



بکارگیری رویه جستجوی تصادفی تطابقی حریصانه برای زمانبندی مسئله جریان کارگاهی بدون صف های میانی با استفاده از تبدیل به مسئله فروشنده دوره گرد

جواد بهنامیان (نویسنده مسؤل)

استادیار، دانشکده فنی و مهندسی، دانشگاه بوعلی سینا همدان

Email: Behnamian@basu.ac.ir

روناک محمدی

کارشناسی ارشد مهندسی صنایع، دانشگاه بوعلی سینا همدان

امید رضایی

کارشناسی ارشد مهندسی صنایع، دانشگاه بوعلی سینا همدان

تاریخ دریافت: ۹۵/۴/۷ * تاریخ پذیرش: ۹۵/۸/۱۹

چکیده

هدف از این مقاله یافتن توالی بهینه به منظور کمینه کردن فاصله زمانی ساخت برای مسئله زمانبندی جریان کارگاهی بدون صفهای میانی میباشد. مسائل زمانبندی بدون انتظار در آن دسته از محیطهای تولیدی رخ میدهد که در آن یک کار میبایست از آغاز تا پایان بر روی یک ماشین یا چند ماشین بدون وقفه پردازش شود. از آنجایی که ساختار این مسئله شباهت بسیاری با مسئله فروشنده دورهگرد دارد، در تحقیق حاضر از یک رویکرد جدید جهت بدست آوردن دیرکردها کمک گرفته شده به گونه ای که با هدف یافتن توالی بهینه عملیاتی که کمترین فاصله زمانی ساخت را داراست از ماتریس دیرکردهای بدست آمده از مسئله فروشنده دورهگرد استفاده شده است. همچنین از الگوریتم جستجوی تصادفی تطابقی حریصانه برای حل مسئله تعیین توالی جریان کارگاهی بدون صفهای میانی استفاده و کارایی آن پس از تعیین پارامتر از طریق روش فاکتوریل، با الگوریتم کلونی مورچگان مقایسه شده است.

کلمات کلیدی: زمانبندی، الگوریتم جستجوی تصادفی تطابقی حریصانه، جریان کارگاهی بدون صف های میانی، مسئله فروشنده دوره گرد.

۱- مقدمه

رقابت‌های موجود بین شرکت‌های تولیدی، موجب انعطاف‌پذیری و کارایی بیشتر آنها در مواجهه با این رقابت‌ها شده است. توالی و زمانبندی، از مسائل تصمیم‌گیری هستند که نقش اساسی در صنایع خدماتی و تولیدی را ایفا می‌کنند. در واقع این دو مقوله برای بهبود کارایی جریان تولیدی به کار می‌روند و از این رو یافتن بهترین توالی به منظور حداقل یا حداکثر کردن معیارهای مدنظر برای افزایش کارایی تولید از اهمیت فراوانی برخوردار است. در مورد مدل‌های جریان کارگاهی، کارها به صورت مجموعه-ای از عملیات در نظر گرفته می‌شود که ساختار تقدمی خاصی دارند. بنابراین تکمیل هر کار مستلزم توالی مشخصی از عملیات است. ویژگی کارگاه با جریان کارگاهی، گردش کار تک جهتی است. به عبارت دیگر این نوع کارگاه دارای توالی طبیعی از ماشین‌آلات است که در آن می‌توان ماشین‌ها را به ترتیبی شماره‌گذاری کرد که اگر عملیات j ام هر کار مقدم بر عملیات k ام آن کار باشد، شماره عملیات j ام کمتر از عملیات k ام آن باشد. ماشین‌ها در جریان کارگاهی به صورت $1, 2, \dots, m$ شماره‌گذاری می‌شود و سلسله عملیات کار i به طور متناوب به صورت $(1, i), (2, i), \dots, (m, i)$ شماره‌گذاری می‌شود. هر کاری را که دقیقاً m نوع عملیات داشته باشد، می‌توان در نظر گرفت و اگر تعداد عملیات کاری کمتر از m باشد، زمان عملیات مربوطه (عملیات صورت نگرفته) را می‌توان صفر در نظر گرفت. در مسئله جریان کارگاهی n توالی کار مختلف برای هر ماشین امکان‌پذیر است و بنابراین $(n!)^m$ برنامه مختلف باید بررسی شود. هدف از این مقاله یافتن توالی بهینه به منظور کمینه کردن فاصله زمانی ساخت برای مسئله زمانبندی جریان کارگاهی بدون صف‌های میانی می‌باشد. مسائل زمانبندی بدون انتظار در آن دسته از محیط‌های تولیدی رخ می‌دهد که در آن یک کار می‌بایست از آغاز تا پایان بر روی یک ماشین یا در بین ماشین‌ها بدون وقفه پردازش شود. علت وقوع چنین محیط‌هایی نوع فن‌آوری و یا فقدان توانایی ذخیره‌سازی بین ماشین‌ها و ایستگاه‌های کاری است؛ به عنوان مثال در تولید فولاد چنین وضعیتی رخ می‌دهد (Baker & Trietsch, 2009).

از آنجایی که ساختار این مسئله شباهت بسیاری با ضابطه مسئله فروشنده دوره‌گرد دارد، در حقیقت با کمی تغییر، مسئله مربوط را می‌توان به صورت مسئله فروشنده دوره‌گرد فرمولبندی کرد. برای فرمولبندی مسئله فروشنده دوره‌گرد می‌توان هر کار را به منزله یک شهر و تاخیرهای هر زوج کار را به منزله فواصل آن دو شهر در نظر گرفت. به علاوه یک شهر مجازی را باید به مسئله اضافه کرد که فاصله این شهر از شهر i برابر با مجموع زمان‌های پردازش کار i و فاصله‌اش از بقیه شهرها برابر صفر می‌باشد. با در نظر گرفتن این شباهت، برای حل مسئله جریان کارگاهی بدون وقفه می‌توان از مدل مسئله فروشنده دوره‌گرد بهره برد و برای بدست آوردن راه حل بهینه این مسئله از رویه جستجوی تصادفی تطابقی حریصانه (GRASP) استفاده کرد.

مسائل زمانبندی جریان کارگاهی بدون وقفه با بیش از دو ماشین جزء مسائل Np -hard دسته‌بندی می‌شوند. تحقیقات بسیاری برای حل مسئله زمانبندی جریان کارگاهی بدون وقفه با در نظر گرفتن معیارهای متفاوتی مانند: حداکثر زمان تکمیل کارها و کل زمان در جریان، صورت گرفته که به ارائه الگوریتم‌های ابتکاری و فراابتکاری زیادی منجر گشته است. در زمینه مسئله جریان کارگاهی بدون وقفه با سه ماشین تحقیقات فراوانی صورت گرفته است که هدف اصلی در این پژوهش‌ها ارائه الگوریتم‌های ابتکاری برای حل مسئله مورد نظر بوده است. ردی و رامامورسی در سال ۱۹۷۲ مسئله جریان کارگاهی بدون وقفه را به عنوان یک مسئله فروشنده دوره‌گرد مدل بندی کردند و سپس از متدهای حل شناخته شده برای مینیمم کردن زمان کل در جریان استفاده کردند. راجندران و چادوی در سال ۱۹۹۰ دو الگوریتم ابتکاری را با در نظر گرفتن معیار زمان در جریان کل، ارائه دادند. آنها برای ایجاد توالی اولیه کارها از دو الگوریتم ابتکاری استفاده کردند و سپس توالی اولیه را بوسیله روش ورود کارها که توسط نواز انسکور هام ارائه شده بود، بهبود دادند و در آخر نشان دادند که کارایی الگوریتم آنها بهتر از الگوریتم‌های ارائه شده توسط بونی و گاندرای در سال ۱۹۷۶ و کینگ و اسپاچیس در سال ۱۹۸۰ است. هال و راجا در سال ۱۹۹۶ مروری جامع بر روی مسائل زمانبندی بدون وقفه ماشین‌ها صورت دادند. آنها در مقاله خود به بررسی و شرح کاربرد این مسئله در صنایع پرداخته و در زمینه پیچیدگی‌های موجود در این مسئله بررسی‌هایی انجام دادند و در نهایت به بررسی نتایج و عملکرد الگوریتم‌های موجود پرداخته و پیشنهاداتی برای انجام دادن تحقیقات آتی ارائه دادند. سلن و هوت در سال ۱۹۸۶ در مطالعه‌ای برای مسئله زمانبندی تصمیم یافته جریان کارگاهی بدون وقفه n کار و m ماشین مدل برنامه‌ریزی آرمانی صحیح مختلط را ارائه کردند و نشان دادند که

رضایت بخش ترین جوابها زمانی حاصل می شود که معیارهای چندگانه ای مانند زمان بیکاری ماشین، زمان جریان ساخت و... را خاطر نشان کرد. الدواسیان و الله وردی در سال ۱۹۹۸ مسئله ای را مورد بررسی قرار دادند که در آن، وقتی کاری پردازش خود را از ماشین ۱ در طول خط تولید شروع می کند کار باید بدون هیچ تاخیری تمام خط تولید را طی کند به طوری که کارها نه فقط بین ماشینها بلکه بر روی آنها نیز نباید منتظر بمانند. برتولوسی در سال ۲۰۰۰ در مقاله ای برای حل مسئله زمانبندی جریان کارگاهی بدون وقفه، الگوریتم ابتکاری را ارائه کرد که هدف از ارائه آن به حداقل رساندن زمان متوسط پردازش کارها می باشد. نتایج محاسباتی بدست آمده از الگوریتم پیشنهادی نشان داد که جواب حاصل از این الگوریتم نسبت به سایر الگوریتمهای استفاده شده برای حل این مسئله از کارایی بهتری برخوردار است چرا که جواب حاصل از آن به جواب بهینه نزدیک تر است. پن و همکاران در سال ۲۰۰۸ الگوریتم بهینه سازی توده ذرات گسسته را برای مسئله زمانبندی جریان کارگاهی بدون وقفه با در نظر گرفتن دو معیار کل زمان جریان و فاصله زمانی ساخت ارائه کردند. لی و همکاران در سال ۲۰۰۸ از یک الگوریتم ابتکاری ترکیبی برای به حداقل رساندن فاصله زمانی ساخت برای مسئله زمانبندی جریان کارگاهی بدون وقفه استفاده کردند و نشان دادند که در مقیاس بزرگ الگوریتم پیشنهادی جوابهای بهتری را بدست می آورد. تسنگ و لین در سال ۲۰۱۰ الگوریتم ترکیبی ژنتیک برای حل مسئله زمانبندی جریان کارگاهی بدون صفهای میانی را با هدف مینیمم کردن فاصله زمانی ساخت بکار گرفتند. گوپتا و استفورد در سال ۲۰۰۶ یک مرور جامع روی مسائل زمانبندی جریان کارگاهی بدون وقفه طی ۵۰ سال گذشته را انجام دادند.

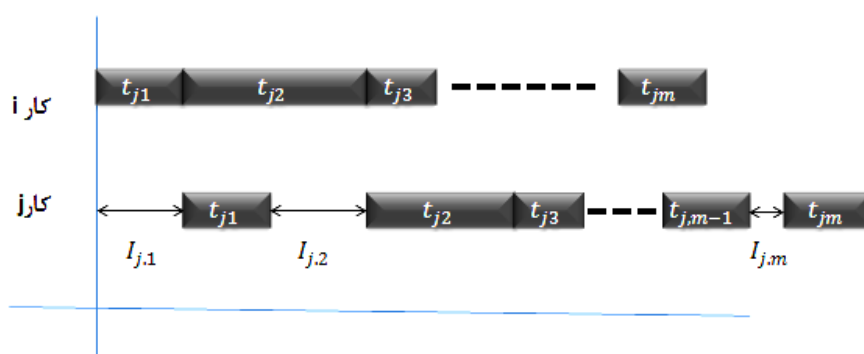
برای مسائل زمانبندی جریان کارگاهی بدون وقفه با معیار کل زمان جریان، الگوریتمهای ابتکاری توسط رجندرا و چاودهوری (۱۹۹۰)، برتولوسی (۲۰۰۰)، الدواسیان و الله وردی (۲۰۰۴) و فرمینان و همکاران (۲۰۱۰) مورد مطالعه قرار گرفته است. از سوی دیگر، الگوریتمهای فراابتکاری شامل الگوریتم ژنتیک چن، نیالی، و آلباجر (۱۹۹۶)، انجماد تدریجی فینک و ووب (۲۰۰۳)، سیستم ایمنی مصنوعی بدن کومار، پراکاش، شانکار و تی واری (۲۰۰۶) و بهینه سازی توده ذرات پن، تسگتیرین، و لی آنگ (۲۰۰۸) برای حل مسئله مدنظر مورد استفاده قرار گرفته اند. در نهایت، فرمینان و همکاران (۲۰۱۰) یک الگوریتم ابتکاری را توسعه دادند به گونه ای که در پژوهش خود برتری روش اکتشافی را نسبت به الگوریتمهای پیشنهادی توسط رجندران و چاودهوری (۱۹۹۰)، برتولوسی (۲۰۰۰)، فینک و ووب (۲۰۰۳) و الدواسیان و الله وردی (۲۰۰۴) نشان دادند، اگرچه الگوریتم پیشنهادی زمان محاسباتی بیشتری را در مقایسه با الگوریتمهای ذکر شده، لازم دارد. در مسئله یافتن کوتاهترین مسیر همیلتونی یا مسئله فروشنده دوره گرد با مجموعه شهرها مفروض و همچنین فواصل بین شهرها با $d(c)$ نشان داده می شود. در این مسئله هدف، یافتن کوتاهترین مسیر همیلتونی است که از یک شهر شروع و به همه شهرها دقیقاً یک بار وارد می شود و در انتها به شهر اول باز می گردد (Hahsler & Hornik, 2009). منشا پیدایش این مسئله دهه ۱۹۲۰ است، زمانی که ریاضیدانی به نام منجر این مسئله را مورد توجه قرار داد. سپس کار بر روی این مسئله توسط انجمن ریاضی برینستون در دهه ۱۹۳۰ ادامه پیدا کرد. در دهه ۱۹۴۰ ریاضیدانی به نام فلود این مسئله را با نام مسئله فروشنده دوره گرد معرفی کرد، مسئله فروشنده دوره گرد، یک مسئله کلاسیک بهینه سازی گسسته است و در زمره مسائل پژوهش عملیاتی قرار می گیرد که توضیح و شرح آن ساده و حل آن مشکل است. مسئله یافتن کوتاهترین مسیر همیلتونی از نتایج حل مسئله فروشنده دوره گرد است. این مسئله می تواند متقارن و یا نامتقارن باشد. در حالت متقارن فاصله دو شهر به جهت حرکت بستگی ندارد (Johnson & McGeoch, 1997). همچنین این مسئله می تواند به صورت احتمالی، با پنجره زمانی، به صورت چندکالایی و در شبکه راه آهن باشد. یافتن کوتاهترین مسیر همیلتونی کاربردهای متفاوتی دارد و در ساخت تراشه های الکترونیکی، زمانبندی کارها، تعیین توالی کارها و در مسیریابی وسایل نقلیه استفاده می شود. این مسئله علاوه بر جنبه نظری از لحاظ جنبه کاربردی نیز اهمیت فراوانی دارد. شناخت روشهای حل این مسئله و مقایسه آنها با یکدیگر و انتخاب روش مناسب، یکی از زمینه های پژوهشی مهم محسوب می شود (Johnson & McGeoch, 1997).

در تحقیق حاضر برای اولین بار یک رابطه جدید ریاضی برای بدست آوردن دیرکردهای مسئله تعیین توالی در جریان کارگاهی بدون وقفه ارائه و این مسئله با الگوریتم GRASP حل و با الگوریتم کلونی مورچگان مقایسه شده است.

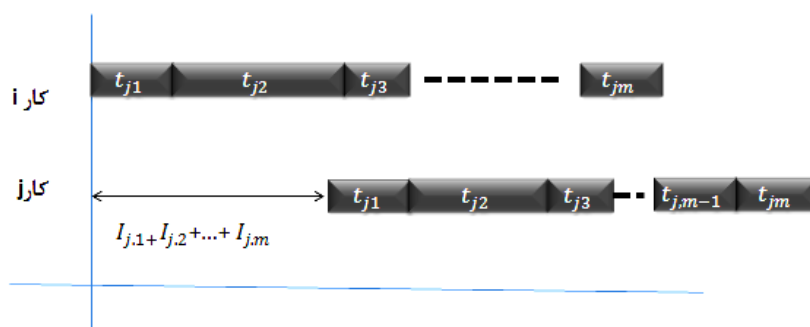
۲- مواد و روش‌ها

دربسیاری از مسائلی که در عمل با آن روبرو هستیم به محض آنکه پردازش یک کار شروع می‌شود، پردازش کار بعدی در توالی عملیات باید بدون انتظار انجام پذیرد. به عبارت دیگر، تشکیل صف در جلوی ماشین‌ها مجاز نیست. چنین مواردی اکثراً در صنایع فلزی اتفاق می‌افتد. خصوصاً در صنایعی که فلزات به صورت گرم غلتک می‌شود. تاخیر بین عملیات موجب سرد شدن فلز می‌شود و عملیات مختلف کاری را بسیار دشوار می‌کند.

فرض کنید دو کار i و j در توالی مجاور یکدیگر است و کار i ام در توالی کارها قبل از کار j ام قرار دارد. شکل ۱ الف مقدار تاخیری را که در انجام کار j ام حاصل می‌شود در حالی که هر دو کار همزمان به کارگاه رسیده باشند و بخواهیم کار i ام بدون انتظار انجام شود را نشان می‌دهد. در این شکل I_{ij} مدت زمان بیکاری یا تاخیر در انجام کار j ام روی ماشین w را نشان می‌دهد. برای انجام بدون انتظار کار j ام و اتمام آن در همان زمانی که در شکل ۱ الف نشان داده شده است لازم است بیکاری‌های بین عملیات کار j ام را همگی به ابتدای عملیات روی کار j ام منتقل کنیم. این انتقال در شکل ۱ ب نشان داده شده است. حال فرض کنیم کار k ام بعد از اتمام کار j ام قرار دارد. ملاحظه می‌شود که تاخیر در انجام عملیات کار k ام به آنچه که قبل از کار j ام اتفاق می‌افتد بستگی ندارد و تنها به زمان انجام عملیات کار j ام بستگی دارد (Baker & Trietsch, 2009).



شکل شماره (۱) الف: زمانبندی عملیات جداگانه روی کارهای متوالی i و j و آرایش مجدد روی هر ماشین بعد از مورد اول، مانع تاخیر در کار j می‌شود بدون اینکه در تکمیل نهایی کار j تاخیری به وجود آید (Baker & Trietsch, 2009).



شکل شماره (۲) ب: زمانبندی عملیات جداگانه روی کارهای متوالی i و j و آرایش مجدد روی هر ماشین بعد از مورد اول، مانع تاخیر در کار j می‌شود بدون اینکه در تکمیل نهایی کار j تاخیری به وجود آید (Baker & Trietsch, 2009).

نمادهای استفاده شده در مقاله به شرح ذیل است:

j : شمارنده کارها

i : کار j ام که در توالی اول قرار دارد

k : کار زام که در توالی دوم قرار دارد

m : شمارنده ماشین‌ها

t_{iw} : زمان پردازش کار i روی ماشین w

Z_{im} : زمان اتمام کار i روی ماشین m

Z_{imk} : زمان اتمام کار k روی ماشین m که بعد از انجام کار i شروع می‌شود

D_{ik} : کل تاخیر کار به گونه ای که اول کار i و بعد کار k انجام شود

D_{ij} نمایانگر کل تاخیر (از شروع کار i ام) در انجام کار j وقتی که بعد از کار i در ترتیب قرار گرفته است باشد. یعنی ،

$$D_{ij} = I_{j1} + I_{j2} + \dots + I_{jm} \quad (1)$$

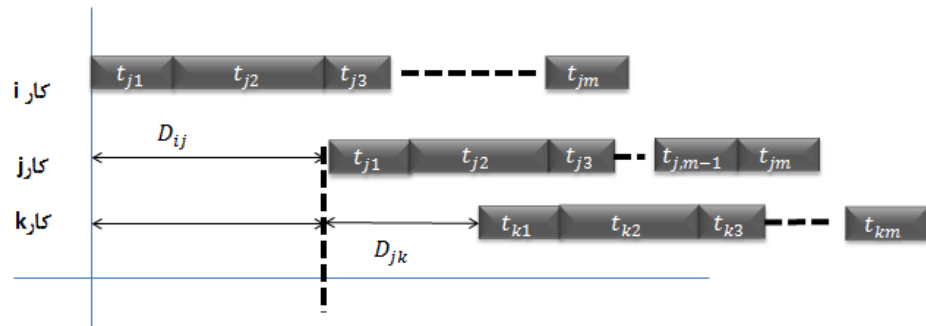
همچنین فرض کنید D_{jk} مقدار کل تاخیر (از شروع کار j ام) در انجام کار k که در توالی بعد از کار j قرار دارد باشد. اگر برنامه زمانی فقط از این سه کار تشکیل شده باشد رابطه مربوط به زمان کل انجام برنامه زمانی مربوط به توالی (شکل ۲) مطابق با برنامه زیر خواهد بود که آن را با M نشان می‌دهند:

$$M = D_{ij} + D_{jk} + \sum_{w=1}^m t_{kw} \quad (2)$$

و لذا در حالت کلی فاصله زمانی ساخت به صورت زیر بیان می‌شود:

$$M = \sum_{j=1}^{n-1} D_{[j],[j+1]} + \sum_{w=1}^m t_{[n]w} \quad (3)$$

بنابراین فاصله زمانی ساخت در حالت کلی عبارت است از مجموع جملات مربوط به تاخیرهای وابسته به توالی و کل زمان پردازش آخرین کار در توالی.



شکل شماره (۳): زمانبندی عملیات جداگانه برای سه کار هنگامی که هیچ گونه تاخیری میان مجاز نباشد (Baker & Trietsch, 2009)

ساختار این مسئله شباهت بسیار زیادی با مسئله فروشنده دوره‌گرد دارد (Baker & Trietsch, 2009). در حقیقت با کمی تغییر در مسئله مربوط به مدت جریان ساخت را می‌توان به صورت مسئله فروشنده دوره‌گرد فرمولبندی کرد که برای بدست آوردن D_{ij} در روابط بالا از روابط زیر استفاده می‌کنیم:

$$Z_{im} = \sum_{w=1}^m t_{iw} \quad \text{for } i=1, \dots, j \quad m=1, \dots, M \quad (4)$$

با استفاده از رابطه ۴ زمان اتمام کار i روی ماشین m که در توالی اول قرار دارد، بدست می‌آید.

$$Z_{imk} = \sum_{w=1}^m t_{kw} + t_{i1} + \sum_{w=1}^m E_{iwk} (Z_{iw} - Z_{i,w-1,k}) \quad \text{for } i=1, \dots, j \quad m=1, \dots, M \quad k=1, \dots, j \quad i \neq k \quad (5)$$

با استفاده از رابطه ۵ زمان اتمام کار k روی ماشین m که در توالی دوم قرار دارد، بدست می‌آید.

$$D_{ik} = \sum_{w=1}^M E_{iwk} (Z_{iw} - Z_{i,w-1,k}) + t_{i1} \quad \text{for } i=1, \dots, j \quad k=1, \dots, j \quad i \neq k \quad (6)$$

در آن از E_{iwk} انجام گیرد، بدست می‌آید که k قبل از کار i با استفاده از رابطه ۶ مقدار دیرکرد کارها در صورتی که کار

خواهد بود. $E_{i1k} = 0$ خواهد بود یا به عبارت دیگر $E_{iwk} = 0$ مقدار $w=1$ رابطه γ حاصل می‌شود که به ازای

$$E_{iwk} = \begin{cases} Z_{iw} \geq Z_{i,w-1,k} & 1 \\ Z_{iw} < Z_{i,w-1,k} & 0 \end{cases} \quad (7)$$

با توجه به اینکه مقادیر بدست آمده از فرمول‌های بالا حاکی از ماتریس فواصل در مسئله فروشنده دوره‌گرد می‌باشد، این مطلب را نیز باید خاطر نشان کرد که مقادیر ستون آخر از رابطه $\sum_{w=1}^M t_{iw}$ بدست می‌آید و سطر آخر دارای درایه‌هایی برابر صفر می‌باشد.

در ادامه این بخش ساختار الگوریتم پیشنهادی تشریح خواهد شد. از آنجایی که تاکنون برای حل مسئله حاضر از الگوریتم پیشنهادی استفاده نشده است در این بخش به توضیح الگوریتم پرداخته شده است به گونه‌ای که در ادامه با استفاده از نتایج بدست آمده از آن به مقایسه کارایی آن پرداخته شده است. برای حل مسئله مسیریابی الگوریتم *GRASP* از دو فاز فراابتکاری برای حل مسائل بهینه‌سازی ترکیباتی استفاده می‌کند: فاز اول یک فاز سازنده می‌باشد که راه حل‌های اولیه را می‌سازد درحالی که در مرحله دوم به طور کلی یک روش جستجوی محلی برای بهبود راه حل‌های اولیه مرحله اول ارائه می‌شود.

فاز ساخت یک رویه تکراری است که با یک آرایه خالی شروع می‌کند و در هر تکرار یک عنصر به جواب ناقص حاصل شده اضافه می‌شود و تا زمانی که جواب کامل شود این رویه ادامه پیدا می‌کند. در ابتدا کلیه عناصری که امکان شرکت در جواب ناقص را دارند، به شرطی که موجه بودن جواب را به هم نزنند، در نظر گرفته می‌شوند که عناصر کاندید نامیده می‌شوند. سپس هزینه شرکت کردن هر عنصر در جواب ناقص محاسبه شده و تعدادی از این عناصر کاندید که دارای هزینه کمتری باشند، انتخاب و در مجموعه لیست کاندیدای محدود، قرار داده می‌شوند. پس از آن به تصادف از *RCL* انتخاب شده و در جواب ناقص شرکت داده می‌شود. هر بار که یک عنصر در جواب ناقص شرکت داده می‌شود لیست عناصر کاندید و همچنین هزینه شرکت دادن هر عنصر مجدداً محاسبه می‌شود. جواب‌های ساخته شده در فاز ساخت ضرورتاً بهینه نمی‌باشند، معمولاً فاز جستجوی محلی، جواب‌های به دست آمده در فاز ساخت را بهبود می‌دهند. در فاز جستجوی محلی جواب جاری به صورت متوالی با جواب‌های بهتر در همسایگی آن تعویض می‌گردد و این کار تا زمانی ادامه پیدا می‌کند که به یک جواب بهینه محلی ختم شود (Selen & Hott, 1986). در فاز ساخت الگوریتم پیشنهادی برای حل مسئله فروشنده دوره‌گرد از یک روش ابتکاری برای یافتن جواب اولیه مناسب که بتوانیم سریع‌تر در فاز بهبود به جواب بهینه برسیم استفاده شده است. این روش یک شهر را به تصادف از میان شهرهای موجود انتخاب می‌کند. این که کدام شهر را برای یافتن جواب اولیه انتخاب کنیم در جواب پایانی ما هیچ تاثیری نخواهد داشت. بعد از این که یک شهر را انتخاب کردیم، این روش در میان شهرهای مجاور این شهر نزدیک‌ترین شهر دیدار نشده را انتخاب و به آن شهر می‌رود و این روش را به همین صورت ادامه داده تا همه شهرها در یک تور قرار گیرند. به این ترتیب یک جواب ممکن و قابل قبول را پیدا کرده‌ایم که در فاز بهبود آن را بهینه می‌نماییم. لازم به ذکر است که این جواب لزوماً جواب بهینه نیست و می‌بایست آن را بهبود دهیم. روش‌های جستجوی محلی تصادفی نشان داده‌اند که برای حل بسیاری از مسائل پیچیده مفیدند. الگوریتم انجماد تدریجی که در میان آنها از همه بیشتر استفاده می‌شود. الگوریتم *SA* از اولین الگوریتم‌های توسعه یافته برای مسئله مسیریابی است. در این مطالعه از روش *SA* استاندارد با ساختار همسایگی متشکل از ۴ اپراتور که در داخل مسیر به عنوان اپراتورهای بهبود می‌باشند، که این اپراتورها با الگوریتم انجماد تدریجی ادغام می‌شوند.

۳- نتایج و بحث

همان‌طور که گفته شد در فاز دوم الگوریتم *GRASP* از الگوریتم انجماد تدریجی برای بهبود جواب به دست آمده در مرحله اول استفاده می‌کنیم. در الگوریتم انجماد تدریجی سه پارامتر مهم ضریب کاهش دما (α) در هر مرحله، دمای اولیه (T_0) و تعداد تکرارها در هر زنجیره مارکوف می‌باشد. پارامتر α در واقع نحوه کاوش در فضای جستجو را تعیین می‌کند. هرچه این مقدار کمتر باشد الگوریتم جواب‌های بیشتری را بررسی می‌کند. دمای اولیه هم مانند پارامتر α هرچه بیشتر باشد الگوریتم جواب‌های بیشتری را مورد بررسی قرار داده ولی بالا بودن دما باعث می‌شود که الگوریتم در زمان قابل قبولی جواب نزدیک به جواب بهینه را به ما ندهد. تعداد تکرارها در هر زنجیره مارکوف تعیین کننده تعداد جواب‌های ممکن است که در هر دما ما مورد بررسی قرار

می‌دهیم.

برای تنظیم پارامترهای الگوریتم که موجب افزایش کارایی الگوریتم می‌شود، عواملی را که در بالا معرفی شد سطوحی را برای آن‌ها در نظر می‌گیریم که در جدول ۱ آورده شده است و به ازای هر سطح از پارامترها، برنامه را ۵ بار اجرا کرده و میانگین این سطح را به عنوان ستون مینا در نرم افزار *Minitab 16* جهت تجزیه و تحلیل و محاسبات قرار می‌دهیم. عوامل و سطوح را به شکل زیر در نظر می‌گیریم:

جدول شماره (۱): پارامترها و سطوح آن

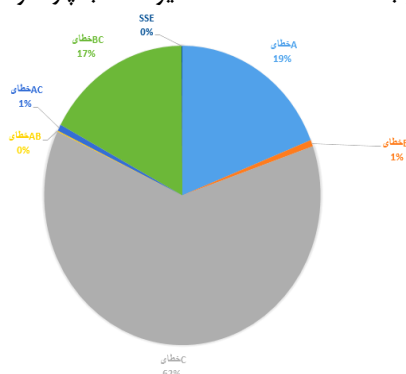
پارامتر	سطح اول	سطح دوم	سطح سوم
تعداد تکرار زنجیره مارکوف	۵	۱۰	۲۰
دمای اولیه	۵۰۰		۱۰۰۰
ضریب کاهش دما	۰/۰۱	۰/۰۵	۰/۰۱

همان طور که مشخص است برای تنظیم پارامترهای الگوریتم که دارای ۳ فاکتور است سطوحی در نظر گرفته می‌شود. عامل اول که تعداد تکرارهای زنجیره مارکوف است را ۳ سطح، دمای اولیه الگوریتم دارای ۲ سطح و ضریب کاهش دما را ۳ سطح در نظر می‌گیریم. بنابراین به تعداد $3 \times 2 \times 3 = 36$ جایگشت مختلف از جواب‌های الگوریتم را مینا قرار داده که مینای محاسبات قرار می‌گیرد. بعد از اجرای برنامه در نرم افزار متلب با استفاده از داده‌های تصادفی، جدول تحلیل واریانس که مطابق جدول ۲ است به شرح زیر به دست می‌آید:

جدول شماره (۲): جدول تحلیل واریانس

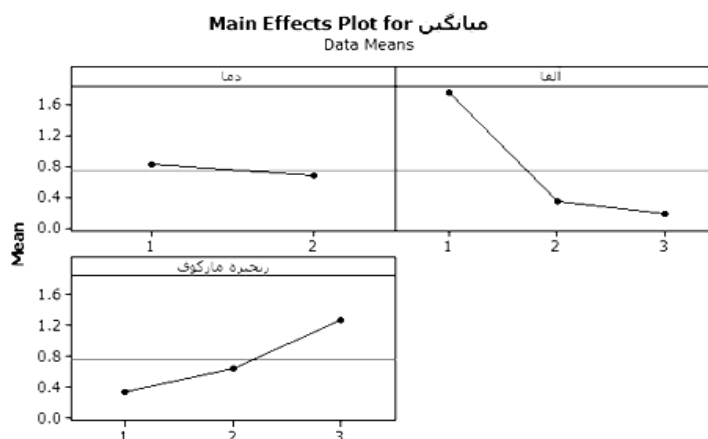
منبع تغییرات	مجموع مربعات	درجه های آزادی	میانگین مربعات	F	سهم پارامتر
خطای A (دما)	۲/۷	۲	۱/۴	۹۱۰۲/۴	۱۹
خطای B (آلفا)	۰/۰۹	۱	۰/۰۹۴	۶۳۷/۲	۰/۷
خطای C (تعداد تکرار)	۸/۹	۲	۴/۴۵	۲۹۹۰۵/۹	۶۲/۴
خطای AB	۰/۰۱۶	۲	۰/۰۰۸	۵۳/۱	۰/۱
خطای BC	۰/۰۹۸	۲	۰/۰۵	۳۳۹/۳	۰/۷
خطای AC	۲/۴۴	۴	۰/۶۱	۴۱۰۱/۲	۱۷/۱
SS _E	۰/۰۰۹	۵۸	۰/۰۰۰۱		۰/۱
کل	۱۴/۲۸۱	۷۱			

همان طور که در جدول ۲ و شکل ۳ مشاهده می‌شود پارامتر تعداد تکرار در زنجیره مارکوف (خطای C) بیشترین سهم را در تغییر جواب‌ها دارد. در تعامل بین پارامترها نیز، تعامل بین BC نیز می‌تواند قابل ملاحظه باشد. حال به وسیله نموداری که از تعاملات بین پارامترها که در نرم افزار به دست آمده است مقادیر مناسب پارامترهای الگوریتم قابل تشخیص می‌باشد.

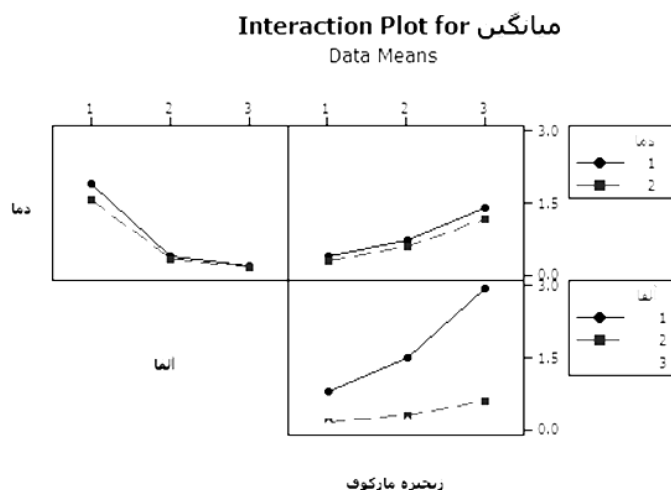


شکل شماره (۳): سهم هر پارامتر در میزان کارا بودن الگوریتم

همان طور که در جدول تحلیل واریانس گفته شد پارامترهای تعداد تکرار در زنجیره مارکوف بیشترین سهم را در تغییر جواب‌های الگوریتم دارند. با توجه به شکل ۴ نیز واضح است که شیب این پارامتر در صورت ثابت بودن دو پارامتر دیگر زیاد است که این سبب تغییر سریع در جواب الگوریتم می‌شود. پارامتر دمای اولیه نیز بی تاثیر نیست ولی نسبت به دو پارامتر دیگر از حساسیت کمتری برخوردار است.



شکل شماره (۴): نمودار اثر پارامترهای الگوریتم به صورت جداگانه



شکل شماره (۵): نمودار اثر تعاملات دوگانه بین پارامترهای الگوریتم

شکل ۵ نمودار اثرات تعاملی بین پارامترها را نشان می‌دهد. در تحلیل این نمودار می‌توان این گونه بیان کرد که مثلاً در نمودار ضریب کاهش دما، دما وقتی که دما ۵۰۰ یا ۱۰۰۰ است، ضریب کاهش دمای ۰/۱ بهترین مقدار تابع هدف را به ما خواهد داد و به همین ترتیب برای سایر نمودارها نیز از چنین استدلالی می‌توان استفاده نمود و پارامترهای الگوریتم را تنظیم نمود. بنابراین با توجه به استدلال‌هایی که صورت گرفت مقادیر $\alpha = 0.1$ و تکرار ۵ و دمای ۱۰۰۰ برای الگوریتم در نظر گرفته خواهد شد. در ضمن پارامترهایی که در فوق بدست آمده است لزوماً برای مسائلی که در اندازه بزرگ طراحی می‌شوند کارایی نداشته و با انجام محاسباتی مشابه، این پارامترها را برای مسائلی با اندازه بیشتر از ۱۵ کار، در جدول ۳ تخمین زدیم.

جدول شماره (۳): پارامترهای تخمین زده شده در اندازه‌های بزرگ

پارامترها	ضریب کاهش دما	تعداد تکرار زنجیره مارکوف	دما
مقدار پارامتر	۰/۰۱	۲۰	۵۰۰ - ۱۰۰۰

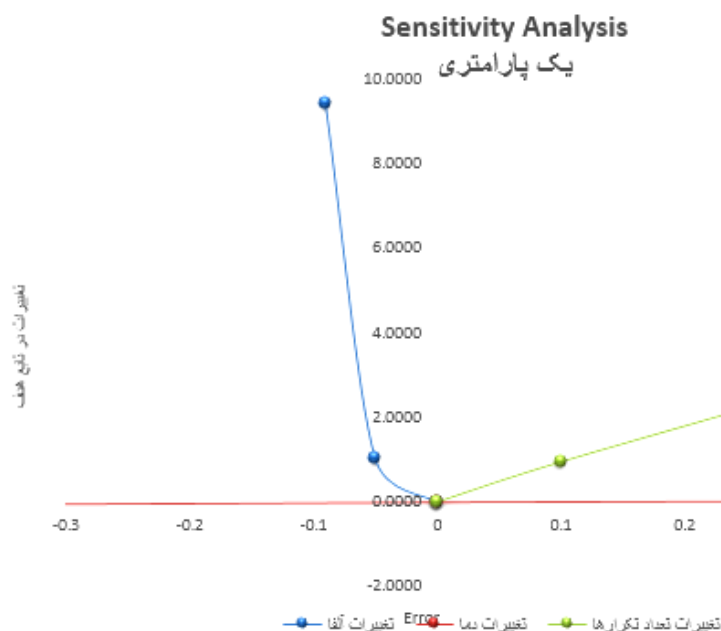
برای ارزیابی کارایی الگوریتم فراابتکاری، نتایج حاصل از آن با نتیجه حاصل از نرم افزار *LINGO* مقایسه شده است. در جدول ۴ داده‌های مسئله با سه مجموعه با اندازه‌های متفاوت و نتایج حاصل از الگوریتم فراابتکاری با روش دقیق مقایسه شده

است.

جدول شماره (۴): مقایسه نتایج

تعداد شهرها	الگوریتم <i>GRASP</i>	<i>LINGO</i>	خطا (درصد)
۷	۵۱	۵۱	۰
۱۰	۵۸	۵۷	۱/۷۵
۱۲	۵۹	۵۸	۱/۷۲

در توضیح مقایسه دو روش همان طور که در جدول ۴ مشاهده می شود می توان نتیجه گرفت که جواب های به دست آمده از الگوریتم *GRASP* در مقایسه با جواب دقیق حاصل از نرم افزار *LINGO* از خطای محاسباتی بسیار کمی برخوردار است و جواب های قابل قبولی را نتیجه می دهد. بنابراین زمان اجرای برنامه معیاری برای تنظیم پارامتر الگوریتم خواهد بود، چراکه جواب های الگوریتم به جواب دقیق بسیار نزدیک بوده و از آن نمی توان به عنوان معیار کارایی الگوریتم استفاده نمود. در تحلیل حساسیت پارامترهای الگوریتم، مقدار به دست آمده توسط تنظیم پارامتر که توسط روش فاکتوریل صورت گرفت را به عنوان معیار در نظر گرفته و با تغییر مقدار پارامتر میزان تغییر تابع هدف برای اولین مثال موجود در جدول ۴ سنجیده می شود. با توجه به نمودار تحلیل حساسیت در شکل ۶ همانطور که از قبل انتظار می رفت میزان حساسیت پارامترهای ضریب کاهش دما و تعداد تکرارها در هر زنجیره نسبت به پارامتر دما از حساسیت بیشتری برخوردار بوده و تغییر کم در مقدار آن ها میزان بیشتری در مقدار تابع هدف تغییر ایجاد می کنند. لازم به ذکر است که این تحلیل حساسیت تک پارامتری بوده یعنی با ثابت نگه داشتن دو عامل دیگر تغییرات روی یک فاکتور اعمال شده و تغییرات آن محاسبه می شود.



شکل شماره (۶): نمودار تحلیل حساسیت تک پارامتری

الگوریتم پیشنهادی با الگوریتم کلونی مورچگان در دو اندازه بزرگ و کوچک مقایسه و نتایج آن به صورت زیر حاصل شده است. پارامترهای الگوریتم کلونی مورچگان در هر دو نوع مسئله بعد از استفاده از روش فاکتوریل، یکسان بدست آمده است و مقادیر آن

در جدول ۵ آورده شده است.

جدول شماره (۵): پارامترهای الگوریتم کلونی مورچگان

پارامتر	اهمیت جذابیت حرکت	اهمیت نسبی غلظت فرومون	مقدار فرومون اولیه	ماکزیمم تعداد تکرارها
مقدار پارامتر	۲	۱	۰/۰۱	۲۰۰

حال برای مقایسه الگوریتم پیشنهادی با الگوریتم کلونی مورچگان در اندازه کوچک از ۲ مثال که در جدول ۶ آمده، استفاده شده است. لازم به ذکر است که مقایسه این دو الگوریتم در دو اندازه کوچک و بزرگ انجام شده و نتایج حاصل از آن متفاوت بدست آمده است.

جدول شماره (۶): مقایسه الگوریتم پیشنهادی در اندازه کوچک

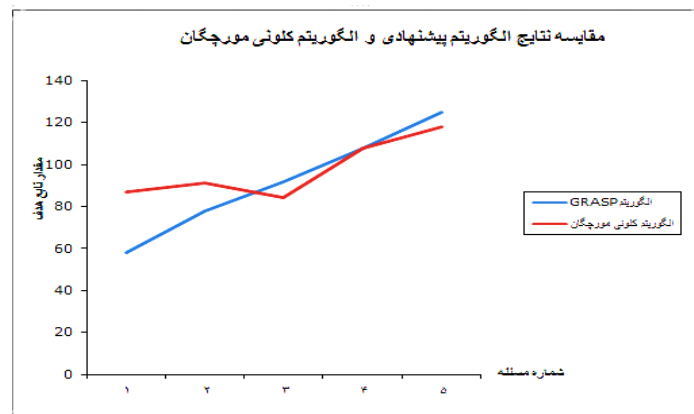
شماره مسئله	جواب‌های بدست آمده از الگوریتم GRASP		جواب‌های بدست آمده از الگوریتم کلونی مورچگان	
	تعداد شهر	بهترین جواب	متوسط زمان (ثانیه)	بهترین جواب
۱	۷	۴۳	۱۱/۶۳	۸۳
۲	۹	۵۱	۱۴/۹۲	۸۶
۳	۱۰	۵۸	۱۵/۶۱	۸۷
۴	۱۲	۶۴	۱۷/۲	۸۹
۵	۱۵	۷۸	۲۰/۴۴	۹۱

همانطور که در جدول ۶ مشاهده می‌شود کارایی الگوریتم پیشنهادی در مقایسه با الگوریتم کلونی در اندازه کوچک بهتر است. در این مطالعه بیشتر از ۱۵ کار را به عنوان اندازه بزرگ در نظر گرفته شده است چرا که نتایج بدست آمده از الگوریتم پیشنهادی نتایج متفاوتی نسبت به الگوریتم کلونی مورچگان نشان می‌دهد و نتایج حاصل از مسئله در اندازه بزرگ در جدول ۷ آمده است.

جدول شماره (۷): مقایسه الگوریتم پیشنهادی در اندازه بزرگ

شماره مسئله	جواب‌های بدست آمده از الگوریتم GRASP		جواب‌های بدست آمده از الگوریتم کلونی مورچگان	
	تعداد شهر	بهترین جواب	متوسط زمان (ثانیه)	بهترین جواب
۱	۲۰	۹۲	۲۲/۶۴	۸۴
۲	۲۵	۱۰۸	۲۷/۴۵	۱۰۸
۳	۳۰	۱۲۵	۲۷/۶	۱۱۸

همانطور که در جدول ۷ و شکل ۷ مشاهده می‌شود الگوریتم پیشنهادی در مقایسه با الگوریتم کلونی مورچگان در اندازه‌های کوچکی از مسئله دارای کارایی بهتری بوده و در اندازه‌های بزرگی از مسئله، الگوریتم کلونی مورچگان در مقایسه با الگوریتم پیشنهادی از کارایی بهتری برخوردار است.



شکل شماره (۷): نمودار مقایسه نتایج الگوریتم پیشنهادی و الگوریتم کلونی مورچگان

در این مقاله، یکی از مسائل پرکاربرد توالی عملیات مورد بررسی قرار گرفت. این مسئله به دلیل پیچیدگی و حجم بالای محاسبات جزء مسائل *NP-hard* می‌باشد. به همین دلیل از الگوریتم‌های فراابتکاری برای حل آن استفاده می‌شود. در این مطالعه روش جدیدی برای تبدیل مسئله جریان کارگاهی بدون صف‌های میانی به مسئله فروشنده دوره‌گرد و حل آن توسط الگوریتم *GRASP* ارائه گردید و در نهایت نتایج حاصل از آن با الگوریتم کلونی مورچگان مقایسه شد. استفاده از روش‌های دیگر برای بدست آوردن مقادیر پارامترهای مناسب برای اجرای الگوریتم و استفاده از سایر الگوریتم‌های فراابتکاری برای حل، برای تحقیقات آتی پیشنهاد می‌شود.

۴- منابع

1. Aldowaisan, T., & Allahverdi, A. (1998). Total flowtime in no-wait flowshops with separated setup times. *Computers & Operations Research*. 25(9), 757-765.
2. Baker, K.R., Trietsch, D. (2009). *Principles of Sequencing and Scheduling*, Wiley.
3. Bertolissi, E. (2000). Heuristic algorithm for scheduling in the no-wait flow-shop.
4. Feo, T.A., & Resende, M. G. (1995). Greedy randomized adaptive search procedures. *Journal of global optimization*. 6(2), 109-133.
5. Festa, P. (2003). Greedy randomized adaptive search procedures. *AIROnews*, 7(4), 7-11.
6. Gupta, J. N., & Stafford E. F. (2006). Flowshop scheduling research after five decades. *European Journal of Operational Research*. 169(3), 699-711.
7. Hahsler, M., & Hornik, H. (2009). Introduction to TSP - infrastructure for the traveling salesperson problem. *Journal of Statistical Software*. 23, 1-21.
8. Hall, N. G., & Sriskandarajah, C. (1996). A survey of machine scheduling problems with blocking and no-wait in process. *Operations research*. 44(3), 510-525.
9. Hejazi, S. R., & Saghafian, S. (2005). Flowshop-scheduling problems with make span criterion: a review. *International Journal of Production Research*. 43(14), 2895-2929.
10. Jaillet, P. (1988). A priori solution of a traveling salesman problem in which a random subset of the customers are visited. *Operations Research*, 36(6). 929-936.
11. Jayaswal, S. (2008). A comparative study of tabu search and simulated annealing for traveling salesman problem. Department of Management Sciences, University of Waterloo. Project Report.
12. Johnson, D.S., & McGeoch, L. A. (1997). The traveling salesman problem: A case study in local optimization. *Local search in combinatorial optimization*. 1, 215-310.
13. Layeb, A., Ammi, M., & Chikhi, S. (2013). A GRASP Algorithm Based on New Randomized Heuristic for Vehicle Routing Problem. *CIT. Journal of Computing and Information Technology*. 21(1), 35-46.
14. Li, X, Wang, Q, & Wu, C. (2008). Heuristic for no-wait flow shops with make span minimization. *International Journal of Production Research*. 46(9), 2519-2530.
15. Pan, Q. K., Wang, L., Tasgetiren, M. F., & Zhao, B. H. (2008). A hybrid discrete particle swarm optimization algorithm for the no-wait flow shop scheduling problem with makespan criterion. *The International Journal of Advanced Manufacturing Technology*. 38(3-4), 337-347.
16. Papadimitriou, C. H., & Kanellakis, P. C. (1980). Flowshop scheduling with limited temporary storage. *Journal of the ACM*. 27(3), 533-549.
17. Rajendran, C., & Chaudhuri, D. (1990). Heuristic algorithms for continuous flow-shop problem. *Naval Research Logistics*. 37(5), 695-705.

18. Reddi, S. S., & Ramamoorthy, C. V. (1972). On the flow-shop sequencing problem with no wait in process. *Operational Research Quarterly*. 323-331.
19. Selen, W.J., & Hott, D.D. (1986). A new formulation and solution of the flow shop scheduling problem with no in process waiting. *Applied Mathematical Modeling*. 10, 246-248.
20. Tseng, L.Y., & Lin, Y.T. (2010). A hybrid genetic algorithm for no-wait flowshop scheduling problem. *International Journal of Production Economics*. 128(1), 144-152.
21. Xie, J., Xing, W., Liu, Z., & Dong, J. (2004). Minimum deviation algorithm for two-stage no-wait flow shops with parallel machines. *Computers & Mathematics with Applications*. 47(12), 1857-1863.