مجله تحقیق در عملیات در کاربردهای آن، سال دوازدهم، شماره چهارم (پیاپی ۴۷)، زمستان ۹۴، ص ص ۱۱۵-۹۹
شاپا ۷۲۸۶-۲۲۵۱

توسعه یک روش کارا برای بهینه‌سازی مساله چیدمان تسهیلات پویا با محدودیت بودجه

حامد فرجی*[1]، بهمن نادری[2]
۱- کارشناس ارشد، دانشگاه آزاد اسلامی، واحد قزوین، گروه مهندسی صنایع، قزوین، ایران
۲- استادیار، دانشگاه آزاد اسلامی، واحد قزوین، گروه صنایع، قزوین، ایران

رسید مقاله: ۱۶ اردیبهشت ۱۳۹۴
پذیرش مقاله: ۲۸ مهر ۱۳۹۴

چکیده
هدف مساله چیدمان پویای تسهیلات، یافتن بهترین چیدمان برای تسهیلات تولیدی در یک افق زمانی با تعدادی دوره زمانی مشخص است. یکی از مهم‌ترین ابعاد مساله، در نظرگرفتن محدودیت بودجه به هنگام جابه‌جایی تسهیلات است. بر طبق نتایج به‌دست آمده از وزارت صنایع ایالات متحده، طراحی چیدمان، اولویت کم‌تری در بیش‌تر سازمان‌ها دارد و بسیاری از طرح‌ها به علت کمبود بودجه غیرعملی خواهد بود؛ بنابراین رعایت بودجه تخصیص داده شده، به‌هنگام طراحی چیدمان ضروری به‌نظر می‌رسد. به‌دلیل ساختار پیچیده و وسیع این مساله، تاکنون تحقیقات کمی برای حل این مساله صورت گرفته است و در تمامی این تحقیقات، هدف، یافتن جواب نزدیک به بهینه از طریق توسعه یک الگوریتم فراابتکاری بوده است. در این تحقیق برای اولین بار دو الگوریتم فراابتکاری ژنتیک و شبیه‌سازی تبرید موازی برای حل این مساله به‌طور همزمان مورد استفاده قرارگرفته و نتایج در مورد طیف وسیعی از مسایل موجود در ادبیات تحقیق به ثبت رسیده است. نتایج محاسباتی نشان می‌دهد که الگوریتم پیشنهادی می‌تواند کیفیت جواب‌هایی که تاکنون توسط محققین قبلی ارایه شده است بهبود بخشد.

کلمات کلیدی: مساله چیدمان تسهیلات چنددوره‌ای، محدودیت بودجه، الگوریتم ژنتیک، الگوریتم شبیه‌ساز تبرید.

۱ مقدمه
هدف مساله چیدمان تسهیلات، تعیین کارآمدترین چیدمان تسهیلات در یک سیستم است. تسهیل می‌تواند یک واحد تولیدی، ساختمان اداری و یا ماشین‌های درون کارخانه باشد. چیدمان مناسب می‌تواند منجر به ایجاد جریان موادکارا بین تسهیلات و در نتیجه کاهش مواد در جریان و هزینه نگهداری مواد شود. همچنین به‌وسیله چیدمان مناسب می‌توان هزینه‌ی حمل مواد بین تسهیلات را کاهش داد.

* عهده‌دار مکاتبات
آدرس الکترونیکی: hfaraji_ie_89@yahoo.com

[Downloaded from jorlu.ir on 2026-11-29]

هزینه حمل مواد مهم‌ترین شاخص برای تعیین کارایی الگوی چیدمان است و از آنجایی که ۲۰ تا ۴۰ درصد هزینه‌های عملیاتی و ۱۵ تا ۷۰ درصد کل هزینه‌های سیستم تولیدی به آن وابسته است [۱]، اغلب به‌عنوان شاخص اصلی کارایی درنظر گرفته می‌شود. در صورت عدم بهینه بودن چیدمان، مبالغ هنگفتی صرف حمل و نقل مواد و در نتیجه افزایش قیمت تمام شده محصول می‌شود. هزینه‌ی حمل، به میزان جریان و فاصله بین تسهیلات بستگی دارد. همچنین نوع سیستم حمل مورد استفاده نیز بر روی هزینه‌ی حمل تأثیر می‌گذارد. به‌عبارت دیگر نوع سیستم حمل مواد، الگوی چیدمان را تحت تأثیر قرار می‌دهد و متقابلاً الگوی چیدمان نیز بر نوع سیستم حمل مواد تأثیرگذار خواهد بود. در شرایط واقعی، درصورتی‌که جریان حمل و نقل مواد بین بخش‌ها برای مدت زمان زیادی تغییر نکند، مسایل چیدمان ایستا می‌تواند جواب‌گوی نیاز ما باشد؛ اما وقتی تقاضا تغییر می‌کند باعث تغییر در جریان مواد می‌شود. در نتیجه چیدمان فعلی کارایی خود را از دست می‌دهد و نیاز به تغییر پیدا می‌کند تا با شرایط جدید هماهنگ شود. ارزیابی مجدد چیدمان، زمانی صورت می‌گیرد که مواد در جریان، افزایش یا کاهش یابند. چیدمان مجدد تسهیلات یک کارخانه به‌منظور به حداقل رساندن مجموع هزینه‌های حمل مواد و جابه‌جایی تسهیلات را مساله چیدمان تسهیلات به شکل پویا گویند.

نیاز به تغییر چیدمان در هرکارخانه‌ای به یکی از دلایل زیر است [۲]:

- تغییر در تقاضا و یا ترکیب محصول
- تغییر در توالی عملیات
- تغییر در استراتژی‌های تولید
- تغییر در تجهیزات و منابع
- تغییر در قوانین و استانداردهای ایمنی

تغییر در تقاضای محصول مهم‌ترین عامل تغییر چیدمان است که در نتیجه آن هزینه‌ی حمل دچار نوسان شده، اغلب افزایش می‌یابد. به علاوه معرفی محصول-ماشین جدید و یا اسقاط محصول-ماشین می‌تواند موجب تغییر در الگوی حمل مواد شود. تغییرات به‌وجود آمده باعث می‌شود تا چیدمان فعلی کارایی خود را از دست بدهد و هزینه‌های اضافی را بر سیستم تحمیل نماید. برای حل این مشکل، چیدمان باید انعطاف‌پذیر و قابل تغییر باشد. توضیح بیشتر این که در یک کارخانه‌ی تولیدی، تولید هر محصول نیازمند انتقال مواد اولیه و کار در جریان، بین تسهیلات است. این انتقال مواد در هر دوره‌ی برنامه‌ریزی دارای هزینه است. اگر در دوره‌های مختلف برنامه‌ریزی، سازمان مجبور به تغییر برنامه و یا تغییر محصول یا فرایند باشد، مسیر انتقال مواد بین تسهیلات دچار تغییر و تحولاتی خواهد شد که ممکن است هزینه‌های بیشتری را بر سیستم تحمیل نماید. برای کاستن از این هزینه‌های انتقال، تغییر چیدمان تجهیزات مطرح می‌گردد. باید توجه داشت که همین تغییر چیدمان نیز به نوبه‌ی خود دارای هزینه می‌باشد. یک مساله چیدمان تجهیزات پویا به دنبال چیدمانی است که به موجب آن، هزینه‌های جابه‌جایی و هزینه‌های انتقال مواد بین تجهیزات کمینه گردد. این انتخاب به گونه‌ای خواهد بود که تغییرات جریان مواد را در دوره‌های آتی برنامه‌ریزی، در نظر گیرد و نیاز به تغییر در آرایش تجهیزات در دوره‌های بعدی را در پایین‌ترین سطح نگاه دارد.

[Downloaded from jorlu.ir on 2026-11-29]

به عقیده شیخ‌زاده و بن جعفر [۳] ایجاد یک طرح چیدمان خوب برای چند دوره، نیازمند تخمین پیوسته جریان مواد بین ماشین‌هاست. اطلاعات جریان مواد برای هر دوره پیش‌بینی می‌گردد و فرض می‌شود که جریان مواد در طول هر دوره ثابت است؛ بنابراین مساله چیدمان برای هر دوره را می‌توان تحت عنوان یک مساله چیدمان تسهلات ایستا در نظرگرفته، حل کرد. زمانی که مساله را به‌صورت دوره‌های جداگانه حل می‌کنیم، هزینه جابه‌جایی تسهیلات در نظرگرفته نمی‌شود.

بنابراین مسایل جداگانه SFLP باید به شکل یک مساله DFLP تبدیل شود. یک جواب برای DFLP شامل مجموعه‌ای از طرح‌هاست که هر یک مربوط به یک دوره‌ی زمانی است. بهینه کردن مسایل چیدمان تسهیلات پویا به دو دلیل دارای اهمیت است: اولا هزینه‌های حمل و نقل مواد از ۱۵ تا ۷۰ درصد کل هزینه‌های تولیدی را دربر می‌گیرد؛ بنابراین هرگونه کاهش در این هزینه‌ها از طریق آرایش بهتر تسهیلات، مشارکتی مستقیم در بهبود اثربخشی کلی عملیات محسوب می‌شود. ثانیاً تغییر طرح چیدمان بسیار هزینه‌بر است و هرگونه تغییرات آتی در آرایش تسهیلات باید از قبل برنامه‌ریزی شود.

از این رو اصولاً مسایل چیدمان پویا، به دنبال تعیین طرح چیدمان برای هر دوره در افق برنامه‌ریزی، برای کمینه کردن مجموع هزینه‌های انتقال مواد، برای تمام دوره‌ها و مجموع هزینه‌های آرایش مجدد بین دوره‌های زمانی می‌باشد. هزینه‌های آرایش مجدد باید زمانی که تسهیلات نیاز به تغییر مکان دارند مد نظر قرار گیرد.

بنابراین هزینه‌ی کل، شامل هزینه‌های حمل مواد در همه دوره‌ها و هزینه‌های جابه‌جایی ماشین‌هاست. در سال‌های اخیر تلاش‌های زیادی برای حل مساله چیدمان تسهیلات به‌شکل پویا انجام گرفته است. رزنبلات [۴] برای اولین بار مساله چیدمان پویا را مطرح کرد و با در نظرگرفتن هزینه‌های حمل مواد و جابه‌جایی ماشین‌ها آن‌را حل کرد و بعدها محققان با رویکردهای مختلف به بررسی ابعاد گوناگون آن پرداختند.

وینچه عبداله و قاسمی [۵] یک مساله بهینه‌سازی چیدمان تسهیلات چندهدفه را با اهداف حداکثر کردن بهره‌وری و سودآوری عرضه کردندکه درآن علاوه‌بر معیارهای کمی، معیارهای کیفی نیز مورد بررسی قرار گرفت. آن‌ها از رویکرد تحلیل سلسله مراتبی (AHP)جهت تعیین مقادیر عملکردی معیارهای کیفی استفاده کردند. کارایی روش پیشنهادی و مدل ریاضی از طریق یک مطالعه موردی به اثبات رسید.

یکی از مهم‌ترین ابعاد مساله در نظرگرفتن محدودیت بودجه به‌هنگام جابه‌جایی تسهیلات است. همان‌طور که اشاره شد تغییر چیدمان نیازمند صرف بودجه هنگفتی است؛ بنابراین یکی از جنبه‌های واقع‌گرایانه مساله، رعایت سقف بودجه به‌هنگام مدل‌سازی و حل مساله است. از این‌رو در این نوشتار تلاش می‌گردد تا الگوریتمی ارایه شود که مسایل چیدمان تسهیلات پویا را با درنظرگرفتن محدودیت بودجه به‌گونه‌ای کارا حل کند.

باقیمانده این نوشتار به‌صورت زیر سازماندهی شده است: در بخش بعد مروری کلی خواهیم داشت بر پیشینه مسایل چیدمان تسهیلات پویا. بخش سوم، ارتباطات ریاضی و مدل سازی مساله را تشریح می‌کند. بخش چهارم به طرح الگوریتم پیشنهادی می‌پردازد و مراحل آن‌را با جزییات و بعضاً در قالب مثال تشریح می‌کند. در بخش پنجم پس از تنظیم پارامترهای الگوریتم پیشنهادی، نتایج حاصل از به‌کارگیری الگوریتم پیشنهادی ارایه می‌گردد و

[Downloaded from jorlu.ir on 2026-11-29]

نتایج به‌دست آمده برای اثبات کارایی الگوریتم با روش‌های موجود در ادبیات استفاده می‌شود. در نهایت در بخش پایانی نتیجه‌گیری و پیشنهادهایی برای کارهای آتی بیان می‌گردد.

۲ مروری بر تحقیقات مشابه

برطبق تحقیقات موجود، دو دیدگاه روش‌های دقیق و روش‌های ابتکاری برای حل DFLP وجود دارد. از آنجا که این مساله از نظر محاسباتی بسیار پیچیده است، بیش‌تر تلاش‌ها برای ایجاد و گسترش روش‌های ابتکاری می‌باشدکه در زمان محاسباتی قابل قبول، به جواب مناسب می‌رسند. بالاکریشنان [۶] یک بررسی جامع بر روی روش‌های حل DFLP انجام داده است. رزنبلات [۴] از برنامه‌ریزی پویا برای حل DFLP استفاده کرد. در تکنیک برنامه‌ریزی پویا، هر دوره به‌عنوان یک مرحله در نظرگرفته می‌شود و به طرح چیدمان هر دوره، وضعیت گفته می‌شود. یک طرح برای یک دوره از میان یک سری طرح‌های پیشنهادی به گونه‌ای انتخاب می‌شود که بیش‌ترین بهبود را در تابع هدف داشته باشد؛ بنابراین برنامه‌ریزی پویا برای حل مسایل بزرگ دارای پیچیدگی‌های محاسباتی فراوان خواهد بود. رزنبلات[۴] برای تقلیل این پیچیدگی از دو روش استفاده کرد:

۱) روش اول بر پایه ی روش ابتکاری بالو [۷] است که در آن از برنامه‌ریزی پویا برای حل مساله چیدمان انبار استفاده شده است.

۲) روش دوم تولید طرح‌های چیدمان به شکل تصادفی است.

هر دو روش بر روی مسایل در ابعاد کوچک آزمایش شد و نتایج به‌دست آمده ثابت کرد که روش اول کارآمدتر است. لاکسونن و انسکر [۸] پنج الگوریتم تحت عناوین: کرفت (CRAFT)، صفحات برشی، شاخه و حد، برنامه‌ریزی پویا و درختان برشی برای حل DFLP پیشنهاد دادند. هر پنج الگوریتم معرفی شده، با استفاده از داده‌های نمونه‌ای ارزیابی شدندکه از میان آن‌ها روش صفحات برشی عملکرد بهتری نسبت به سایر الگوریتم‌ها داشت.

آربن [۹] از الگوریتم جابه‌جایی دوتایی برای حل DFLP استفاده کرد. وی در این روش از اصل دریچه‌های پیش بینی استفاده کرد. مفهوم این تکنیک این است که به هر الگوی چیدمان اجازه بدهیم تا در همه‌ی دوره‌ها اجرا شود و هزینه‌ی جابه‌جایی را درنظر نگیریم . بالاکریشنان و چنگ[۱۰] دو روش برای بهبود الگوریتم جابه‌جایی دوتایی آربن [۹] ارایه کردند. در روش اول جواب نهایی الگوریتم آربن [۹]، به عنوان جواب نهایی معرفی شده و از طریق روش بازگشتی به جستجوی جواب‌های بهتر پرداخته می‌شود. در روش دوم الگوریتم ارایه شده در [۹] با برنامه‌ریزی‌پویا ترکیب می‌گردد. کانوی و ونکاتارامنان [۱۱] برای اولین بار مساله DFLP را با استفاده از الگوریتم ژنتیک حل کردند. بالاکریشنان و چنگ [۱۰] روش حل ارایه شده توسط [۱۱] را بهبود دادند. الگوریتم آنها شامل دو حلقه تو در توی داخلی و خارجی بود. در حلقه داخلی، عملگرهای تقاطع و جهش برای تولید فرزندان به کار می‌رفتند، سپس جواب‌های موجه انتخاب شده و آن‌هایی که کم‌ترین هزینه را داشتند با جواب‌های دارای بیش‌ترین هزینه جایگزین می‌شدند. در حلقه خارجی، بعضی جواب‌های ضعیف با جواب‌های تصادفی جایگزین می‌شدند. حلقه خارجی باعث می‌شد تا حلقه داخلی کار خود را با یک سری جواب‌های مشابه

[Downloaded from jorlu.ir on 2026-11-29]

ادامه ندهد و به این ترتیب تنوع جمعیت را افزایش می‌داد. چنگ و همکاران [۱۲] الگوریتم ژنتیک ابتکاری را برای حل DFLP معرفی کردند. در این روش هر جواب از جمعیت، انتخاب شده و با قرار دادن طرح‌های چیدمان یک دوره‌ای در آن ارزیابی می‌گردد. نتایج به‌دست آمده نشان دادکه این روش در بعضی شرایط کاراست. دانکار و همکاران [۱۳] از ترکیب الگوریتم ژنتیک با برنامه‌ریزی پویا استفاده کردند. کاکو و مازولا [۱۴] یک الگوریتم جستجوی ممنوعه (TS) ابتکاری برای حل DFLP به‌کار بردند. TS در دو مرحله اجرا شد که مکانیزم‌های تنوع و تمرکز را به خوبی رعایت می‌کرد. آن‌ها از سه مکانیزم تولید طرح‌های تصادفی اولیه، ایجاد طرح‌های اولیه با استفاده از یک روش ابتکاری و ایجاد تنوع بر پایه‌ی فراوانی بعضی از جواب‌های موجود در لیست ممنوعه استفاده کردند. الگوریتم پیشنهادی بر روی داده‌هاآزمایش شد و اثبات گردید که روش ارایه شده نسبت به روش [۹] عملکرد بهتری دارد. بایکوسالگو و گیندی [۱۵] اولین کسانی بودند که از الگوریتم شبیه‌سازی تبرید (SA) برای حل DFLP استفاده کردند. الگوریتم آنها همان الگوریتم SA سنتی بود که در آن ابتدا یک طرح چیدمان به عنوان جواب اولیه انتخاب شده و برای بهبود، به الگوریتم داده می‌شد. در روش آن‌ها، برای ایجاد همسایگی، ابتدا یک دوره و دو تسهیل به طور تصادفی انتخاب می‌شوند؛ سپس تسهیلات انتخاب شده موقعیت خود را با هم عوض می‌کنند. مک کندال و همکاران [۱۶] الگوریتم SA ابتکاری را برای حل مساله DFLP معرفی کردند. در روش اول، آن‌ها SA را به طور سنتی اجرا کردند. ابتکار آن‌ها در انتخاب پارامترهای موثر بر الگوریتم بود. در روش دوم، آن‌ها الگوریتم را با یک استراتژی کنترل مراحل گذشته و کنترل مراحل پیش‌رو ترکیب کردند. ارل و همکاران [۱۷] یک دیدگاه سه مرحله‌ای را برای حل DFLP مطرح کردند. در مرحله اول ماتریس‌های جریان با وزن‌های مختلفی با هم ترکیب می‌شد. در نتیجه مساله از حالت پویا به حالت ایستا تبدیل می‌گردید و یک سری جواب‌های ایده‌آل ایستا را مشخص می‌کرد. در مرحله دوم با استفاده از جواب‌های مرحله اول و روش برنامه‌ریزی پویا، طرح‌های چیدمان چند دوره‌ای تعیین می‌شد و در مرحله سوم جواب‌های به‌دست آمده در مرحله قبل با استفاده از روش جابه‌جایی دوتایی بهبود می‌یافت. بایکاسگلو و همکاران [۱۸] نتایج به‌دست آمده از الگوریتم‌های ارایه شده را برای مساله ابعاد کوچک با هم مقایسه کردند و به این نتیجه رسیدند که تفاوت معنی‌داری بین جواب‌ها وجود ندارد. اما برای مسایل با ابعاد وسیع الگوریتم چنگ و بالاکریشنان [۱۰] عملکرد نسبتاً بهتری دارد. بالاکریشنان و همکاران [۱۹] برای اولین بار مساله چیدمان پویای تسهیلات را با محدودیت بودجه معرفی کرد و چگونگی اعمال این محدودیت را در مدل نشان دادند. بالاکریشنان یک بودجه‌ی خاص برای همه‌ی دوره‌های زمانی درنظر گرفت و از این طریق جابه‌جایی تسهیلات در دوره‌های زمانی مختلف را محدود کرد. میزان بودجه تخصیص داده شده از روش زیر به‌دست می‌آمد. ابتدا مساله، بدون محدودیت بودجه، در نظرگرفته، حل می‌شود؛ سپس کسری از بودجه تخصیص داده شده برای جابه‌جایی تسهیلات به عنوان محدودیت بودجه در نظر گرفته می‌شود. بالاکریشنان از دو مقدار ۹۰ درصد در حالت سهل‌گیرانه و ۵۰ درصد در حالت سخت گیرانه استفاده کرد. ساهین و همکاران [۲۰] مساله ارایه شده [۱۸] را با الگوریتم شبیه‌سازی تبرید حل کردند. آن‌ها روش متفاوتی را برای تعیین پارامترهای اولیه اتخاذ کردند. به‌علاوه جواب اولیه نیز از طریق یک روش ابتکاری تعیین شد.

[Downloaded from jorlu.ir on 2026-11-29]

۳ فرمول‌بندی مساله

مساله DFLP می‌تواند به شکل مساله تخصیص نمایی مدل شود. متغیرهای مورد استفاده در این مساله به‌قرار زیر است:

X_{tij} ۱ اگر تسهیل i در دوره t به موقعیت j تخصیص پیدا کند

۰ در غیر اینصورت

۳-۱ نمادها و پارامترهای مساله

نمادها و پارامترهای در نظر گرفته شده در این مساله به شرح زیر است:

جدول ۱. پارامترهای مدل

N	تعداد تسهیلات یا موقعیت‌ها
T	تعداد دوره‌های برنامه‌ریزی شده
C_{tijkl}	هزینه حمل مواد در دوره t بین تسهیل i که در موقعیت j است و تسهیل k که در موقعیت l است.
A_{tijl}	هزینه جابه‌جایی تسهیل i از موقعیت j به l در دوره t
LB_t	بودجه باقی مانده از دوره t برای دوره $t+۱$
B_t	بودجه در دسترس برای دوره t
AB_t	بودجه تخصیص داده شده برای دوره t
N	تعداد تسهیلات یا موقعیت‌ها
T	تعداد دوره‌های برنامه‌ریزی شده
C_{tijkl}	هزینه حمل مواد در دوره t بین تسهیل i که در موقعیت j است و تسهیل k که در موقعیت l است.
A_{tijl}	هزینه جابه‌جایی تسهیل i از موقعیت j به l در دوره t

۳-۲ مدل مساله

$$Min.Z = \left(\sum_{t=۲}^{T} \sum_{i=۱}^{N} \sum_{j=۱}^{N} \sum_{l=۱}^{N} A_{tijl} * X_{t-۱,ij} * X_{til} + \sum_{t=۱}^{T} \sum_{i=۱}^{N} \sum_{j=۱}^{N} \sum_{k=۱}^{N} \sum_{l=۱}^{N} C_{tijkl} * X_{tij} * X_{tkl} \right) \quad (۱)$$

s.t.

$$\sum_{j=۱}^{N} X_{tij} = ۱ , \quad \forall i = ۱,۲,\ldots,N \ , \ \forall t = ۱,۲,\ldots,T \ , \quad (۲)$$

[Downloaded from jorlu.ir on 2026-11-29]

$$\sum_{i=1}^{N} X_{tij} = 1 , \quad \forall j = 1,2,\ldots,N, \quad \forall t = 1,2,\ldots,T , \tag{٣}$$

$$LB_t = B_t - \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{l=1}^{N} A_{tijl} * X_{t-1,ij} * X_{til} , , \forall t = 1,2,\ldots,T, \tag{٤}$$

$$B_t = AB_t + LB_{t-1}, \quad \forall t = 1,2,\ldots,T, \tag{٥}$$

$$\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{l=1}^{N} A_{tijl} * X_{t-1,ij} * X_{til} \le B_t, \quad , \forall t = 1,2,\ldots,T, \tag{٦}$$

$$X_{tij} \in \{0,1\} \ , \forall i, j = 1,2,\ldots,N \ , \forall t = 1,2,\ldots,T, \tag{٧}$$

$$LB_t \ , B_t , AB_t \ge 0, \quad \forall t = 1,2,\ldots,T, \tag{٨}$$

در این مساله تابع هدف (١) سعی می‌کند تا مجموع هزینه‌های جابه‌جایی و حمل مواد را حداقل کند. محدودیت (٢) تضمین می‌کندکه هر تسهیل فقط در یک موقعیت قرار گیرد و محدودیت (٣) تضمین می‌کند که در هر موقعیت فقط یک تسهیل قرار بگیرد. همچنین محدودیت (۴) بیان می‌کند که بودجه‌ی منتقل شده به دوره بعد برابر بودجه‌ی دردست دوره فعلی منهای هزینه‌های جابه‌جایی در دوره فعلی است. محدودیت (۵) بودجه‌ی در دسترس برای هر دوره را برابر مجموع بودجه تخصیص داده شده به آن دوره و بودجه باقی مانده از دوره قبل قرار می‌دهد و محدودیت (۶) سقف بودجه در هر دوره را نشان می‌دهد.

می‌توان نشان دادکه در یک مساله‌ی چیدمان تسهیلات پویا با N دپارتمان و T دوره برنامه‌ریزی، $(N!)^T$ طرح، برای یافتن جواب بهینه باید ارزیابی گردد. با یک محاسبه ساده می‌توان دریافت که حتی برای یک مساله با ۶ دپارتمان و ۵ دوره، ۱۰۱۴×۱/۹۳ ترکیب (طرح چیدمان) مختلف امکان‌پذیر است که باید مورد بررسی قرار گیرد که در واقع مساله‌ای است بسیار بزرگ؛ بنابراین مساله معرفی شده، جزء مسایل دسته NP-Hard است [٢١] و این موضوع حل این مساله را دچار مشکل می‌کند و ضرورت توسعه الگوریتم های کارا را برای حل این مساله مطرح می‌سازد [٧].

٤ الگوریتم حل

همان‌گونه که ذکر شد به علت NP-hard بودن مساله، استفاده از روش‌های دقیق برای حل مساله در زمان معقول ممکن نیست و باید از روش‌های ابتکاری یا فراابتکاری برای حل مساله استفاده کرد. به‌علت این‌که در ادبیات موضوع، استفاده از الگوریتم ژنتیک از اقبال بیشتری نسبت به سایر روش‌های فراابتکاری برخوردار بوده است، در این نوشتار نیز سعی می‌شود تا با به‌کارگیری یک الگوریتم ترکیبی ژنتیک- شبیه‌ساز تبرید موازی، کیفیت جواب‌هایی که تاکنون توسط محققین قبلی ارایه شده است بهبود داده شود.

[Downloaded from jorlu.ir on 2026-11-29]

٤-١ الگوریتم ژنتیک

الگوریتم ژنتیک یک جستجوی تصادفی قدرتمند بر پایه مکانیسم انتخاب طبیعی است. مبنای الگوریتم که برگرفته از طبیعت است، استفاده از جستجوی تصادفی برای بهینه‌سازی مسایل و فرآیندهای یادگیری است. در طبیعت از ترکیب کروموزوم‌های مناسب نسل‌های بهتری پدید می‌آید. در این بین گاهی جهش‌هایی نیز در کروموزوم‌ها رخ می‌دهدکه ممکن است باعث بهتر شدن نسل بعدی شود. الگوریتم ژنتیک می‌تواند نواحی مختلف فضای جواب را همزمان جستجو کند؛ اما هیچ ساختاری برای ایجاد تغییرات کوچک و هدفمند و حرکت دنباله‌ای به سمت جواب‌های بهتر ندارد، در نتیجه الگوریتم ژنتیک توانایی یک جستجوی محلی کامل را ندارد. این ضعف از طریق ترکیب آن با یک الگوریتم جستجوی محلی مانند شبیه‌ساز تبرید قابل حل است.

٤-٢ الگوریتم شبیه‌سازی تبرید

الگوریتم شبیه‌سازی تبرید (SA) یک روش جستجوی احتمالی، دنباله‌ای و بهبود دهنده است که از یک جواب اولیه شروع می‌کند و سپس در یک حلقه تکرار به سوی جواب‌های همسایه حرکت می‌کند. اگر جواب همسایه بهتر از جواب فعلی باشد، الگوریتم، آن را به عنوان جواب فعلی قرار می‌دهد (به‌سمت آن حرکت می‌کند). در غیر این صورت، الگوریتم، آن جواب را با احتمال $e^{\frac{-\Delta E}{T}}$ به عنوان جواب فعلی می‌پذیرد. با کاهش تدریجی دما، در گام‌های پایانی، جواب‌های بدتر با احتمال کم‌تری پذیرفته می‌شوند؛ بنابراین الگوریتم به‌سمت جواب‌های بهتر (یا با همان کیفیت) همگرا می‌شود. آزمایش‌ها نشان می‌دهد که شروع الگوریتم با یک جواب اولیه‌ی خوب منجر به همگرایی سریع‌تر می‌شود. همچنین جستجوی فضای جواب با چند جواب اولیه به‌صورت موازی منجر به یافتن جواب‌هایی با کیفیت بهتر می‌شود. این تکنیک، موازی‌سازی اجراهای چندگانه‌ی مستقل نامیده می‌شود. در روش MIR هر یک از جواب‌های اولیه به صورت کاملا مستقل اقدام به جستجوی فضای جواب می‌کنند.

٤-٣ الگوریتم ترکیبی ژنتیک - شبیه‌سازی تبرید موازی (GA-PSAA)

الگوریتم ژنتیک هیچ ساختاری برای ایجاد تغییرات کوچک و هدفمند و حرکت دنباله‌ای به سمت جواب‌های بهتر ندارد، در نتیجه این الگوریتم توانایی یک جستجوی محلی کامل را ندارد. این ضعف الگوریتم ژنتیک از طریق ترکیب آن با یک الگوریتم جستجوی محلی مانند شبیه‌ساز تبرید قابل حل است. این ترکیب باعث بهبودکارایی هر دو الگوریتم می‌شود. در روش ترکیبی، الگوریتم ژنتیک با استفاده از عملگرهای تقاطع و جهش، مجموعه‌ای از جواب‌ها را تولید می‌کند. تعدادی از این جواب‌ها به‌عنوان جواب‌های اولیه الگوریتم SA انتخاب می‌شود و فرایند جستجوی محلی به طور موازی بر روی جواب‌های انتخابی آغاز می‌شود. برای ایجاد تنوع تعدادی از جواب‌های خوب متوسط و بد انتخاب می‌شود.

[Downloaded from jorlu.ir on 2026-11-29]

	تسهیل شماره ١					تسهیل شماره ٢					تسهیل شماره ٣					تسهیل شماره ٤					تسهیل شماره ٥				
شماره موقعیت	١	٢	٣	٤	٥	١	٢	٣	٤	٥	١	٢	٣	٤	٥	١	٢	٣	٤	٥	١	٢	٣	٤	٥
دوره اول	١	٠	٠	٠	٠	٠	٠	٠	١	٠	٠	٠	٠	٠	١		١	٠	٠	٠	٠	٠	١	٠	٠
دوره دوم	١	٠	٠	٠	٠	٠	٠	٠		١	٠	٠	١	٠	٠		١	٠	٠	٠	٠	٠	٠	١	٠
دوره سوم	٠	٠	٠	٠	١		١	٠	٠	٠	١	٠	٠	٠	٠	٠	٠	٠	١	٠	٠	٠	١	٠	٠

شکل ١. نمایش یک کروموزوم $N^{٢}\times T$ بیتی

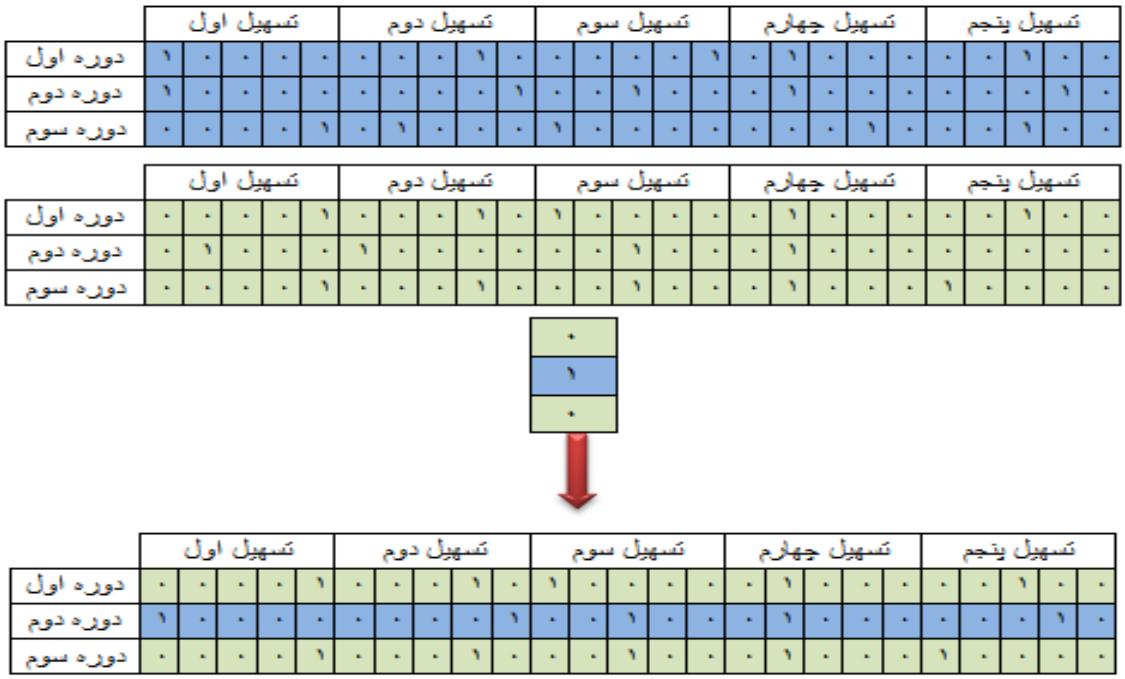
٤-٣-١ ساختار رشته‌ها در الگوریتم GA-PSAA

در کروموزوم طراحی شده، در این نوشتار از کدگذاری دو دویی استفاده شده است. هر سطر نشان‌دهنده یک دوره‌ی زمانی است و هر بیت یکی از دو مقدار صفر و یک را می‌پذیرد که عدد یک، نشان‌دهنده قرار گیری تسهیل مورد نظر در موقعیت بیت مورد نظر است؛ بنابراین برای معرفی یک طرح چیدمان با ٥ تسهیل در ٣ دوره نیازمند یک رشته ٣*٥*٥ بیتی هستیم که در شکل (١) نمایش داده شده است. به عنوان مثال کروموزوم بالا نشان‌دهنده‌ی الگوی چیدمان در سه دوره است. که در دوره‌ی اول، تسهیل شماره یک در موقعیت یک، تسهیل شماره دو در موقعیت چهار، تسهیل شماره سه در موقعیت پنج، تسهیل شماره چهار در موقعیت دو و تسهیل شماره پنج در موقعیت سه قرار گرفته است. مدلی که در بخش قبل ارایه شد تلاش می‌کند هزینه زنجیره تامین را حداقل نماید ولی با این حال پس از تحقق تقاضای بازار و در جهان واقعی، مقدار واقعی هزینه‌ها می‌تواند بسیار متفاوت از ارزش تابع هدف شود. برای کاهش میزان این اختلاف، در این بخش مدل استوار مساله ارایه می‌شود.

٤-٣-٢ عملگر تقاطع

عملگر اصلی جهت تولیدکروموزوم‌های جدید در الگوریتم ژنتیک، عملگر تقاطع می‌باشد. این عملگر مشابه همتای خودش در طبیعت افراد جدیدی تولید می‌نماید، که اجزای (ژن‌های) آن از والدینش تشکیل می‌گردد. این عملگر بر روی یک جفت از کروموزوم‌ها عمل نموده، می‌تواند به صورت تک نقطه‌ای، چند نقطه‌ای یکنواخت باشد. در الگوریتم پیشنهادی، از تقاطع یکنواخت استفاده شده است. یک بردار تصادفی (بردار مرجع) از ارقام صفر و یک به طول تعداد دوره‌های زمانی ایجاد می‌شود و چگونگی ترکیب والدین و ایجاد فرزندان توسط بردار مرجع تعیین می‌شود. هر خانه از بردار مرجع نشان‌دهنده چگونگی الگوپذیری فرزند از والدین است، به این ترتیب که اگر یک باشد، فرزند مانند والد اول (پدر) رفتار می‌کند و اگر صفر باشد، فرزند مانند والد دوم (مادر) رفتار می‌کند (شکل (٢)).

[Downloaded from jorlu.ir on 2026-11-29]



شکل ۲. عملگر تقاطع

٤-۳-۳ عملگر جهش تطبیقی در GA-PSAA

در سیر تکاملی طبیعی، جهش، یک فرایند تصادفی است که در آن محتوای یک ژن با ژن دیگر جهت تولید یک ساختار ژنتیکی جدید جایگزین می گردد. در الگوریتم ژنتیک، جهش به طور تصادفی با احتمال کم (معمولاً بین ۰/۰۰۱ و ۰/۰۱) انجام شده، عناصر را درکروموزوم تغییر می‌دهد. نقش جهش اغلب به عنوان تضمینی است برای آن‌که تنوع کروموزوم‌ها در طی فرایند جستجو حفظ شود. وقتی که جمعیت به سمت جواب خاصی همگرا می‌شود، احتمال جهش باید زیاد شده تا از این عمل جلوگیری نماید و بالعکس وقتی جمعیت دارای جواب‌های غیر یکسان است باید احتمال جهش کم شود؛ بنابراین احتمال جهش، تابع معکوسی از تعداد جواب‌های غیر یکسان جمعیت است؛ بنابراین جهش در جمعیت‌ها باید تا حد ممکن منطقی و هوشمند باشد. در نتیجه ما از یک عملگر جهش تطبیقی استفاده کردیم که در آن جهش فقط زمانی اعمال می‌شود که میزان شباهت کروموزوم‌ها در هر جمعیت به حد خاصی برسد. ضریب شباهت بین دو کروموزوم از طریق رابطه‌ی زیر محاسبه می‌شود:

$$SC_{ab} = \frac{\sum_{i=1}^{N}\sum_{t=1}^{T}\partial\left(X_{ita}, X_{itb}\right)}{N \times T} \qquad (۹)$$

که در آن X_{itb} و X_{ita} موقعیت تسهیل i در دوره t در کروموزوم a و b می‌باشد. میزان بین دو بیت X_{itb} و X_{ita} از رابطه زیر محاسبه می‌شود:

$$\partial\left(X_{ita}, X_{itb}\right) = \begin{cases} ۱ & if\ X_{ita} = X_{itb} \\ ۰ & otherwise \end{cases} \qquad (۱۰)$$

[Downloaded from jorlu.ir on 2026-11-29]

متوسط ضریب شباهت بین کروموزوم‌های یک جمعیت از طریق رابطه زیر تعیین می‌شود:

$$\overline{SC} = \frac{\sum_{a=1}^{N-1}\sum_{b=a+1}^{N} SC_{ab}}{\binom{N}{2}} \qquad (11)$$

در نتیجه در هر نسل، فقط در صورت برقرار بودن شرط $\overline{SC} \geq \gamma$ عملگر جهش می‌تواند بر روی کروموزوم‌ها اعمال شود. γ مقداری بین صفر و یک است که برطبق آزمایش‌های از پیش طراحی شده به‌دست می‌آید. برای اعمال جهش بر روی کروموزوم‌ها (در صورت برقرار بودن شرایط ابتدایی) از گام‌های زیر پیروی می‌کنیم:

١. اولین کروموزوم موجود در جمعیت انتخاب می‌شود.

٢. یک عدد تصادفی حقیقی در فاصله ی صفر و یک تولید می‌نماییم (r).

اگر $r<P_m$ باشد کروموزوم مورد نظر تحت عملگر جهش قرار می‌گیرد . برای اعمال جهش، ابتدا یک دوره‌ی زمانی انتخاب شده و سپس موقعیت تسهیلات در دوره‌ی زمانی مورد نظر به شکل کاملا تصادفی تعیین می‌شود. در شکل (٣)، دوره‌ی زمانی اول به تصادف انتخاب شده و موقعیت تسهیلات در دوره اول به طور تصادفی با هم جابه‌جا شده است که در نتیجه آن طرح چیدمان مربوط به دوره اول کاملاً متحول خواهد شد.

	تسهیل اول					تسهیل دوم					تسهیل سوم					تسهیل چهارم					تسهیل پنجم				
دوره اول	١	٠	٠	٠	٠	٠	٠	٠	١	٠	٠	٠	٠	٠	١	٠	١	٠	٠	٠	٠	٠	١	٠	٠
دوره دوم	١	٠	٠	٠	٠	٠	٠	٠	٠	١	٠	٠	١	٠	٠	٠	١	٠	٠	٠	٠	٠	٠	١	٠
دوره سوم	٠	٠	٠	٠	١	٠	١	٠	٠	٠	١	٠	٠	٠	٠	٠	٠	٠	١	٠	٠	٠	١	٠	٠

	تسهیل اول					تسهیل دوم					تسهیل سوم					تسهیل چهارم					تسهیل پنجم				
دوره اول	٠	٠	٠	٠	١	٠	٠	١	٠	٠	١	٠	٠	٠	٠	٠	١	٠	٠	٠	٠	١	٠	٠	٠
دوره دوم	١	٠	٠	٠	٠	٠	٠	٠	٠	١	٠	٠	١	٠	٠	٠	١	٠	٠	٠	٠	٠	٠	١	٠
دوره سوم	٠	٠	٠	٠	١	٠	١	٠	٠	٠	١	٠	٠	٠	٠	٠	٠	٠	١	٠	٠	٠	١	٠	٠

شکل ٣. چگونگی عملکرد اپراتور جهش

٤-٣-٤ مکانیزم تولید جواب همسایگی

یکی از موارد مهم در الگوریتم شبیه‌ساز تبرید موازی، نحوه‌ی ایجاد جواب همسایگی می‌باشد. این جواب باید دارای دو خصصیه مهم ذیل باشد:

١. به‌صورت تصادفی انتخاب شود.

٢. به نحوی از روی جواب قبلی، به‌دست آید.

[Downloaded from jorlu.ir on 2026-11-29]

بدین منظور ابتدا دو تسهیل و یک دوره‌ی زمانی، انتخاب شده، سپس موقعیت تسهیلات انتخابی در دوره‌ی زمانی مورد نظر با هم تعویض می‌شود. در شکل (۴)، دوره‌ی زمانی اول به تصادف انتخاب شده و موقعیت تسهیلات اول و سوم با هم جابه‌جا شده است.

٤-۳-۵ شرط توقف

شرط توقف در حلقه خارجی (الگوریتم ژنتیک) و حلقه داخلی (الگوریتم شبیه‌ساز تبرید موازی) الگوریتم پیشنهادی به‌قرار زیر است. الگوریتم ژنتیک بدون هیچ محدودیتی به تکامل خود ادامه می‌دهد، تا هنگامی که بهترین کروموزوم به‌دست آمده در ۱۰۰ نسل متوالی یکسان باشد و بهبودی در آن مشاهده نشود. همچنین در هر تکرار، الگوریتم شبیه‌ساز تبرید موازی، تا هنگامی که احتمال پذیرش جواب‌های غیر بهبود دهنده برابر صفر (صفر حدی) بشود تکرار می‌شود، از آن پس هنگامی که کیفیت جواب‌های به‌دست آمده در ۱۰۰ تکرار متوالی تغییر نکند، الگوریتم متوقف می‌شود.



شکل ٤. چگونگی ایجاد ساختار همسایگی در الگوریتم شبیه‌ساز تبرید موازی

۵ تنظیم پارامتر و نتایج محاسباتی

۵-۱ تنظیم پارامترها

یکی از روش‌های بسیار موثر در طراحی آزمایش، روش تاگوچی است. در این روش از نسبت هشدار به اغتشاش برای تعیین بهترین ترکیب آزمایش استفاده می‌شود. با توجه به هدف آزمایش‌ها، از معادله نسبت هشدار به اغتشاش (S/N) هرچه کم‌تر- بهتر، استفاده شد:

$$\left(\frac{S}{N}\right) = -۱۰\log\left(\frac{۱}{n}\right)\sum_{i}\left(y_i^{۲}\right) \qquad (۱۲)$$

که n و y_i به ترتیب معرف تعداد آزمایش‌ها و مقدار پاسخ مورد نظر فرآیند در آزمایش iام می‌باشد.

[Downloaded from jorlu.ir on 2026-11-29]

در تحقیق حاضر نیز برای یافتن شرایط بهینه‌ی الگوریتم GA_PSAA، شش عامل نرخ تقاطع، نرخ جهش، دمای اولیه، نرخ کاهش دما و تعداد جواب‌های اولیه برای شروع کار الگوریتم SA و نرخ جهش تطبیقی γ، مورد بررسی قرار می‌گیرد. همچنین تعداد جمعیت اولیه و حداکثر تکرار نیز به‌عنوان عامل هفتم و به‌صورت زوج مرتب در نظر گرفته شده است. عوامل مذکور در چهار سطح مورد بررسی قرار گرفتند. هر آزمایش به تعداد ده بار تکرار شد و سطوح بهینه‌ی آن‌ها در جدول (۲) آمده است.

جدول ۲. سطوح بهینه پارامترهای الگوریتم

عوامل	سطح بهینه
نرخ تقاطع	۰/۸
نرخ جهش	۰/۱۵
لاندا (γ)	۰/۹۵
دمای اولیه	۱۰۰۰
نرخ کاهش دما	۰/۹۸۵
تعداد جواب‌های اولیهSA	۵
حداکثر تکرار-جمعیت اولیه	(۵۰۰،۵۰)

۵-۲ ارزیابی عملکرد الگوریتم ترکیبی ژنتیک – شبیه‌سازی تبرید موازی برای حل مساله DFLPBC

تمامی محاسبات توسط یک PC با چهار گیگابایت RAM و پردازشگر Core i5, 2 duo (2.2 GHz) انجام شده و برای تنظیم پارامترهای الگوریتم نیز از نرم افزار Minitab16 استفاده شده است. برای ارزیابی الگوریتم پیشنهادی، مسایل به‌دست آمده از چنگ و بالاکریشنان [۱۰] که شامل داده‌هایی برای مسایلی با ۶، ۱۵ و ۳۰ تسهیل در افق زمانی ۵ و ۱۰ دوره‌ای است، مورد بررسی قرار داده، نتایج به‌دست آمده را با ساهین و همکاران [۲۰] و نیز بایکاسگلو [۱۸] مقایسه می‌کنیم. چگونگی تخصیص بودجه به هر دوره مانند ساهین و همکاران [۲۰] است. در این تحقیق، برای ارزیابی الگوریتم‌ها از معیار درصد انحراف نسبی (RPD) استفاده می‌شود. RPD بر طبق فرمول زیر محاسبه می‌شود:

$$RPD_{ij} = \frac{Alg_{sol}(ij) - min_{sol}(j)}{min_{sol}(j)} \qquad (۱۳)$$

در معادله بالا i شماره الگوریتم و j شماره مساله و $min_{sol}(j)$ بهترین جواب به‌دست آمده در مساله j است.

جدول ۳ متوسط درصد انحراف نسبی را در هر سایز از مساله نمایش می‌دهد. مقدار RPD نشان می دهد که جواب‌ها در هر الگوریتم تا چه اندازه از بهترین جواب به دست آمده فاصله دارند. با توجه به زیاد بودن این فاصله در الگوریتم ارایه شده توسط بایکاسگلو می‌توان گفت که الگوریتم بایکاسگلو جواب‌های بسیار ضعیفی

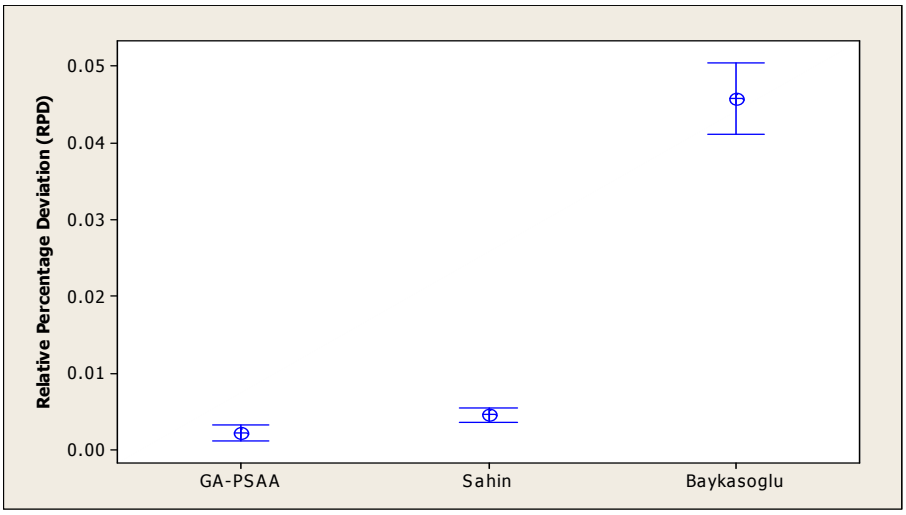
[Downloaded from jorlu.ir on 2026-11-29]

تولید می‌کند و نیاز به بررسی عمیق‌تر این الگوریتم وجود ندارد. در حالی که الگوریتم‌های ساهین و GA-PSAA، جواب‌های بسیار بهتری تولید می‌کنند و بنابراین الگوریتم‌های کاراتری هستند.

جدول ۳. متوسط درصد انحراف نسبی به ازای تعداد تسهیلات و دوره‌های زمانی

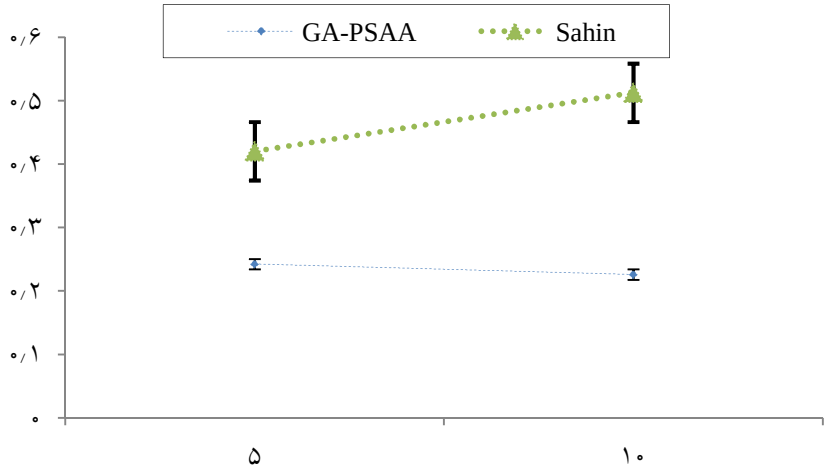
T	N	الگوریتم ها		
		GA-PSAA	ساهین	بایکاسگلو
۵	۶	۰/۵۲۴۵	۱/۳۰۵۱	۰/۱۶۶۱
	۱۵	۰/۱۴۰۷	۶/۳۱۷۶	۰/۳۹۱۶
	۳۰	۰/۰۶۲	۵/۶۲۹۵	۰/۷۰۳۴
۱۰	۶	۰/۴۷۴۷	۱/۸۱۸۵	۰/۵۲۲۷
	۱۵	۰/۰۳۳۷	۶/۹۹۹۷	۰/۲۶۴
	۳۰	۰/۱۶۹۴	۵/۳۸۳۴	۰/۷۵۰۸
	Average	۰/۲۳۴۱۶۷	۴/۵۷۵۶۳۳	۰/۴۶۶۴۳۳

همچنین براساس فاصله‌ی اطمینان ۹۵ درصدی برحسب معیار درصد انحراف نسبی که در شکل (۵) نمایش داده شده است، الگوریتم ارایه شده توسط بایکاسگلو هم از نظر معیار میانگین و هم از نظر معیار انحراف استاندارد، در شرایط بسیار بدتری نسبت به الگوریتم‌های ساهین و GA-PSAA قرار دارد که این خود دلیلی بر لزوم حذف این الگوریتم در مقایسات آتی است. برای ارزیابی الگوریتم‌های GA-PSAA و ساهین در دوره‌های زمانی مختلف، میانگین و فاصله اطمینان درصد انحراف نسبی آن‌ها بر حسب تعداد دوره‌ی زمانی و تعداد تسهیلات محاسبه شده است (شکل‌های (۶ و ۷)).



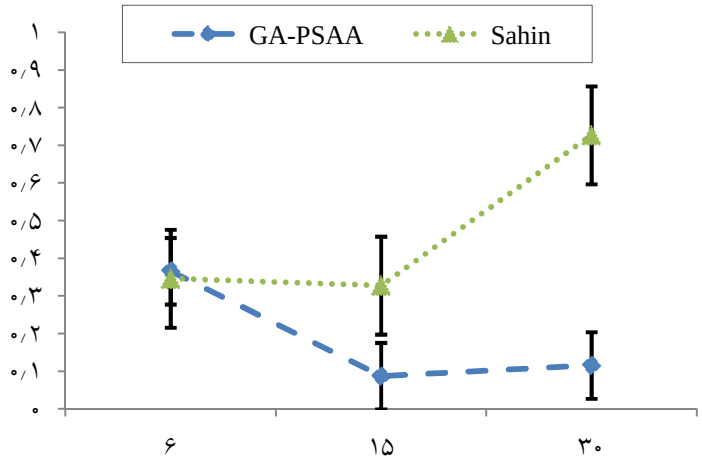
شکل ۵. فاصله اطمینان ۹۵ درصدی برای درصد انحراف نسبی

[Downloaded from jorlu.ir on 2026-11-29]



شکل ٦. رفتار الگوریتم‌ها بر حسب شاخص RPD در تعداد دوره‌های زمانی متفاوت

همان‌طور که در شکل (۶) نشان داده شده است الگوریتم GA-PSAA عملکرد پایدارتری را نسبت به ساهین داشته است. به‌طوری که با افزایش تعداد دوره‌ها، کارایی الگوریتم پیشنهادی بیش‌تر آشکار می‌شود و این نشان می‌دهد که الگوریتم پیشنهادی عملکرد قابل قبول‌تری در مسایل با ابعاد وسیع دارد.



شکل ۷. رفتار الگوریتم‌ها بر حسب شاخص RPD در تعداد تسهیلات متفاوت

نتایج به‌دست آمده از شکل (۷) نشان می‌دهد که عملکرد الگوریتم‌ها هنگامی که تعداد تسهیلات برابر ۶ می‌باشد، تفاوت معناداری ندارد، اما با افزایش تعداد تسهیلات به ویژه برای N=۳۰ کارایی الگوریتم پیشنهادی بهتر خواهد شد؛ بنابراین می‌توان امیدوار بود که برای مسایل واقعی که ممکن است بسیار بزرگ‌تر باشد، الگوریتم GA-PSAA کاراتر خواهد بود. برای مقایسه‌ی آماری نتایج به‌دست آمده از الگوریتم‌ها، از آزمون آماری T استفاده

[Downloaded from jorlu.ir on 2026-11-29]

می‌شود. فرض صفر در آزمون T بیان می‌کند که میانگین درصد انحراف نسبی دو الگوریتم هیچ تفاوت معنی داری از نظر آماری ندارد.

$H_0 : \mu_1 = \mu_2$

$H_1 : \mu_1 < \mu_2$

μ_1 و μ_2 به ترتیب میانگین درصد انحراف نسبی الگوریتم GA-PSAA و ساهین است. نتایج به‌دست آمده از آزمون آماری Tدر جدول (۳) آمده است.

آزمون آماری T بر اساس ۹۵ درصد بازه اطمینان، نشان‌دهنده اختلاف معنی‌دار، بین میانگین‌های دو جامعه است. به‌عبارت دیگر بر طبق آماره‌ی T و P-value به این نتیجه می‌رسیم که میانگین درصد انحراف نسبی الگوریتم‌ها با هم برابر نیست و با توجه به رد شدن فرض صفر می‌توان بیان کرد که میانگین درصد انحراف نسبی در الگوریتم GA-PSAA کم‌تر است که این خود دلیلی بر بهتر بودن الگوریتم پیشنهادی از لحاظ کیفیت جواب است.

نتیجه‌گیری و پیشنهاد تحقیقات آتی

مساله چیدمان پویای تجهیزات با در نظر گرفتن محدودیت بودجه، یکی از مسایل پرکاربرد در دنیای واقعی است که تاکنون تحقیقات کمی روی آن صورت گرفته است. در این تحقیق، یک الگوریتم فراابتکاری موثر برای مساله چیدمان پویای تجهیزات با محدودیت بودجه، توسعه و مورد آزمون قرار گرفت. برای مقایسه‌ی الگوریتم پیشنهادی با سایرالگوریتم‌های موجود، ابتدا یک مرحله غربالگری انجام شد تا الگوریتم‌هایی انتخاب شوند که توانایی رقابت با هم را دارند، سپس توسط آزمون‌های آماری و معیار درصد انحراف نسبی یک مقایسه‌ی جامع بر روی الگوریتم‌های انتخاب شده انجام شد. نتایج به‌دست آمده حاکی از آن بود که الگوریتم پیشنهادی با بهره گرفتن از مزایای خاص الگوریتم ژنتیک و شبیه‌سازی تبرید توانسته است کیفیت جواب‌هایی را که تاکنون توسط محققین قبلی ارایه شده است بهبود بخشد. نکته‌ی آخر این که این تحقیق مستقیماً سایت‌های تولیدی و به عبارتی چیدمان ماشین‌آلات تولیدی در کارخانجات صنعتی را مورد توجه قرار می‌دهد؛ اما به علت کلیت تحقیق می‌توان از نتایج آن برای سایر چیدمان‌ها مخصوصا چیدمان‌های دفتری بهره برد. جهت تحقیقات آتی ملاحظات ذیل پیشنهاد می‌شود:

- ✓ در نظر گرفتن وسایل حمل و نقل و هزینه‌های ثابت و متغیر آن‌ها.
- ✓ در نظر گرفتن ارزش زمانی پول در دوره مختلف و همچنین مساله فازی بودن هزینه‌ها.
- ✓ عملکرد الگوریتم پیشنهادی می‌تواند بر روی مسایلی که از نظر محاسباتی پیچیده هستند مورد بررسی قرار گیرد.
- ✓ حل مساله DFLPBC در حالت چند هدفه (اهداف مورد نظر در مسایل جایابی در فصل اول بیان شده است).
- ✓ متغییر و نابرابر در نظر گرفتن اندازه تسهیلات و حل مساله با استفاده از الگوریتم پیشنهادی.

[Downloaded from jorlu.ir on 2026-11-29]

منابع

[۵] وینچه عبداله، ه.، قاسمی، ا.م.، (۱۳۸۸). یک مدل بهینه‌سازی وزین غیر خطی جهت رتبه‌بندی مدل‌های مختلف چیدمان در مساله طراحی چیدمان تسهیلات. مجله تحقیق در عملیات در کاربردهای آن، ۶ (۲۳)، ۶۰-۵۱.

[1] Tompkins, J. A., White, J. A., Bozer, Y. A., Frazelle, E. H., Tanchoco, J. M. A., Trevino, J., (1996) Facilities Planning, NY: John Wiley & Sons, Inc.

[2] Francis, R. L., McGinnis Jr., L. F., White, J. A., (1992). Facility Layout and Location: An Analytical Approach. Englewood Cliffs, NJ: Prentice-Hall.

[3] Ben jaafar, S., Sheikhzadeh, M., (2000). Design of flexible plant layouts. IIE Transactions, 32, 309–322.

[4] Rosenblatt, M. J., (1986). The dynamics of plant layout. Management Science, 32(1), 76–86.

[6] Balakrishnan, J., Cheng, C. H., (2000). An improved pairwise exchange for the dynamic plant layout problem. International Journal of Production Research, Vol. 38, No. 13, pp. 3067-3077.

[7] Ballou, R. H., (1967). Improving the physical layout of merchandise in warehouses. Journal of Marketing 31(3), 60-64.

[8] Lacksonen, T. A., Enscore, E. E., (1993). Quadratic assignment algorithms for the dynamic layout problem. International Journal of Production Research, 31(3), 503–17.

[9] Urban, T. L., (1993). A heuristic for the dynamic facility layout problem. IIE Transactions, 25(4), 57–63.

[10] Balakrishnan, J., Cheng, C. H., (2000). Genetic search and the dynamic layout problem. Computers and Operations Research, 27(6), 587–593.

[11] Conway, D. G., Venkataramanan, M. A., (1994). Genetic search and the dynamic facility layout problem. Computers and Operations Research, 21(8), 955– 960.

[12] Chang, M., Sugiyama, M., Ohkura, K., Ueda, K., (2002) .A coevolutionary genetic algorithm approach to the dynamic facility layout problem,' Proceedings of 2002 Japan-USA Symposium on Flexible Automation (2002 JUSFA), 93-96.

[13] Dunker, T., Radons, G., Westkamper, E., (2005). Combining evolutionary computation and dynamic programming for solving a dynamic facility layout problem. European Journal of Operational Research, 165(1), 55–69.

[14] Kaku, B. K., Mazzola, J. B., (1997). A tabu search heuristic for the dynamic plant layout problem. INFORMS Journal on Computing, 9(4), 374–84.

[15] Baykasoglu, A., Gindy, N. N. Z., (2001). A simulated annealing algorithm for dynamic facility layout problem. Computers and Operations Research, 28(14), 1403–1426.

[16] McKendall, A. R., Shang, J., Kuppusamy, S., (2006). Simulated annealing heuristics for the dynamic facility layout problem. Computers and Operations Research, 33 (8), 2431–2444.

[17] Erel, J.B., Ghosh, J., Simon, J. T., (2003). New heuristic for the dynamic layout problem. Journal of the Operational Research Society, 54, 1275–82.

[18] Baykasoglu, A., Dereli, T., Sabuncu, I., (2006). An ant colony algorithm for solving budget constrained and unconstrained dynamic facility layout problems. Omega, 34(4), 385–396.

[19] Balakrishnan, J., Jacobs, F. R., Venkataramanan, M. A., (1992). Solutions for the constrained dynamic facility layout problem. European Journal of Operational Research, 57(2), 280–286.

[20] Sahin, R., Ertogral, K., Turkbey, O., (2010). A simulated annealing heuristic for the dynamic facility layout problem with budget constraint. Computers & Industrial Engineering, 59, 308-313.

[21] Garey, M. R., Johnson, D. S., (1979). Computers and Intractability: A Guide to the Theory of NP-completeness, Freeman & Co., San Francisco.

[Downloaded from jorlu.ir on 2026-11-29]