعه یافت [۱۹].

کیون و همکارانش (Kioon et al, 2009) یک مدل ریاضی برای یکپارچه سازی مسئله تشکیل سلول پویا و برنامه ریزی تولید ارائه دادند که مواردی همچون توالی عملیات، پیکره بندی پویای سلول، مسیرهای فرآیند چندگانه، شکستن انباشته، برون سپاری قطعات، حداکثر تعداد ماشین در دسترس و محدودیت اندازه سلول را مد نظر قرار می دهد.

در ادامه مدل مذکور با حداقل تعداد محدودیت ها خطی شد. به منظور اعتبار سنجی این مدل، مسائل مختلفی که داده های آن ها از مقالات مرتبط استخراج شده بود، به کمک نرم افزار CPLEX 9.0.1 حل شد [۲۰].

دل جو و همکارانش نشان دادند که هزینه پیکره بندی مجدد سیستم تولید و نیز هزینه ثابت (خرید یا کرایه) ماشین آلات در تمامی مقالاتی که بر مبنای مدل مونگ واتانا توسعه یافته اند، اشتباه محاسبه شده است. آن ها بعد از اصلاح نقطه ضعف های مقالات مذکور در مدلشان، آن را به کمک یک الگوریتم ژنتیک جدید حل کردند [۲۱].

در تمامی مقالات مرتبط با مسئله تشکیل سلول پویا، هزینه حمل و نقل مواد فقط به تعداد حرکات درون و بین سلولی انجام شده بستگی دارد. در این جا با یک مثال ساده نقطه ضعف این مقالات را بیان می کنیم. فرض کنید سه سلول تولیدی با ابعاد متفاوت داریم و وجود یک حرکت بین سلولی از سلول ۱ تا یکی از دو سلول ۲ یا ۳ اجتناب ناپذیر است.

<sup>۱</sup>. Nonlinear Integer Programming

$D_{cc'} =$  متوسط فاصله بین مکان های کاندید متعلق به سلول  $C$   
اگر  $(c = c')$

$FF_{cc'}: 1$  اگر  $(c \neq c')$  و  $0$  اگر  $(c = c')$

$B_p^{inter}$ : اندازه انباشته برای حرکت بین سلولی قطعه  $p$

$B_p^{intra}$ : اندازه انباشته برای حرکت درون سلولی قطعه  $p$

$\gamma^{inter}$ : هزینه حرکت یک انباشته به اندازه واحد طول بین سلولها

$\gamma^{intra}$ : هزینه حرکت یک انباشته به اندازه واحد طول درون

سلول. به منظور توجه پذیری سیستم تولید سلولی فرض می کنیم که:

$$\left(\gamma^{intra} / B_p^{intra}\right) < \left(\gamma^{inter} / B_p^{inter}\right)$$

$\alpha_m$ : هزینه خرید ماشین  $m$  در هر دوره

$\beta_m$ : هزینه متغیر ماشین  $m$  برای هر واحد زمانی کار

$INS_m$ : هزینه نصب یک واحد ماشین  $m$

$UNS_m$ : هزینه حذف یک واحد ماشین  $m$

$T_m$ : ظرفیت زمانی ماشین  $m$  در هر دوره

$t_{pjm}$ : زمان پردازش عملیات  $j$  ام از قطعه  $p$  روی ماشین  $m$

$FN_{mc}$ : تعداد ماشین  $m$  که در ابتدای دوره برنامه ریزی درون سلول  $C$  وجود دارد

$UB_{ch}$ : تعداد مکان های کاندید متعلق به سلول  $C$  در دوره  $h$

### ۲-۲-۳. متغیرهای تصمیم

$x_{pjmch} = 1$  اگر در دوره  $h$ ،  $j$  امین عملیات قطعه  $p$ ، توسط

ماشین  $m$  درون سلول  $C$  انجام شود و  $0$  در غیر این صورت

$N_{mch} =$  تعداد ماشین نوع  $m$  که در دوره  $h$  درون سلول  $C$  وجود دارد

$KA_{mch} =$  تعداد ماشین نوع  $m$  که در دوره  $h$  به سلول  $C$  اضافه می شود

$KR_{mch} =$  تعداد ماشین نوع  $m$  که در دوره  $h$  از سلول  $C$  حذف می شود

### ۲-۳. مدل ریاضی

با توجه به نمادهای معرفی شده، مدل پیشنهادی به صورت زیر است:

۵- هزینه سرمایه گذاری یا خرید هر ماشین معلوم بوده و مستقل از بار کاری تخصیص یافته به آن است.

۶- هزینه متغیر هر ماشین معلوم بوده و مستقیماً به بارکاری تخصیص یافته به آن بستگی دارد.

۷- هزینه عملیاتی هر ماشین در واحد زمان معلوم است.

۸- قطعات به صورت انباشته ای درون و بین سلولها حرکت می کنند. هزینه حرکت یک انباشته بین سلولها از هزینه حرکت یک انباشته درون هر سلول بیشتر است.

۹- حداکثر تعداد سلولهایی که در هر دوره می توانند تشکیل شوند، معلوم است.

۱۰- مکان های از قبل تعیین شده ای (مکان های کاندید) برای چینش ماشین آلات درون هر سلول، در نظر گرفته شده است. تعداد این مکانها بیانگر حداکثر تعداد ماشین آلات مجاز درون آن سلول است. ضمناً ابعاد سلولها، لزوماً یکسان نیست.

۱۱- هزینه مکانیابی مجدد هر ماشین معلوم بوده و مستقل از فاصله ای است که ماشین برای مکانیابی مجدد طی می کند. زمان لازم برای مکانیابی مجدد، صفر فرض می شود.

۱۲- سفارش معوقه و موجودی بین دورهها صفر بوده و تمامی تقاضاها باید در همان دوره ارضاء شوند.

۱۳- از زمان تنظیم ماشین آلات و احتمال خرابی آن ها صرف نظر می شود.

### ۲-۲. نماد گذاری ها

در این قسمت اندیس ها، پارامترهای ورودی و متغیرهای تصمیم به کار رفته در مدل، نماد گذاری می شوند:

#### ۲-۲-۱. اندیس ها

$p$ : اندیس انواع قطعات ( $p = 1, 2, \dots, P$ )

$j$ : اندیس عملیات قطعه ( $j = 1, 2, \dots, O_p$ )

$m$ : اندیس انواع ماشین ( $m = 1, 2, \dots, M$ )

$C, C'$ : اندیس سلول های تولیدی ( $C, C' = 1, 2, \dots, C$ )

$h$ : اندیس دوره برنامه ریزی ( $h = 1, 2, \dots, H$ )

#### ۲-۲-۲. پارامترهای ورودی

$DE_{ph}$ : تقاضای قطعه  $p$  در دوره  $h$

$PP_{ph}$ : اگر قطعه  $p$  در دوره  $h$  تقاضا داشته باشد و  $0$  در غیر این صورت

$a_{pjm}$ : اگر  $j$  امین عملیات قطعه  $p$ ، بتواند روی ماشین  $m$  انجام شود و  $0$  در غیر این صورت فاصله مرکز سلول  $C$  تا مرکز سلول  $C'$  اگر  $(c \neq c')$

$$\begin{aligned}
 \text{Min } Z = & \sum_{m=1}^M \alpha_m \text{Max} \left\{ 0, \text{Max}_h \left\{ \sum_{c=1}^C (N_{mch} - FN_{mc}) \right\} \right\} + \sum_{h=1}^H \sum_{c=1}^C \sum_{p=1}^P \sum_{j=1}^{O_p} \sum_{m=1}^M \beta_m DE_{ph} t_{pjm} x_{pjmch} + \\
 & \sum_{h=1}^H \sum_{p=1}^P \sum_{j=1}^{(O_p-1)} \sum_{c=1}^C \sum_{c'=1}^C \gamma^{\text{inter}} FF_{cc'} D_{cc'} \left[ \frac{DE_{ph}}{B_p^{\text{inter}}} \right] \left( \sum_{m=1}^M x_{pjmch} \right) \left( \sum_{m=1}^M x_{p(j+1)mc'h} \right) + \\
 & \sum_{h=1}^H \sum_{p=1}^P \sum_{j=1}^{(O_p-1)} \sum_{c=1}^C \sum_{c'=1}^C \gamma^{\text{intra}} (1 - FF_{cc'}) D_{cc'} \left[ \frac{DE_{ph}}{B_p^{\text{intra}}} \right] \left( \sum_{m=1}^M x_{pjmch} \right) \left( \sum_{m=1}^M x_{p(j+1)mc'h} \right) + \\
 & \sum_{h=1}^H \sum_{c=1}^C \sum_{m=1}^M INS_m KA_{mch} + \sum_{h=1}^H \sum_{c=1}^C \sum_{m=1}^M UNS_m KR_{mch}
 \end{aligned} \tag{1}$$

Subject to :

$$\sum_{c=1}^C \sum_{m=1}^M x_{pjmch} = PP_{ph} \quad \forall p, j, h \tag{2}$$

$$x_{pjmch} \leq a_{pjm} \quad \forall p, j, m, c, h \tag{3}$$

$$\sum_{p=1}^P \sum_{j=1}^{O_p} DE_{ph} t_{pjm} x_{pjmch} \leq T_m N_{mch} \quad \forall m, c, h \tag{4}$$

$$N_{mc(h-1)} + KA_{mch} - KR_{mch} = N_{mch} \quad \forall m, c, h \quad h > 1 \tag{5}$$

$$FN_{mc} + KA_{mc1} - KR_{mc1} = N_{mc1} \quad \forall m, c \tag{6}$$

$$\sum_{m=1}^M N_{mch} \leq UB_{ch} \quad \forall c, h \tag{7}$$

$$x_{pjmch}^1 \in \{0, 1\}, N_{mch}, KA_{mch}, KR_{mch} \geq 0 \text{ and Integer.} \tag{8}$$

ثابت، هزینه متغیر، توانایی پردازشی ماشین و ظرفیت زمانی آن انجام خواهد شد [۷]. جمله سوم هزینه حمل و نقل بین سلولی مواد را به ازای تمامی قطعات در کل افق برنامه ریزی محاسبه می کند. در مقالات [۷، ۱۳، ۱۴، ۱۵، ۱۶، ۱۷، ۱۸، ۱۹، ۲۰، ۲۱] این هزینه از حاصل ضرب تعداد حرکات بین سلولی در هزینه هر حرکت به دست می آید. به عبارت دیگر در این مقالات فقط تعداد حرکات بین سلولی کمینه می شود، اما همان طور که در مقدمه نشان دادیم، لزوماً این حرکات به کوتاه ترین فواصل تخصیص نمی یابند.

در این مقاله، هزینه حمل و نقل بین سلولی مواد علاوه بر تعداد حرکات بین سلولی، به متوسط طول آنها نیز بستگی خواهد داشت. بدین منظور باید در هر حرکت بین سلولی علاوه بر مبدأ، مقصد حرکت نیز مشخص شود تا بتوان فاصله بین این دو را کمینه کرد (پیاده سازی ایده جریان مواد بیشتر در فاصله کوتاه

### ۱-۳-۲. تشریح جملات تابع هدف و محدودیت ها

تابع هدف (معادله ۱) شامل شش جمله است. جمله اول هزینه خرید ماشین آلات در کل افق برنامه ریزی را با توجه به تعداد ماشین آلات موجود از هر نوع در هر سلول در ابتدای دوره برنامه ریزی، محاسبه می کند. دل جو و همکارانش نشان دادند که این جمله در مقالات [۷، ۱۳، ۱۴، ۱۵، ۱۶، ۱۷، ۲۰] اشتباه محاسبه شده است. آن ها تعداد واقعی ماشین آلاتی که در کل افق برنامه ریزی خریداری و به سیستم اضافه می شوند، را به صورت جمله اول تابع هدف بیان کردند [۲۱].

این جمله موجب می گردد تا کل تعداد ماشین آلات خریداری شده در افق برنامه ریزی کمینه شود و در نتیجه نرخ بهره برداری از هر ماشین به حداکثر برسد. جمله دوم هزینه عملیاتی ماشین آلات را با توجه به کل بارکاری تخصیص داده شده به آنها محاسبه می کند. در نتیجه انتخاب یک ماشین براساس چهار معیار هزینه

و  $c'$  است. در این حالت حاصلضرب هزینه حرکت یک انباشته بین سلول ها به اندازه واحد طول، در فاصله بین مرکز سلول  $C$  تا مرکز سلول  $c'$ ، در تعداد انباشته های قطعه  $p$  در دوره  $h$ ، به تابع هدف اضافه می شود:

$$\left\{ \begin{array}{l} \sum_{m=1}^M x_{pjmch} = 1 \\ \sum_{m=1}^M x_{p(j+1)mc'h} = 1 \\ FF_{cc'} = 1 \end{array} \right\} \Rightarrow \text{Thrid term} = \gamma^{\text{inter}} \times 1 \times D_{cc'} \times \left[ \frac{DE_{ph}}{B_p^{\text{inter}}} \right] \times (1) \times (1) \quad (9)$$

اگر در دوره  $h$ ، عملیات  $j$  ام از قطعه  $p$ ، درون سلول  $C$  و عملیات  $(j+1)$  ام آن درون سلول  $c'$  انجام شود و ( $c=c'$ ) باشد، یک حرکت درون سلول  $C$  اتفاق می افتد. در این حالت حاصلضرب هزینه حرکت یک انباشته درون سلول به اندازه واحد طول، در متوسط فاصله بین مکان های کاندید درون آن سلول، در تعداد انباشته های قطعه  $p$  در دوره  $h$ ، به تابع هدف اضافه می شود:

$$\left\{ \begin{array}{l} \sum_{m=1}^M x_{pjmch} = 1 \\ \sum_{m=1}^M x_{p(j+1)mc'h} = 1 \\ FF_{cc'} = 0 \end{array} \right\} \Rightarrow \text{Forth term} = \gamma^{\text{intra}} \times (1-0) \times D_{cc'} \times \left[ \frac{DE_{ph}}{B_p^{\text{inter}}} \right] \times (1) \times (1) \quad (10)$$

نصب) داریم که در ( $INS_m$ ) لحاظ می شوند. اگر یک ماشین از سلولی حذف و به محل ماشین های حذف شده (انبار) برود، فقط هزینه حذف و انتقال (هزینه حذف) داریم که در ( $UNS_m$ ) لحاظ می شوند. اگر تعدادی ماشین خریداری و تعدادی از سلولی به سلول دیگر منتقل شده باشند، هزینه پیکره بندی مجدد به صورت زیر است:

$$\left\{ \sum_{h=1}^H \sum_{c=1}^C KA_{mch} = 5, \sum_{h=1}^H \sum_{c=1}^C KR_{mch} = 3 \right\} \Rightarrow \text{Reconfiguration Cost} = 3 \times (INS_m + UNS_m) + 2 \times INS_m \quad (11)$$

قطعه  $p$  در دوره  $h$  تقاضا داشته باشد،  $j$  امین عملیات آن فقط توسط یک ماشین و درون یک سلول انجام شود. معادله ۳ باعث می شود تا  $j$  امین عملیات قطعه  $p$  فقط بتواند به یکی از ماشین هایی که توانایی انجامش را دارند، تخصیص

تر). در مدل پیشنهادی اگر در دوره  $h$ ، عملیات  $j$  ام از قطعه  $p$ ، درون سلول  $C$  و عملیات  $(j+1)$  ام آن درون سلول  $c'$  انجام شود و ( $c \neq c'$ ) باشد، یک حرکت بین سلول های  $C$  و  $c'$  اتفاق می افتد. متوسط طول این حرکت برابر فاصله بین مراکز دو سلول  $C$

جمله چهارم هزینه حمل و نقل درون سلولی مواد را به ازای تمامی قطعات در کل افق برنامه ریزی محاسبه می کند. با در نظر گرفتن همزمان هزینه جریان مواد درون و بین سلولی، اندازه سلول های تشکیل شده متعادل می شود [۷]. در این مقاله متوسط فاصله بین مکان های کاندید در هر سلول نیز در این هزینه دخیل است.

جمله پنجم و ششم در مجموع هزینه پیکره بندی مجدد سلول ها را در خلال دوره های برنامه ریزی محاسبه می کنند. دل جو و همکارانش نشان دادند که این هزینه در مقالات [۷، ۱۳، ۱۴، ۱۵، ۱۶، ۱۷، ۲۰] اشتباه محاسبه شده است. آن ها و همچنین دفرشا و چن، مستقلاً راهکار مناسبی برای غلبه بر این مشکل اندیشیدند [۱۸، ۱۹، ۲۱]. سه حالت امکان پذیر است. اگر یک ماشین جدید خریداری شود، فقط هزینه خرید و حمل (هزینه

در این حالت ۳ ماشین از سلولی حذف و به سلول دیگری منتقل و در آن جا نصب شده اند، لذا هر دو هزینه به تابع هدف اضافه می شود. اما در مورد ۲ ماشین دیگر که خریداری شده اند، فقط هزینه خرید و حمل داریم. معادله ۲ تضمین می کند که اگر

در نهایت با جایگزینی جمله اول تابع هدف با  $\left(\sum_{m=1}^M \alpha_m Pur_m\right)$

و افزودن معادله (۹) به مدل، جمله اول خطی می شود. کیون و همکارانش برای خطی کردن حاصلضرب دو متغیر صفر و یک با حداقل تعداد محدودیت ها، راهکار جدیدی ارائه دادند که در این مقاله از آن استفاده می کنیم [۲۰]. آن ها ثابت کردند که اگر ضریب جملات حاصلضرب در تابع هدفی از نوع کمینه سازی، مثبت باشد، می توان تنها با افزودن یک دسته متغیر صفر و یک جدید و یک دسته محدودیت جدید، حاصلضرب را خطی کرد (اثبات مربوطه در قسمت پیوست آمده است). دسته متغیر صفر و یک جدید  $(y_{pjcc'h})$  به صورت زیر تعریف می شود:

$$y_{pjcc'h} = \left(\sum_{m=1}^M x_{pjmch}\right) \left(\sum_{m=1}^M x_{p(j+1)mc'h}\right) \quad (14)$$

دسته محدودیت جدیدی که باید به مدل اضافه شود، به صورت است:

$$y_{pjcc'h} \geq \left(\sum_{m=1}^M x_{pjmch}\right) + \left(\sum_{m=1}^M x_{p(j+1)mc'h}\right) - 1 \quad \forall p, j, c, c', h \quad (15)$$

با توجه به رابطه تنگاتنگ بین متغیرهای  $(x_{pjmch})$  و  $(y_{pjcc'h})$  به افزودن دسته محدودیت جدیدی نیاز داریم که تضمین کند، متغیر  $(y_{pjcc'h})$  وقتی برابر ۱ خواهد شد که قطعه  $p$  در دوره  $h$  تقاضا داشته باشد  $(PP_{ph} = 1)$ . این محدودیت به صورت زیر است:

$$\sum_{c=1}^C \sum_{c'=1}^C y_{pjcc'h} = PP_{ph} \quad \forall p, j, h \quad (16)$$

با جایگزین کردن متغیر  $(y_{pjcc'h})$  به جای حاصلضرب دو متغیر صفر و یک در جملات سوم و چهارم تابع هدف و نیز افزودن دو معادله (۱۰) و (۱۱) به مدل، این جملات نیز خطی می شوند.

### ۳. جستجوی محلی تعبیه شده در شاخه و برش

نرم افزارهای بهینه سازی برای حل مسائل برنامه ریزی عدد صحیح از روش «شاخه و برش»<sup>۱</sup> استفاده می کنند. اما با افزایش ابعاد تست مسئلهها، این روش کارایی خود را از دست می دهد و نمی تواند در مدت زمانی معقول به جواب بهینه برسد. در این مقاله، روش جدیدی تحت عنوان «جستجوی محلی تعبیه شده در شاخه و برش»<sup>۲</sup> برای رفع این مشکل توسعه یافته

یابد. معادله ۴ از تخصیص فراتر از ظرفیت زمانی ماشین آلات جلوگیری می کند. با وجود این معادله تقاضای قطعات در هر دوره ارضاء خواهد شد. این معادله همچنین تعداد ماشین آلات مورد نیاز از هر نوع در هر دوره را مشخص می کند. معادله ۵ محدودیت بالانس ماشین آلات نام دارد.

بر طبق این معادله تعداد ماشین ها در دوره فعلی برابر است با تعداد ماشین ها در دوره قبل به علاوه تعداد ماشین هایی که اضافه شده اند، منهای تعداد ماشین هایی که حذف شده اند. معادله ۶ نیز بالانس تعداد ماشین آلات در دوره اول را به عهده دارد. معادله ۷ تضمین می کند که کل تعداد ماشین آلات در دوره  $h$  درون سلول  $c$ ، نباید از تعداد مکان های کاندید متعلق به سلول  $c$  در دوره  $h$  تجاوز کند و لذا محدودیت حداکثر اندازه سلول رعایت خواهد شد. معادله ۸ نوع متغیرهای به کار رفته در مدل پیشنهادی را بیان می کند.

### ۲-۳-۲. خطی سازی مدل پیشنهادی

مدل پیشنهادی به دلیل وجود عملگر  $Max$  در جمله اول و حاصلضرب دو متغیر صفر و یک در جملات سوم و چهارم تابع هدف، از نوع برنامه ریزی عدد صحیح غیر خطی است. روش های مختلفی برای خطی کردن ارائه شده است که به روش های ارائه شده اخیر توسط دلجو و همکارانش [۲۱] و هران و همکارانش [۲۲] می توان اشاره کرد. روش ارائه شده متناسب با ساختار مدل در این مقاله به شرح زیر است. ابتدا دو دسته متغیر عدد صحیح جدید به صورت زیر تعریف می شوند:

$Maxim_{mh}$ : حداکثر تعداد ماشین  $m$  که دوره  $h$  به سیستم تولید وارد می شوند

$Pur_m$ : حداکثر تعداد ماشین  $m$  که کل افق برنامه ریزی به سیستم تولید وارد می شوند

دو دسته محدودیت جدیدی که باید به مدل اضافه شوند، به صورت هستند:

$$Pur_m \geq Maxim_{mh} \quad \forall m, h$$

$$Maxim_{mh} \geq \sum_{c=1}^C (N_{mch} - FN_{mc}) \quad \forall m, h \quad (12)$$

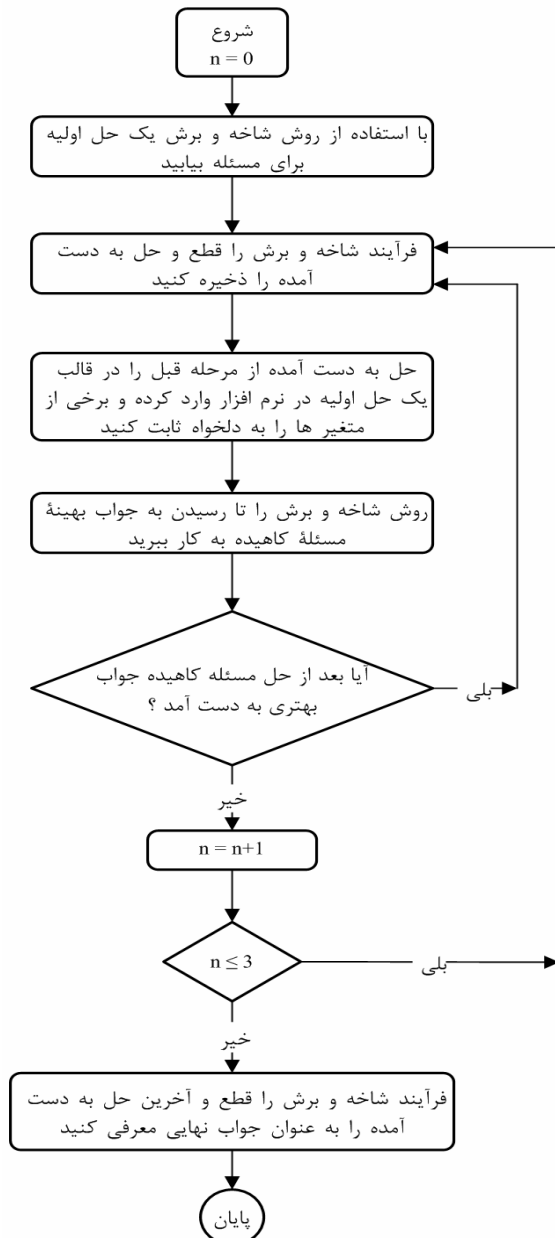
با نگاهی ساده می توان دریافت که دو محدودیت بالا قابل ادغام هستند. در نتیجه تنها با یک دسته متغیر عدد صحیح جدید و یک دسته محدودیت جدید نیز می توان جمله اول را خطی کرد:

$$Pur_m \geq \sum_{c=1}^C (N_{mch} - FN_{mc}) \quad \forall m, h \quad (13)$$

<sup>1</sup>. Branch and Cut (BC)

<sup>2</sup>. Local Search Embedded Branch and Cut (LSEBC)





شکل ۱. فلوجارت روش جستجوی محلی تعبیه شده در شاخه و برش

جدول ۱. پارامترهای ورودی

نوع ماشین	$T_m$	$\alpha_m$	$\beta_m$	$INS_m$	$UNS_m$
۱	۱۰۰۰	۱۲۰۰	۷	۳۰۰	۳۰۰
۲	۱۰۰۰	۱۶۰۰	۹	۴۰۰	۴۰۰
۳	۱۰۰۰	۱۸۰۰	۱۰	۱۵۰	۱۵۰
۴	۱۰۰۰	۱۵۰۰	۹	۳۷۵	۳۷۵
۵	۱۰۰۰	۱۴۰۰	۸	۳۵۰	۳۵۰
۶	۱۰۰۰	۱۲۰۰	۷	۲۵۰	۲۵۰
۷	۱۰۰۰	۱۷۰۰	۹	۴۲۵	۴۲۵
۸	۱۰۰۰	۱۳۰۰	۸	۳۲۵	۳۲۵
۹	۱۰۰۰	۱۵۰۰	۱۰	۳۷۵	۳۷۵
۱۰	۱۰۰۰	۱۸۰۰	۸	۴۵۰	۴۵۰

است. بر طبق این روش ابتدا مسئله را به کمک روش شاخه و برش تا رسیدن به یک جواب اولیه ادامه می دهیم. سپس فرآیند شاخه و برش را قطع و حل به دست آمده را ذخیره می کنیم. در ادامه، این حل اولیه را به عنوان نقطه شروع در نرم افزار وارد کرده و به دلخواه برخی از متغیرها را در مقدار فعلی شان ثابت نگه می داریم. حال مسئله کاهیده (مسئله ای که تعداد متغیرها و محدودیت های آن کاهش یافته است) را مجدداً با روش شاخه و برش حل می کنیم. سپس حل به دست آمده در این تکرار را ذخیره کرده و از آن به عنوان یک نقطه شروع برای تکرار بعدی استفاده می کنیم.

با انجام این کار پیچیدگی مسئله اصلی کاهش می یابد و دستیابی به حل هایی که در متغیرهای ثابت شده با حل فعلی مشترک هستند، در مدت زمان کوتاه تری امکان پذیر می شود. به عبارت دیگر این روش با ثابت کردن برخی از متغیرها و در نتیجه زائد شدن برخی از محدودیت ها در هر تکرار، فضای حل مسئله را محدود می کند و باعث می شود تا الگوریتم شاخه و برش بتواند در مدت زمانی محدود کارا تر عمل کرده و به جواب های بهتری دست یابد.

در هر مسئله و در هر تکرار، متغیرهای ثابت شونده باید طوری انتخاب شوند که بتوان مسئله کاهیده را در مدت زمانی معقول به صورت بهینه حل کرد.

عمل ثابت کردن برخی از متغیرها در یک مسئله برنامه ریزی عدد صحیح و آزاد گذاشتن بقیه، به منظور کاهش پیچیدگی و دستیابی به جواب هایی با کیفیت بهتر (تابع هدف کمتر) در زمانی کوتاه تر را جستجو در محل متغیرهای ثابت نشده یا « جستجوی محلی<sup>۱</sup> نامیده ایم.

این رویه تا زمانی ادامه می یابد که در ۳ تکرار متوالی هیچ بهبودی در تابع هدف به وجود نیاید (عدد ۳ به صورت تجربی به دست آمده است). در این حالت پس از توقف، مجموع زمان های صرف شده تا این مرحله را به عنوان کل زمان حل و تابع هدف به دست آمده در آخرین تکرار را به عنوان جواب نهایی معرفی می کنیم. فلوجارت مربوط به روش جستجوی محلی تعبیه شده در شاخه و برش، در شکل ۱ نمایش داده شده است.

#### ۴. حل یک مثال عددی

در این قسمت برای درک بهتر مدل و روش پیشنهادی یک مثال عددی  $(P \times M \times C \times H) = (15 \times 10 \times 3 \times 2)$  را با دو روش حل می کنیم.

پارامترهای مربوطه در جدول ۱ و ۲ آمده اند:

<sup>۱</sup>. Local Search

جدول ۲. پارامترهای ورودی

تقاضا	نوع قطعه														
	۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰	۱۱	۱۲	۱۳	۱۴	۱۵
$DE_{p1}$	۰	۹۰۰	۰	۰	۰	۴۵۰	۵۵۰	۴۵۰	۴۰۰	۰	۱۰۰۰	۶۰۰	۴۰۰	۰	۷۵۰
$DE_{p2}$	۴۰۰	۰	۷۵۰	۸۰۰	۴۵۰	۰	۰	۰	۸۵۰	۹۵۰	۱۰۰۰	۴۰۰	۰	۵۰۰	۹۰۰
$B_p^{inter}$	۳۰	۴۰	۲۵	۳۰	۲۰	۲۵	۳۵	۴۰	۴۰	۵۰	۳۵	۴۰	۳۵	۳۵	۲۵
$B_p^{intra}$	۶	۸	۵	۶	۴	۵	۷	۸	۸	۱۰	۷	۸	۷	۷	۵

مراحل اجرای روش جستجوی محلی تعبیه شده در شاخه و برش برای حل این مثال، در قالب جدول ۳ نشان داده شده است:

جدول ۳. لیست متغیرهای تثبیت شده در هر تکرار

شماره تکرار	متغیرهای ثابت شده	مقدار تابع هدف (واحد هزینه)	زمان صرف شده (ثانیه)
۱	حل اولیه	۱۹۲۰۶۳/۵۰	۵۲/۶۹
۲	$(N_{mch}, KA_{mch}, KR_{mch}) \forall (c = 1)$	۱۹۰۲۷۷/۰۰	۰/۵۳
۳	$(N_{mch}, KA_{mch}, KR_{mch}) \forall (c = 2)$	۱۸۶۵۶۷/۰۰	۳/۰۶
۴	$(N_{mch}, KA_{mch}, KR_{mch}) \forall (c = 3)$	۱۸۵۴۶۴/۵۰	۳۴/۸۱
۵	$(N_{mch}, KA_{mch}, KR_{mch}) \forall (c = 1)$	۱۸۵۴۶۴/۵۰	۴۲/۳۸
۶	$(N_{mch}, KA_{mch}, KR_{mch}) \forall (c = 2)$	۱۸۳۷۲۵/۵۰	۵۱/۹۲
۷	$(N_{mch}, KA_{mch}, KR_{mch}) \forall (c = 1)$	۱۸۳۶۵۴/۵۰	۴۲/۹۵
۸	$(N_{mch}, KA_{mch}, KR_{mch}, x_{pjmch}, y_{pjcc'h}) \forall (c = 2)$	۱۸۳۶۵۴/۵۰	۰/۲۸
۹	$(N_{mch}, KA_{mch}, KR_{mch}, x_{pjmch}, y_{pjcc'h}) \forall (c = 3)$	۱۸۳۶۵۴/۵۰	۳/۴۴
۱۰	$(N_{mch}, KA_{mch}, KR_{mch}, x_{pjmch}, y_{pjcc'h}) \forall (h = 1)$	۱۸۳۶۵۴/۵۰	۵/۹۲

تصادفی تولید کرده ایم. حرف U در این جدول نماد توزیع یکنواخت<sup>۱</sup> است. این تست مسئله ها به کمک نرم افزار بهینه سازی 11 GAMS/CPLEX بر روی یک کامپیوتر شخصی (Pentium IV)، با پردازنده مرکزی ۳ گیگا هرتزی و ۲ گیگابایت RAM، تحت سیستم عامل Widow XP SP2 با دو روش شاخه و برش استاندارد و جستجوی محلی تعبیه شده در شاخه و برش، حل شده اند. یکی از ویژگی های منحصر به فرد حل کننده<sup>۲</sup>

بر طبق جدول ۳، روش پیشنهادی طی ۱۰ تکرار و در مدت ۲۳۷/۹۸ ثانیه توانسته است به تابع هدفی برابر ۱۸۳۶۵۴/۵۰ برسد، در حالی که روش استاندارد برای رسیدن به تابع هدفی برابر ۱۸۳۸۷۶/۵۰ (یعنی ۲۲۲ واحد هزینه بیشتر) به ۱۰۸۰۰ ثانیه (۳ ساعت) زمان نیاز دارد.

### ۵. نتایج محاسباتی

به منظور اعتبار سنجی مدل ارائه شده و بررسی کارایی روش پیشنهادی برای حل آن، ۱۱ تست مسئله فرضی از کوچک مقیاس تا بزرگ مقیاس، بر طبق اطلاعات جدول ۴، به صورت

<sup>۱</sup>. Uniform Distribution

<sup>۲</sup>. Solver

جدول ۴. اطلاعات مربوط به تولید تصادفی تست مسئله ها

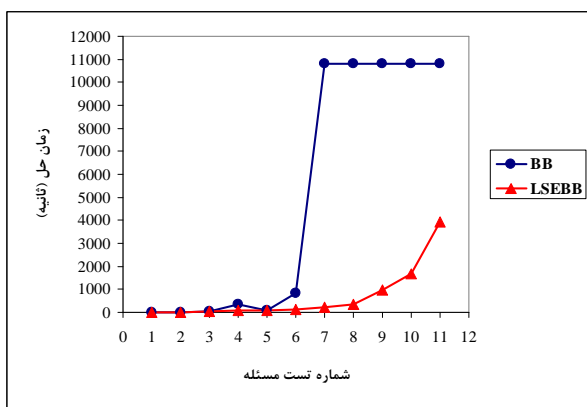
پارامتر	مقدار	پارامتر	مقدار
$DE_{ph}$	U (100-1000)	$UNS_m = INS_m$	U (250-450)
$t_{pjm}$	U (0-1)	$B_p^{inter}$	U (10-50)
$\sum_{m=1}^M a_{pjm}$	2-3	$B_p^{intra}$	$B_p^{inter} / 5$
$\alpha_m$	U (1200-1800)	$\gamma^{inter}$	5
$\beta_m$	U (6-10)	$\gamma^{intra}$	1

CPLEX امکان شروع حل یک مسئله برنامه ریزی عدد صحیح از یک نقطه موجه یا غیر موجه است که کاربر آن را وارد می کند. در پیاده سازی روش پیشنهادی، از این ویژگی به خوبی بهره برداری شده است. در حالتی که روش شاخه و برش استاندارد نمی تواند تست مسئله را به صورت بهینه حل کند، پس از ۳ ساعت (۱۰۸۰۰ ثانیه) فرآیند حل را قطع می کنیم. خلاصه نتایج محاسباتی را در جدول ۵ آورده ایم:

جدول ۵: نتایج حل تست مسئله ها

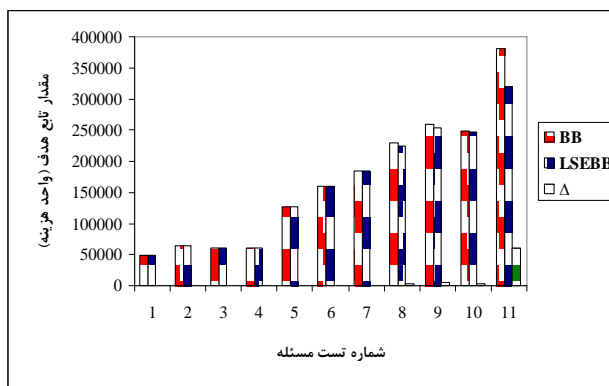
شماره تست مسئله	ابعاد $H \times C \times M \times P$	شاخه و برش استاندارد			جستجوی محلی تعبیه شده در شاخه و برش			اختلاف زمان حل دو روش	اختلاف تابع هدف دو روش ( $\Delta$ )
		زمان حل (ثانیه)	شکاف بهینگی (درصد)	تابع هدف (واحد هزینه)	زمان رسیدن به حل اولیه (ثانیه)	کل زمان حل (ثانیه)	تابع هدف (واحد هزینه)		
۱	۲×۳×۵×۶	۱/۱۴	۰	۴۹۰۸۸/۷۵	۱/۱۴	۱/۱۴	۴۹۰۸۸/۷۵	۰	۰
۲	۳×۳×۵×۶	۲/۸۱	۰	۶۴۰۵۲/۵۴	۲/۸۱	۲/۸۱	۶۴۰۵۲/۵۴	۰	۰
۳	۲×۳×۱۰×۸	۶۰/۳۶	۰	۶۰۲۲۱/۹۷	۶۰/۳۶	۶۰/۳۶	۶۰۲۲۱/۹۷	۰	۰
۴	۳×۳×۱۰×۸	۳۴۷/۸۱	۰	۶۱۳۲۰/۴۴	۶۳/۰۹	۷۱/۸۵	۶۱۳۲۰/۴۴	۲۷۵/۹۶	۰
۵	۲×۳×۱۰×۱۱	۶۹/۳۰	۰	۱۲۶۹۳۹/۷۶	۶۹/۳۰	۶۹/۳۰	۱۲۶۹۳۹/۷۶	۰	۰
۶	۳×۳×۱۰×۱۱	۸۲۱/۰۰	۰	۱۵۹۳۹۶/۹۷	۷۳/۰۳	۱۳۲/۷۵	۱۵۹۳۹۶/۹۷	۶۶۸/۲۵	۰
۷	۲×۳×۱۰×۱۵	۱۰۸۰۰	۰/۸۸	۱۸۳۸۷۶/۵۰	۵۲/۶۹	۲۳۷/۹۸	۱۸۳۶۵۴/۵۰	۱۰۵۶۲/۰۲	۲۲۲
۸	۳×۳×۱۰×۱۵	۱۰۸۰۰	۷/۲۳	۲۲۸۷۳۸/۵۰	۸۲/۱۱	۳۳۱/۴۵	۲۲۴۸۰۳/۰۰	۱۰۴۶۸/۵۵	۳۹۳۵/۵
۹	۲×۳×۱۵×۲۵	۱۰۸۰۰	۸/۵۶	۲۵۸۴۲۵/۷۰	۲۰۹/۹۴	۹۷۹/۰۹	۲۵۴۰۲۱/۱۰	۹۸۲۰/۹۱	۴۴۰۴/۶
۱۰	۲×۳×۱۵×۳۰	۱۰۸۰۰	۶/۶۹	۲۴۹۳۳۵/۰۰	۲۳۶/۲۹	۱۶۷۴/۲۹	۲۴۶۲۲۷/۲۰	۹۱۲۵/۷۱	۳۱۰۷/۸
۱۱	۲×۴×۲۰×۴۰	۱۰۸۰۰	۳۵/۶۷	۳۸۰۲۵۹/۶۰	۲۹۳/۸۰	۳۹۳۵/۹۵	۳۱۹۶۳۷/۸۰	۶۸۶۴/۰۵	۶۰۶۲۱/۸

شکل ۳، کل زمان حل در دو روش را به ازای هر تست مسئله نشان می دهد:



شکل ۳. مقایسه کل زمان حل در دو روش

شکل ۲، مقدار تابع هدف دو روش و همچنین اختلاف آن ها را به ازای هر تست مسئله نشان می دهد:



شکل ۲. مقایسه تابع هدف در دو روش

## مراجع

- [1] Wemmerlov, U., Johnson, D., "Cellular Manufacturing at 46 user Plants: Implementation Experiences and Performance Improvements", International Journal of Production Research, Vol. 35, No. 1, Jan. 1997, pp. 29-49.
- [2] Fry, T., Breen, M., Wilson, M., "A Successful Implementation of Group Technology and Cell Manufacturing", Production and Inventory Management Journal, Vol. 28, No. 3, 1987, pp. 4-6.
- [3] Wemmerlov, U., Hyer, N.L., "Procedures for Part-Family/machine Group Identification Problem in Cellular Manufacturing", Journal of Operations Management, Vol. 6, No. 2, Feb. 1986, pp. 125-147.
- [4] Papaioannou, G., Wilson, J.M., "The Evolution of Cell Formation Problem Methodologies Based on Recent Studies (1997-2008): Review and Directions for Future Research", Vol. 206, No. 3, Nov. 2010, pp. 509-521.
- [5] Selim, H., Askin, R., Vakharia, A., "Cell Formation in Group Technology: Review, Evaluation and Direction for Future Research", Computers & Industrial Engineering, Vol. 34, No. 1, Jan. 1998, pp. 2-30.
- [6] Rheault, M., Drolet, J., Abdounour, G., "Physically Reconfigurable Virtual Cells: A Dynamic Model for a Highly Dynamic Environment", Computers & Industrial Engineering, Vol. 29, No. (1-4), Sep. 1995, pp. 221-225.
- [7] Safaei, N., Saidi-Mehrdad, M., Jabal-Ameli, M. S., "A Hybrid Simulated Annealing for Solving an Extended Model of Dynamic Cellular Manufacturing Systems", European Journal of Operational Research, Vol. 185, No. (1-2), Mar. 2008, pp. 563-592.
- [8] Marcoux, Y., Drolet, J., Abdounour, G., "Studying the Performance of a Dynamic Cellular Manufacturing System", Computers & Industrial Engineering, Vol. 33, No. 1, Oct. 1997, pp. 239-242.
- [9] Wilhelm, W., Chou, C., Chang, D., "Integrating Design and Planning Considerations in Cell Formation", Annals of Operations Research, Vol. 77, No. 1, 1998, pp. 97-107.
- [10] Chen, M., "A Mathematical Programming Model for Systems Reconfiguration in a Dynamic Cell Formation Condition", Annals of Operations Research Vol. 77, No. 1, 1998, pp. 109-128.
- [11] Wicks, E.M., Reasor, R.J., "Designing Cellular Manufacturing Systems with Dynamic Part Populations", IIE Transactions, Vol. 31, No. 1, Jan. 1999, pp. 11-20.
- [12] Balakrishnan, J., Cheng, C.H., "Multi-Period Planning and Uncertainty Issues in Cellular Manufacturing: a Review and Future Directions", European Journal of Operational Research, Vol. 117, No. (1, 16), Feb. 2007, pp. 281-309.
- [13] Mungwattana, A., "Design of Cellular Manufacturing Systems for Dynamic and Uncertain Production

## ۶. نتیجه گیری

در این مقاله یک مدل ریاضی جدید برای مسئله تشکیل سلول پویا در یک سیستم تولید سلولی ارائه شد که علاوه بر کمینه کردن تعداد حرکات بین سلولی، آن‌ها را به کوتاه‌ترین فواصل تخصیص می‌دهد. در ادامه مدل پیشنهادی به کمک روش مطرح شده در مقاله [۲۰] با حداقل تعداد محدودیت‌ها خطی شد. استفاده از محدودیت‌های کمتر در خطی‌سازی باعث می‌شود تا بتوانیم تست مسائل بزرگتری را در زمان کمتری حل کنیم. به دلیل NP-hard بودن مدل خطی شده، نمی‌توان مسائل بزرگ مقیاس را در مدت زمانی معقول به صورت بهینه حل کرد. بدین منظور، روش جدیدی تحت عنوان «جستجوی محلی تعبیه شده در شاخه و برش» توسعه داده شد. ایده اصلی این روش کاهش پیچیدگی مسئله اصلی با تثبیت برخی از متغیرها و در نتیجه زائد شدن برخی از محدودیت‌ها در هر تکرار و دستیابی به حل‌هایی با کیفیت بهتر (تابع هدف کمتر) در مدت زمانی کوتاه‌تر است. حل ۱۱ تست مسئله در ابعاد مختلف به کمک نرم افزار بهینه‌سازی 11 GAMS/CPLEX، سرعت و کارایی روش پیشنهادی را تأیید می‌کند. در تمامی موارد، روش «جستجوی محلی تعبیه شده در شاخه و برش» نسبت به روش «شاخه و برش استاندارد» بهتر عمل کرده است. بر طبق شکل ۲، روش پیشنهادی توانسته است جواب‌های یکسان یا بهتری را در مقایسه با روش استاندارد به دست آورد (تأیید کارایی روش پیشنهادی). در چهار تست مسئله آخر (۸، ۹، ۱۰ و ۱۱)، این موضوع محسوس‌تر است. شکل ۳ برتری روش توسعه یافته را نسبت به روش استاندارد، از نظر معیار سرعت به خوبی نشان می‌دهد (تأیید سرعت روش پیشنهادی). در کل می‌توان گفت که «جستجوی محلی تعبیه شده در شاخه و برش» جواب‌های یکسان یا بهتری را در مدت زمانی بسیار کوتاه‌تر نسبت به «شاخه و برش استاندارد» به دست می‌آورد.

## ۷. پیشنهاداتی برای تحقیقات آتی

می‌توان مدل ارائه شده در این مقاله را با استفاده از روش‌های فرا ابتکاری نظیر الگوریتم ژنتیک، شبیه‌سازی تیرید، جستجوی ممنوعه و ... حل کرد و نتایج آن را با روش پیشنهاد شده در این مقاله مقایسه نمود. چه بسا اگر حل اولیه روش «جستجوی محلی تعبیه شده در شاخه و برش» توسط یک الگوریتم فرا ابتکاری تأمین شود، بتوان از مزایای هر دو روش بهره‌مند شد (روش‌های ترکیبی). علاوه بر این، فرصت توسعه مدل ارائه شده در این مقاله به کمک تغییر فرضیات آن یا افزودن فرض‌های جدید به آن، و استفاده از روش‌های دیگر خطی‌سازی و مقایسه آنها همچنان برای دیگر محققان باز است.

*Requirements with Presence of Routing Flexibility*", Ph.D. thesis, 2000, Virginia Polytechnic Institute and State University, Blackburg, VA.

- [14] Tavakkoli–Moghaddam, R., Aryanezhad, M.B., Safaei, N., Azaron, A., "Solving a Dynamic Cell Formation Problem using Metaheuristics", *Applied Mathematics and Computation*, Vol. 170, No. 2, Nov. 2005, pp. 761–780.
- [15] Saidi–Mehrabad, M., Safaei, N., "A New Model of Dynamic Cell Formation by a Neural Approach", *International Journal of Advance Manufacturing Technology*, Vol. 33, No. (9-10), Jul. 2007, pp. 1001–1009.
- [16] Bajestani, A.M., Rabbani, M., Rahimi-Vahed, A.R., Khoshkhou, G.B., "A Multiobjective Scatter Search for a Dynamic Cell Formation Problem". *Computers and Operations Research* 36, Mar. 2009, pp.777–794.
- [17] Safaei, N., Tavakkoli–Moghaddam, R., "Integrated Multi-Period Cell Formation and Subcontracting Production Planning in Dynamic Cellular Manufacturing Systems", *International Journal of production Economics*, Vol. 120, No. 2, Aug. 2009, pp. 301–319.
- [18] Defersha, F.M., Chen, M., "A Comprehensive Mathematical Model for the Design of Cellular Manufacturing Systems", *International Journal of production Economics*, Vol. 103, No. 2, Oct. 2006, pp. 767–783.
- [19] Defersha, F.M., Chen, M., "A Linear Programming Embedded Genetic Algorithm for an Integrated Cell Formation and Lot Sizing Considering Product Quality". *European Journal of Operational Research* 187, May. 2008, pp. 46–69.
- [20] Kioon, S.A., Bulgak, A.A., Bektas, T., "Integrated Cellular Manufacturing System Design with Production Planning and Dynamic System Reconfiguration", *European Journal of Operational Research*, Vol. 192, No. 2, Jan. 2009, pp. 414–428.
- [21] Deljoo, V., Mirzapour Al-e-hashem, S.M.J., Deljoo, F., Aryanezhad, M.B., "Using Genetic Algorithm to Solve Dynamic Cell Formation Problem", *Applied Mathematical Modelling*, Vol. 34, No. 4, Apr. 2010, pp. 1078–1092.
- [22] Herrán, A. Defersha, F.M., Chen, M., de la Cruz, J.M., "An Integrated Multi-Period Planning of the Production and Transportation of Multiple Petroleum Products in a Single Pipeline System", [International Journal of Industrial Engineering Computations](#), Vol.2, No.1, 2011, pp. 19-44

**پیوست**

برای خطی سازی حاصلضرب دو متغیر صفر و یک، در جملات سوم و چهارم تابع هدف مدل پیشنهادی، راهکار کلاسیک به صورت زیر است:

ابتدا یک دسته متغیر صفر و یک جدید  $(y_{pjcc'h})$  برای جایگزینی با جمله حاصلضرب به صورت زیر تعریف می کنیم:

$$y_{pjcc'h} = \left( \sum_{m=1}^M x_{pjmch} \right) \left( \sum_{m=1}^M x_{p(j+1)mc'h} \right)$$

سپس دو دسته محدودیت زیر را به مدل می افزاییم، تا مقدار متغیر  $(y_{pjcc'h})$ ، برابر مقدار حاصلضرب باشد:

$$y_{pjcc'h} \geq \left( \sum_{m=1}^M x_{pjmch} \right) + \left( \sum_{m=1}^M x_{p(j+1)mc'h} \right) - 1 \quad \forall p, j, c, c', h \quad (I)$$

$$y_{pjcc'h} \leq \frac{1}{2} \left( \sum_{m=1}^M x_{pjmch} + \sum_{m=1}^M x_{p(j+1)mc'h} \right) \quad \forall p, j, c, c', h \quad (II)$$

برای اثبات این روش به بیان چهار حالت امکان پذیر می پردازیم:

$$\left( \sum_{m=1}^M x_{pjmch} = 0 \right), \left( \sum_{m=1}^M x_{p(j+1)mc'h} = 0 \right) \Rightarrow \begin{cases} y_{pjcc'h} \geq -1 \\ y_{pjcc'h} \leq 0 \end{cases}, y_{pjcc'h} \in \{0, 1\} \Rightarrow y_{pjcc'h} = 0 \quad (الف)$$

$$\left( \sum_{m=1}^M x_{pjmch} = 1 \right), \left( \sum_{m=1}^M x_{p(j+1)mc'h} = 0 \right) \Rightarrow \begin{cases} y_{pjcc'h} \geq 0 \\ y_{pjcc'h} \leq 0.5 \end{cases}, y_{pjcc'h} \in \{0, 1\} \Rightarrow y_{pjcc'h} = 0 \quad (ب)$$

$$\left( \sum_{m=1}^M x_{pjmch} = 0 \right), \left( \sum_{m=1}^M x_{p(j+1)mc'h} = 1 \right) \Rightarrow \begin{cases} y_{pjcc'h} \geq 0 \\ y_{pjcc'h} \leq 0.5 \end{cases}, y_{pjcc'h} \in \{0, 1\} \Rightarrow y_{pjcc'h} = 0 \quad (ج)$$

$$\left( \sum_{m=1}^M x_{pjmch} = 1 \right), \left( \sum_{m=1}^M x_{p(j+1)mc'h} = 1 \right) \Rightarrow \begin{cases} y_{pjcc'h} \geq 1 \\ y_{pjcc'h} \leq 1 \end{cases}, y_{pjcc'h} \in \{0, 1\} \Rightarrow y_{pjcc'h} = 1 \quad (د)$$

ها ثابت کردند که اگر ضریب جمله حاصلضرب در تابع هدف کمینه سازی، مثبت باشد (یا بالعکس، ضریب جمله حاصلضرب در تابع هدف بیشینه سازی، منفی باشد)، می توان فقط دسته محدودیت (I) را به کار برد و از آوردن دسته محدودیت (II) صرف نظر کرد [۲۰]. برای اثبات این روش چهار حالت زیر را در نظر بگیرید:

نتیجه این که با معرفی  $(P)$  متغیر جدید و  $(P)$  محدودیت جدید می توان حاصلضرب دو متغیر صفر و یک را خطی کرد. در واقع تعداد محدودیت های جدید لازم برای خطی سازی، دو برابر تعداد متغیر ها است و نتیجتاً با افزایش ابعاد مسئله، حل مدل خطی شده به کمک نرم افزار بهینه سازی GAMS/CPLEX 11 امکان پذیر نخواهد بود. برای حل چنین مشکلی کیون و همکارانش (Kioon et al, 2009) راهکار جدیدی ارائه کردند. آن

$$\left( \sum_{m=1}^M x_{pjmch} = 0 \right), \left( \sum_{m=1}^M x_{p(j+1)mc'h} = 0 \right) \Rightarrow \{y_{pjcc'h} \geq -1\} \Rightarrow \begin{cases} y_{pjcc'h} = 0 \text{ or } 1 \\ \min(+ y_{pjcc'h}) \end{cases} \Rightarrow y_{pjcc'h} = 0 \quad (الف)$$

$$\left( \sum_{m=1}^M x_{pjmch} = 1 \right), \left( \sum_{m=1}^M x_{p(j+1)mc'h} = 0 \right) \Rightarrow \{y_{pjcc'h} \geq 0\} \Rightarrow \begin{cases} y_{pjcc'h} = 0 \text{ or } 1 \\ \min(+ y_{pjcc'h}) \end{cases} \Rightarrow y_{pjcc'h} = 0 \quad (ب)$$

$$\left( \sum_{m=1}^M x_{pjmch} = 0 \right), \left( \sum_{m=1}^M x_{p(j+1)mc'h} = 1 \right) \Rightarrow \{y_{pjcc'h} \geq 0\} \Rightarrow \begin{cases} y_{pjcc'h} = 0 \text{ or } 1 \\ \min(+ y_{pjcc'h}) \end{cases} \Rightarrow y_{pjcc'h} = 0 \quad (ج)$$

$$\left( \sum_{m=1}^M x_{pjmch} = 1 \right), \left( \sum_{m=1}^M x_{p(j+1)mch} = 1 \right) \Rightarrow \left\{ \begin{array}{l} y_{pjcc'h} \geq 1 \\ y_{pjcc'h} \in \{0,1\} \end{array} \right\} \Rightarrow y_{pjcc'h} = 1 \quad (د)$$

در واقع ضریب مثبت جمله حاصلضرب در تابع هدف، ما را از آوردن دسته محدودیت (II) بی نیاز می کند. با به کار بردن این روش، تعداد محدودیت های خطی کننده به نصف تقلیل می یابد و نتیجتاً قادر به حل تست مسائل بزرگتری خواهیم بود.