

مدلسازی زمان‌بندی و اندازه انباشته اقتصادی در جریان کارگاهی جایگشتی توزیع‌شده با کارخانه‌های متفاوت

محمد علاقه‌بندها^۱، بهمن نادری^{۲*}، محمد محمدی^۳

۱- دانشجوی دکتری، گروه مهندسی صنایع، دانشکده فنی و مهندسی، دانشگاه خوارزمی، تهران، ایران

۲- دانشیار، گروه مهندسی صنایع، دانشکده فنی و مهندسی، دانشگاه خوارزمی، تهران، ایران

۳- دانشیار، گروه مهندسی صنایع، دانشکده فنی و مهندسی، دانشگاه خوارزمی، تهران، ایران

پذیرش: ۱۳۹۶/۱۱/۳۰

دریافت: ۱۳۹۶/۹/۹

چکیده

این مقاله مدل جدید زمان‌بندی و اندازه انباشته اقتصادی در جریان کارگاهی جایگشتی توزیع شده با چندین کارخانه و ماشین متفاوت را نشان می‌دهد که تاکنون این موضوع مدنظر قرار نگرفته است. بدین منظور، محصولات متفاوت باید در بین کارخانه‌ها توزیع شود و سپس توالی محصولات تخصیصی به هر کارخانه نیز مشخص شود. هدف مسئله، حداقل کردن مجموع هزینه‌های راه‌اندازی، موجودی در جریان و موجودی محصول نهایی در واحد زمان و هزینه کل تولید شامل هزینه تخصیص نامتوازن محصولات بین کارخانه‌ها است. از آنجایی که مدل مذکور NP-hard است، از الگوریتم ترکیبی بهینه‌سازی میرایی ارتعاش^۱ (VDO) با الگوریتم‌های رقابت امپریالیستی^۲ (ICA) و شبیه‌سازی تبرید^۳ (SA) با نام^۴ (HVDO) و همچنین از الگوریتم ژنتیک^۵ (GA) و VDO برای مقایسه استفاده شده است. برای تعیین بهترین مقادیر پارامترهای هر یک از الگوریتم‌ها که منجر به بهترین جواب می‌شود، روش رویه پاسخ^۶ (RSM) به‌کاربرده شده است.

به‌منظور به دست آوردن جواب بهینه و مقایسه آن با مدل غیرخطی، از روش خطی‌سازی استفاده کرده و سپس مدل خطی با نرم‌افزار لینگو حل شده است. نتایج نشان می‌دهد که HVDO عملکرد بهتری در به دست آوردن حداقل تابع هدف در این مسئله دارد.

واژگان کلیدی: جریان کارگاهی جایگشتی توزیع شده؛ خطی‌سازی؛ الگوریتم بهینه‌سازی میرایی ارتعاش؛ الگوریتم رقابت امپریالیستی؛ روش رویه پاسخ.

۱- مقدمه

زمان‌بندی و برنامه‌ریزی، تخصیص منابع برای انجام مجموعه‌ای از وظایف در طی یک دوره زمانی است. در بسیاری از کارخانه‌هایی که دارای ماشین‌آلات و تجهیزات گسسته هستند، هر کار/محصول دارای چندین عملیات است. مسئله زمان‌بندی جریان کارگاهی^۱ یک طبقه معروف از دسته مسائل زمان‌بندی با کاربردهای زیاد در سیستم‌های صنعتی و تولیدی است که در سال‌های اخیر، بسیار مورد مطالعه قرار گرفته است. در یک مسئله FSSP هدف اصلی، تعیین توالی n کار مختلف $\{J_1, J_2, \dots, J_n\}$ است که باید توسط m ماشین $\{M_1, M_2, \dots, M_m\}$ پردازش شود؛ به طوری که تابع هدف از پیش تعیین‌شده مسئله بهینه شود. در مسائل جریان کارگاهی، مجموعه‌ای از پردازش‌ها باید به صورت متوالی با همان مسیر یکسان برای تکمیل همه کارها انجام شود. اگرچه تابع هدفی که به‌طور معمول در این مسائل استفاده می‌شود حداقل‌سازی حداکثر زمان تکمیل یا makespan است، ولی اهداف دیگری مانند زمان تکمیل، زمان جریان و تأخیر نیز می‌تواند در نظر گرفته شود. الفت [۱] در مقاله خود به حل مسئله زمان‌بندی جریان کارگاهی با استفاده از الگوریتم ژنتیک، شبیه‌سازی تبرید و ازدحام ذرات پرداخته است. در این مدل که حالت خاصی از مسئله عمومی زمان‌بندی جریان کارگاهی است، هدف حداقل‌کردن مجموع دیرکردهای مرحله‌ای، موردبررسی و حل قرار گرفته است. زمان‌بندی جریان کارگاهی جایگشتی^۲ به‌عنوان یک مسئله بهینه‌سازی ترکیبی به‌طور گسترده‌ای موردبررسی قرار گرفته و نقش بسیار مهمی را در سیستم‌های تولیدی و فرآیندهای صنعتی ایفا می‌کند. در این مسئله، همه کارها باید در یک کارخانه

1. FSSP: flow shop scheduling problem

2. PFSP: permutation flow-shop scheduling problem

پردازش شوند. وجود یک مرکز تولیدی یا یک کارخانه، یکی از فرض‌های رایج در این مسئله است و تمام کارها به‌طور جایگشتی به این کارخانه تخصیص می‌یابند. همان‌طور که در محیط جریان کارگاهی جایگشتی تعریف می‌شود، جایگشت و تعویض کار روی هر ماشین به‌طور مشابه تکرار می‌شود؛ یعنی از اصل FIFO پیروی می‌کند و هر فعالیت زمانی انجام می‌شود که فعالیت قبل از آن انجام شده باشد. به‌عبارت‌دیگر، مسئله زمان‌بندی جریان کارگاهی دارای n کار روی m ماشین است. هر کار J_j دارای یک سری عملیات $\{O_{1j}, O_{2j}, \dots, O_{mj}\}$ است و باید بدون انقطاع روی هر ماشین با زمان‌بندی جایگشتی پردازش شود. زمان پردازش نرمال عملیات O با P نشان داده می‌شود. مسئله زمان‌بندی جریان کارگاهی جایگشتی معمولاً به‌صورت $n/m/P/C_{max}$ نشان داده می‌شود که در آن، n نشان‌دهنده تعداد کارها، m تعداد ماشین‌ها، P زمان پردازش و C_{max} نشان‌دهنده تابع هدف است.

امروزه، شرکت‌های تک کارخانه‌ای کمتر رایج هستند و شرکت‌هایی با چند کارخانه و زنجیره عرضه آن‌ها نقش مهمی در عمل بازی می‌کنند. با توسعه فرهنگ همکاری در تولید محصولات مختلف در چند کارخانه، مسئله زمان‌بندی توزیع شده برای استفاده بهتر از منابع، افزایش بهره‌وری و سود، کاهش هزینه‌های تولید، کاهش ریسک مدیریت و افزایش کیفیت محصول بسیار مفید بوده است. بر همین اساس و در محیط توزیع شده، نادری و روئیز [۲] مسئله PFSP را به‌صورت مسئله جریان کارگاهی جایگشتی توزیع شده^۱ توسعه داده‌اند که از ساختار شرکت‌های امروزی الهام گرفته و در آن شرکت‌ها برای ساخت یک شرکت مشترک با هم ادغام می‌شوند. برنامه‌ریزی و زمان‌بندی در سیستم‌های توزیع شده خیلی پیچیده‌تر از مسائل ساده زمان‌بندی کار است. به‌طور خاص، تخصیص کارها به کارخانه‌ها و زمان‌بندی کارها در هر کارخانه باید هم‌زمان در تصمیم‌گیری در نظر گرفته شود. مسئله جریان کارگاهی جایگشتی توزیع شده به‌طور خلاصه به‌صورت زیر تعریف می‌شود: g کارخانه مشابه وجود دارد. هرکدام از آن‌ها دارای m ماشین هستند که به‌صورت جریان کارگاهی قرار دارند. n کار وجود دارد که باید پردازش شود و هر یک از کارها می‌تواند به هر یک از کارخانه‌ها تخصیص یابد. همه کارخانه‌ها می‌توانند همه فرآیندهای تمامی کارها را انجام دهند. زمانی که تخصیص کار به

1. DPFSP: distributed permutation flow-shop scheduling problem

کارخانه انجام شود، نمی‌توان یک کار را از یک کارخانه به کارخانه دیگر انتقال داد. هر کار باید به‌طور سری از ماشین ۱ تا ماشین m پردازش شود و این توالی برای همه کارها یکسان است. هر کار می‌تواند به یک ماشین تخصیص یابد و هر ماشین فقط یک کار را می‌تواند پردازش نماید. همچنین فرض می‌شود که زمان پردازش همه کارخانه‌ها یکسان است. هدف این مسئله تخصیص کارها به کارخانه‌ها و تعیین توالی بهینه کارها در هر کارخانه است؛ به‌طوری‌که تابع هدف مسئله بهینه شود. مسائل اندازه انباشته اقتصادی^۱ یکی از مسائل معروف و شناخته‌شده تصمیم‌گیری برنامه‌ریزی تولید میان‌مدت است. در این نوع مسئله، مجموعه‌ای از n محصول روی یک ماشین پردازش می‌شود. هر ماشین حداکثر می‌تواند یک محصول را در واحد زمان پردازش کند. تقاضا و نرخ تولید هر محصول نیز ثابت و مشخص است. همه تقاضاها باید برآورده شوند و بنابراین کمبود مجاز نیست. بعضی از راه‌اندازی‌ها ممکن است قبل از شروع تولید انجام شود. این راه‌اندازی‌ها می‌تواند روی هر دو موضوع زمان و هزینه تأثیر داشته باشد. افق زمانی فرض می‌شود که پیوسته و نامحدود است. هدف مسئله مشخص کردن برنامه تولیدی است که بتواند کل هزینه‌های طولانی‌مدت مانند هزینه‌های راه‌اندازی و نگهداری موجودی را به حداقل برساند. زمان‌بندی دوره‌ای اغلب به‌عنوان چرخه مشترک تعریف می‌شود و فرض شده است که زمان‌های چرخه تولیدی همه محصولات یکسان است. برای مقابله با شرایط واقعی موجود در صنایع، مسئله ELSP به‌صورت مسائل اندازه انباشته اقتصادی در جریان کارگاهی جایگشتی توزیع‌شده توسعه داده شده است. در نتیجه، بر اساس نوع مسئله و تحقیقات انجام‌شده در مسئله زمان‌بندی و اندازه انباشته اقتصادی در جریان کارگاهی جایگشتی توزیع‌شده با کارخانه‌های متفاوت، سه تصمیم باید گرفته شود؛ تخصیص محصولات به کارخانه‌ها، زمان‌بندی تولید در هر کارخانه و همچنین تصمیم‌گیری در مورد اندازه انباشته.

۲- پیشینه پژوهش

پژوهش‌های انجام‌شده در زمینه موردتحقیق، در این بخش ارائه شده است. مسئله

1. ELSP: Economic Lot Sizing Problems

اندازه انباشته اقتصادی یکی از مسائل معروف در برنامه‌ریزی تولید است که از سال ۱۹۵۸ مورد توجه قرار گرفته است. قبل از سال ۱۹۹۹، تحقیقات بر این ایده تمرکز داشت که به دلیل تأثیر کمی که توالی تولید بر هزینه کل دارد، این تأثیر قابل چشم‌پوشی است و بنابراین، محققین بیشتر روی تصمیم‌گیری اندازه انباشته تمرکز کرده‌اند و به مسئله توالی نپرداخته‌اند. اونیچه و همکاران [۳] مسئله اندازه انباشته اقتصادی در جریان کارگاهی جایگشتی را مورد بررسی قرار داده و اثر توالی بر هزینه کل نهایی را بررسی کرده‌اند. آن‌ها یک مدل برنامه‌ریزی غیرخطی عدد صحیح مختلط را به همراه روش‌های فرا ابتکاری برای حل آن ارائه کرده‌اند. هلاح [۴] در مقاله خود مسئله زمان‌بندی جریان کارگاهی جایگشتی m مرحله‌ای با در نظر گرفتن زمان بیکاری و تابع هدف تأخیر و تعجیل را مورد بررسی قرار داده است. روش حل دقیق، جستجوی همسایگی و همچنین رویکرد به دست آوردن حد بالا برای این مسئله در نظر گرفته شده است. در مقاله هی [۵] مسئله زمان‌بندی جریان کارگاهی جایگشتی با اثر یادگیری و تابع هدف کمینه‌سازی حداکثر تأخیر مورد مطالعه قرار گرفته است؛ از الگوریتم زودترین زمان تحویل کار و روش شاخه و حد برای حل این مسئله استفاده شده است. در تحقیق نادری و روئیز [۲]، مسئله زمان‌بندی جریان کارگاهی جایگشتی توزیع شده با تابع هدف کمینه‌سازی حداکثر زمان تکمیل ارائه شده است. در این تحقیق، شش مدل مختلف برنامه‌ریزی خطی عدد صحیح مختلط و روش‌های ابتکاری برای حل آن‌ها پیشنهاد شده است. لی و یین [۶] در مقاله خود از الگوریتم گسسته کلونی زنبور عسل برای حل مسئله زمان‌بندی جریان کارگاهی جایگشتی استفاده کرده‌اند. هدف این مسئله حداقل کردن کل زمان در جریان و حداکثر زمان تأخیر کارها است. بهنامیان و قمی [۷] مسئله زمان‌بندی تولید با کارخانه‌های متفاوت و ماشین‌های موازی با سرعت‌های مختلف را که در مناطق مختلف جغرافیایی پراکنده هستند، مورد بررسی قرار داده‌اند که هدف مسئله حداقل‌سازی حداکثر زمان تکمیل کارخانه‌ها است. الگوریتم‌های حل متفاوتی برای مسائل زمان‌بندی استفاده شده است [۸، ۱]. الگوریتم توزیع شده برای حل مسئله زمان‌بندی جریان کارگاهی جایگشتی توزیع شده که وانگ و همکاران [۹] پیشنهاد داده‌اند، استفاده شده است. همچنین برای تنظیم پارامتر این الگوریتم پیشنهادی نیز از روش

تاگوچی استفاده شده است. در مقاله نادری و روئیز [۱۰] الگوریتم جستجوی پراکنده برای حل مسئله زمان‌بندی جریان کارگاهی جایگشتی توزیع شده مورد استفاده قرار گرفته است. این الگوریتم به ندرت برای حل مسئله جریان کارگاهی بکار برده می‌شود. در این الگوریتم، از روش‌های مجموعه مرجع، راه‌حل‌های کامل و جزئی، راه‌اندازی مجدد و جستجوی محلی استفاده شده است. لین و ژانگ [۱۱] از الگوریتم بهینه‌سازی مبتنی بر زیست‌شناسی برای حل مسئله زمان‌بندی جریان کارگاهی جایگشتی توزیع شده با تابع هدف حداقل‌سازی makespan استفاده کرده‌اند. ترکیبی از استراتژی جستجوی محلی و تقاطع ابتکاری در این الگوریتم برای به دست آوردن جواب نزدیک به بهینه بکار برده شده است. الگوریتم ممتیک رقابتی برای حل مسئله چندهدفه زمان‌بندی جریان کارگاهی جایگشتی توزیع شده با توابع هدف makespan و کل زمان تأخیر در مقاله دنگ و ونگ [۱۲] پیشنهاد شده است. آن‌ها از یک الگوریتم جستجوی محلی مبتنی بر دانش برای بهبود قدرت الگوریتم استفاده کرده‌اند. برای تنظیم پارامترهای الگوریتم از روش تاگوچی استفاده شده است. با توجه به اینکه روش‌های حل مختلفی از نوع دقیق و فراابتکاری برای حل مسائل زمان‌بندی استفاده شده است، با این حال تعیین بهترین روش حل با توجه به نوع مسئله و محدودیت‌ها دشوار است.

بر اساس بهترین تحقیقی که انجام شده است، این پژوهش اولین مقاله‌ای است که به زمان‌بندی و اندازه انباشته اقتصادی در جریان کارگاهی جایگشتی توزیع شده با کارخانه‌های متفاوت پرداخته است. در این مقاله، علاوه بر ارائه روش خطی‌سازی مدل که پیش‌ازاین در مقاله دیگری با این نوع مسئله ارائه نشده است، برای اولین مرتبه از الگوریتم ترکیبی بهینه‌سازی میرایی ارتعاش با الگوریتم‌های رقابت امپریالیستی و شبیه‌سازی تبرید (HVDO) و همچنین الگوریتم ژنتیک و الگوریتم بهینه‌سازی میرایی ارتعاش استفاده شده است. عملکرد هر یک از این الگوریتم‌ها با استفاده از حل بهینه‌ای که به وسیله مدل خطی شده در سه اندازه کوچک، متوسط و بزرگ به دست می‌آید، مقایسه می‌شوند. این تحقیق از لحاظ هدف تحقیق، کاربردی و روش تحقیق این مطالعه از نوع روش‌های تحقیق تحلیلی ریاضی است.

مقاله به این صورت ادامه می‌یابد: بخش ۳ به معرفی مدل ریاضی و روش خطی‌سازی آن می‌پردازد. بخش ۴ به تولید مسئله اشاره دارد. بخش ۵ الگوریتم‌های فراابتکاری را معرفی، تنظیم و ارزیابی می‌کند. و در نهایت، بخش ۶ به نتیجه‌گیری و تحقیقات آتی خواهد پرداخت.

۳- مدل‌های ریاضی

مسئله جریان کارگاهی جایگشتی توزیع شده با کارخانه‌های غیر یکسان می‌تواند به این صورت تعریف شود: n محصول که در واقع تقاضای مختلف مشتریان با مقادیر متفاوت است باید در g کارخانه غیر یکسان پردازش شود که هر کارخانه دارای جریان کارگاهی با m ماشین بوده و هر ماشین دارای سرعت μ است. هر تقاضا متشکل از m فعالیت است که باید یکی پس از دیگری انجام شوند. هنگامی که یک محصول به کارخانه‌ای تخصیص می‌یابد نمی‌تواند به کارخانه دیگری انتقال یابد و تمامی فعالیت‌های مربوط به آن باید در آن کارخانه انجام شود. همچنین فرضیات مربوط به مسئله کلاسیک زمان‌بندی جریان کارگاهی در اینجا نیز بکار گرفته می‌شود. تمامی تقاضاها مستقل بوده و در زمان صفر وجود دارند. هر ماشین در یک زمان فقط می‌تواند یک محصول را پردازش کند و هر محصول در یک زمان فقط می‌تواند روی یک ماشین پردازش شود. خرابی و تعمیرات وجود ندارد و ماشین‌ها همیشه در دسترس هستند. تقاضای مشتریان همیشه برای محصول نهایی است که به‌طور پیوسته تحویل می‌شود. تقاضا، نرخ تولید، هزینه و زمان تنظیم و هزینه نگهداری موجودی در یک افق زمانی برنامه‌ریزی نامحدود، مشخص و ثابت هستند. هر محصول روی ماشین‌های مختلف دارای نرخ تولید منحصربه‌فرد است. سطح موجودی و زمان نگهداری، هزینه نگهداری موجودی را به‌طور مستقیم تعیین می‌کنند. کمبود مجاز نیست. بر اساس [۳] اندازه انباشته به‌صورت ضرب مقادیر نرخ تقاضا در زمان چرخه است. هنگامی که تولید یک محصول آغاز می‌شود، نمی‌تواند قطع شود. بافر نامحدود بین ماشین‌ها وجود دارد؛ یعنی محصولات به‌مدت نامحدودی می‌توانند برای ماشین بعدی منتظر بمانند. در این بخش مسئله موردنظر مدل‌سازی می‌شود. قبل از ارائه مدل، پارامترها و زیروندها در جدول ۱ تعریف شده است:

جدول ۱ پارامترها و زیروندهای استفاده‌شده در مدل

علامت	توضیحات
n	تعداد محصولات
m	تعداد ماشین‌ها
g	تعداد کارخانه‌ها
$j, k \in \{1, 2, \dots, n\}$	زیروند محصولات
$i \in \{1, 2, \dots, m\}$	زیروند ماشین‌ها
$f \in \{1, 2, \dots, g\}$	زیروند کارخانه‌ها
$p_{z,i}$	نرخ تولید محصول زروی ماشین i
d_j	نرخ تقاضای محصول نهایی j
$st_{j,i,f}$	زمان راه‌اندازی محصول زروی ماشین i در کارخانه f
sc_j	کل هزینه راه‌اندازی محصول زروی همه ماشین‌ها
$h_{z,i}$	هزینه نگهداری موجودی هر واحد محصول زدر واحد زمان بین ماشین i و i+1
h_j	هزینه نگهداری هر واحد محصول نهایی زدر واحد زمان
m_f	تعداد ماشین در کارخانه f
μ_f	سرعت نسبی ماشین‌ها در کارخانه f
T_f	زمان چرخه کارخانه f
M	عدد بزرگ مثبت

۱-۳- ارائه مدل غیرخطی

قبل از ارائه مدل، متغیرها به صورت زیر تعریف می‌شوند:

$X_{j,k,f}$: متغیر باینری؛ اگر محصول j موقعیت k را در کارخانه f اشغال کند، مقدار ۱ می‌گیرد و در غیر این صورت، صفر است.

$Y_{j,f}$: متغیر باینری؛ اگر محصول زدر کارخانه f تولید شود، مقدار ۱ می‌گیرد و در غیر این صورت، صفر است.

β_f : متغیر باینری، اگر کارخانه f برای تولید استفاده شود، مقدار ۱ می‌گیرد و در غیر این صورت، صفر است.

$S_{z,i,f}$: متغیر پیوسته؛ زمان شروع محصول زروی ماشین i در کارخانه f.

مدل مسئله زمان‌بندی در جریان کارگاهی جایگشتی توزیع‌شده با کارخانه‌های

متفاوت به صورت زیر تعریف می‌شود:

$$\begin{aligned} \text{Minimize } Z = & \sum_{f=1}^g [\sum_{j=1}^n \left(\frac{sc_j}{T_f} \right) + \sum_{j=1}^n \left(\left[h_j \frac{d_j}{2} \left(1 - \frac{d_j}{\frac{p_{j,m}}{\mu_f}} \right) + \right. \right. \\ & \left. \left. \sum_{i=2}^m h_{j,i-1} \frac{d_i^2}{2} \left(\frac{1}{\frac{p_{j,i}}{\mu_f}} - \frac{1}{\frac{p_{j,i-1}}{\mu_f}} \right) \right] \cdot T_f \right) + \sum_{k=1}^n \sum_{i=2}^m ([\sum_{j=1}^n X_{j,k,f} (h_{j,i-1} \cdot d_j)] [S_{j,i,f} - \\ & S_{j,i-1,f}]) + \sum_{f=1}^g \sum_{j=1}^n \sum_{i=1}^m \beta_f \cdot \{(Y_{j,f} \cdot (st_{j,i,f} + (S_{j,i,f} - S_{j,i-1,f}))) + \\ & (n \cdot g)\} \end{aligned} \quad (1)$$

Subject to:

$$\sum_{f=1}^g Y_{j,f} = 1 \quad \forall_j \quad (2)$$

$$\sum_{j=0}^n \sum_{f=1}^g X_{j,k,f} = 1 \quad \forall_{k,k \neq j} \quad (3)$$

$$\sum_{k=1}^n X_{0,k,f} = m_f \quad \forall_f \quad (4)$$

$$\sum_{f=1}^g X_{0,k,f} \leq 1 \quad \forall_k \quad (5)$$

$$\sum_{f=1}^g X_{j,k,f} + X_{k,j,f} = 1 \quad \forall_{j,k,k \neq j} \quad (6)$$

$$\sum_{k=1}^n \sum_{f=1}^g X_{j,k,f} \leq 1 \quad \forall_{j,j \neq k} \quad (7)$$

$$S_{j,i,f} \geq S_{j,i-1,f} + \frac{d_j \cdot T_f}{p_{j,i-1}/\mu_f} \quad \forall_{j,f,i > 1} \quad (8)$$

$$\begin{aligned} S_{j,i,f} \geq & S_{k,i,f} + \frac{d_k \cdot T_f}{p_{k,i}/\mu_f} + st_{j,i,f} - M(1 - X_{j,k,f}) - M(1 - Y_{j,f}) - \\ & M(1 - Y_{k,f}) \end{aligned} \quad \forall_{j,k \neq j,i,f} \quad (9)$$

$$S_{k,i,f} \geq S_{j,i,f} + \frac{d_j \cdot T_f}{p_{j,i}} + st_{k,i,f} - M \cdot X_{j,k,f} - M(1 - Y_{j,f}) - \quad \forall_{j,k \neq j,i,f} \quad (10)$$

$$M(1 - Y_{k,f}) \\ T_f \geq S_{j,i,f} + \frac{d_j \cdot T_f}{p_{j,i}} + st_{k,i,f} - S_{k,i,f} - M(1 - X_{j,k,f}) - \quad \forall_{j,k \neq j,i,f} \quad (11)$$

$$M(1 - Y_{j,f}) - M(1 - Y_{k,f}) \\ T_f \geq S_{k,i,f} + \frac{d_k \cdot T_f}{p_{k,i}} + st_{j,i,f} - S_{j,i,f} - M \cdot X_{j,k,f} - M(1 - Y_{j,f}) - \quad \forall_{j,k \neq j,i,f} \quad (12)$$

$$M(1 - Y_{k,f}) \\ T_f \geq \sum_{j=1}^n Y_{j,f} \cdot \left(st_{j,i,f} + \frac{d_j \cdot T_f}{p_{j,i} / \mu_f} \right) \quad \forall_{f,i} \quad (13)$$

$$S_{j,i,f} \geq st_{j,i,f} \quad \forall_{j,i,f} \quad (14)$$

$$T_f, S_{j,i,f} \geq 0 \quad \forall_{j,i,f} \quad (15)$$

$$X_{j,k,f}, \beta_f \in \{0, 1\} \quad \forall_{j,k,f} \quad (16)$$

$$Y_{j,f} \in \{0, 1\} \quad \forall_{j,f} \quad (17)$$

تابع هدف ۱ مجموع هزینه‌های راه‌اندازی، موجودی در جریان و موجودی محصول نهایی در واحد زمان و هزینه کل تولید را که شامل هزینه تخصیص نامتوازن محصولات بین کارخانه‌ها است، حداقل می‌کند. محدودیت ۲ تضمین می‌کند که هر محصول دقیقاً به یک کارخانه تخصیص می‌یابد. محدودیت ۳ بیان می‌کند که محصول زدر کارخانه f فقط به یک ماشین نیاز دارد و تضمین می‌کند که یک محصول می‌تواند توسط هر کارخانه‌ای پردازش شود و همچنین توسط یک ماشین در هر کارخانه تولید می‌شود. رابطه ۴ تضمین می‌کند که محصول ساختگی صفر دقیقاً یک پس‌نیاز روی هر ماشین در کارخانه دارد. محدودیت ۵ تضمین می‌کند که محصول k می‌تواند حداکثر به موقعیت اول هر کارخانه اختصاص یابد. محدودیت ۶ بیان می‌کند که یک محصول نمی‌تواند در یک زمان هم مقدم و هم مؤخر محصول

دیگری باشد. رابطه ۷ نشان می‌دهد که هر محصول حداکثر دارای یک پس‌نیاز است. محدودیت ۸ بیان می‌کند که پردازش محصول زروی ماشین i نمی‌تواند قبل از تکمیل پردازش آن روی ماشین $i-1$ شروع شود. محدودیت‌های ۹ و ۱۰ اطمینان می‌دهند که یک ماشین حداکثر یک محصول را در واحد زمان می‌تواند پردازش کند. محدودیت‌های ۱۱ و ۱۲ زمان چرخه را محاسبه می‌کنند؛ درحالی‌که محدودیت ۱۳ محدودیت زمان هر کارخانه را برای تولید محصولات تعیین شده مشخص می‌کند. محدودیت ۱۴ تضمین می‌کند که محصول اول در هر کارخانه بعد از راه‌اندازی آن شروع می‌شود. محدودیت‌های ۱۵ و ۱۶ و ۱۷ متغیرهای تصمیم را تعریف می‌کنند.

۲-۳- خطی‌سازی مدل پیشنهادی

مدل پیشنهادی به دلیل وجود جمله‌های غیرخطی در تابع هدف و محدودیت‌ها، جزء مدل برنامه‌ریزی غیرخطی عدد صحیح مختلط است. به منظور حذف جمله‌های غیرخطی، ما مشابه روش‌های خطی‌سازی رودریگز و همکاران [۱۳]، یو و گراس من [۱۴] و پاکزاد و همکاران [۱۵] عمل می‌کنیم. در مدل پیشنهادی، جمله‌های غیرخطی که به دلیل وجود متغیرهای پیوسته T و S و متغیرهای باینری X ، Y و β در تابع هدف و محدودیت ۱۳ که به دلیل ضرب متغیر پیوسته T در متغیر باینری Y حالت‌های غیرخطی را ایجاد کرده‌اند، می‌بایست به صورت زیر خطی شوند:

$$\begin{aligned} \text{Minimize } Z = & \sum_{f=1}^g \left(\sum_{j=1}^n \frac{sc_j}{\sqrt{\frac{\sum_{j=1}^n sc_j}{A}}} + \sqrt{\sum_{j=1}^n sc_j \cdot A} \right) + \sum_{f=1}^g \sum_{k=1}^n \sum_{i=2}^m \left(\left[\sum_{j=1}^n (G_{j,i,k,f} - \right. \right. \\ & \left. \left. G_{j,i-1,k,f}) (h_{j,i-1} \cdot d_j) \right] \right) \\ & + \sum_{f=1}^g \sum_{j=1}^n \sum_{i=1}^m W_{j,i,f} \cdot st_{j,i,f} + (GW_{j,i,f} - GW_{j,i-1,f}) \\ & + \beta_f \cdot (n \cdot g) \end{aligned} \quad (1)$$

Subject to:

تکرار محدودیت‌های ۲ تا ۱۷

$$T_f \geq \sqrt{\frac{\sum_{j=1}^n sc_j}{A}} \quad (18)$$

$$G_{j,i,k,f} \leq X_{j,k,f} \cdot M \quad \forall_{j,k,i,f} \quad (19)$$

$$G_{j,i,k,f} \geq S_{k,i,f} - (1 - X_{j,k,f}) \cdot M \quad \forall_{j,k,i,f} \quad (20)$$

$$G_{j,i,k,f} \leq S_{k,i,f} + (1 - X_{j,k,f}) \cdot M \quad \forall_{j,k,i,f} \quad (21)$$

$$GW_{j,i,f} \leq W_{j,i,f} \cdot M \quad \forall_{j,k,i,f} \quad (22)$$

$$GW_{j,i,f} \geq S_{j,i,f} - (1 - W_{j,i,f}) \cdot M \quad \forall_{j,k,i,f} \quad (23)$$

$$GW_{j,i,f} \leq S_{j,i,f} + (1 - W_{j,i,f}) \cdot M \quad \forall_{j,k,i,f} \quad (24)$$

$$W_{j,i,f} \leq \beta_f \quad \forall_{j,i,f} \quad (25)$$

$$W_{j,i,f} \leq Y_{j,f} \quad \forall_{j,i,f} \quad (26)$$

$$W_{j,i,f} \geq Y_{j,f} + \beta_f - 1 \quad \forall_{j,i,f} \quad (27)$$

$$G_{j,i,k,f}, W_{j,i,f}, GW_{j,i,f} \geq 0 \quad \forall_{j,k,i,f} \quad (28)$$

$$A = \sum_{j=1}^n \left(\left[h_j \frac{d_j}{2} \left(1 - \frac{d_j}{\mu_f} \right) + \sum_{i=2}^m h_{j,i-1} \frac{d_j^2}{2} \left(\frac{1}{\mu_f} - \frac{1}{\mu_f} \right) \right] \right) \quad (29)$$

در اینجا برای ضرب یک متغیر دودویی X در متغیر پیوسته S از متغیر پیوسته G و محدودیت‌های ۱۹ تا ۲۱ استفاده شده است. برای ضرب دو متغیر دودویی Y و β از متغیر پیوسته W و محدودیت‌های ۲۵ تا ۲۷ استفاده شده است. در ادامه، در تابع هدف بجای ضرب متغیرهای S، Y و β در یکدیگر از ضرب متغیرهای S و W استفاده

شده و همانند مرحله قبل متغیر GW و محدودیت‌های ۲۲ تا ۲۴ جایگزین شده است. عبارت اول در تابع هدف و همچنین محدودیت‌های ۱۸ و ۲۹ برای رفع حالت غیرخطی به دلیل وجود متغیر T است.

۴- تولید مسئله

عملکرد مدل پیشنهادی و همچنین روش‌های حل از طریق ۱۸۰ مسئله در ابعاد کوچک، متوسط و بزرگ مورد مطالعه قرار گرفته است. ما مسائل مختلفی را با ۳، ۵ و ۱۰ محصول و ۳، ۴، ۵، ۸، ۱۰ و ۱۵ ماشین و ۲، ۴، ۵، ۶ و ۱۰ کارخانه در سه اندازه در نظر گرفته‌ایم. هر مجموعه مسئله از ۲۰ مسئله‌ای که به‌طور تصادفی بر اساس جدول ۲ تولید شده‌اند، بررسی می‌شوند. در اینجا، ۹ ترکیب مختلف از n، m و g وجود دارد که شامل سه اندازه مسائل کوچک (یعنی ۳×۳×۲، ۳×۴×۲ و ۴×۵×۴)، مسائل متوسط (یعنی ۵×۸×۵، ۵×۱۰×۶ و ۱۰×۵×۵) و مسائل بزرگ (یعنی ۵×۱۵×۴، ۱۰×۱۰×۱۰ و ۱۵×۱۵×۱۰) است. ۲۰ مثال برای هر ترکیب و در مجموع ۱۸۰ مثال مختلف تولید شده است.

جدول ۲ تولید مسائل نمونه

توضیحات	پارامتر
اندازه کوچک (۳ و ۵)، اندازه متوسط (۵) و اندازه بزرگ (۱۰)	n
اندازه کوچک (۳، ۴ و ۵)، اندازه متوسط (۸، ۱۰ و ۱۵) و اندازه بزرگ (۵، ۱۰ و ۱۵)	m
اندازه کوچک (۲ و ۴)، اندازه متوسط (۴، ۵ و ۶) و اندازه بزرگ (۵ و ۱۰)	g
توزیع یکنواخت (۱،۱۰)	$h_{j,i}$
توزیع یکنواخت (۱۰،۱۷۰)×٪۱۰	h_j
عدد تصادفی در فاصله (۱،۲)	μ_f
۱۰۰۰	M
توزیع یکنواخت (۳۰۰،۹۰۰۰)	$p_{j,i}$
توزیع یکنواخت (۵۰،۵۰۰)	d_j
توزیع یکنواخت (۰/۰،۰/۱/۵)	$st_{j,i}$
توزیع یکنواخت (۰،۱) × ۱۰۰۰ + ($15000 \times \sum_{i=1}^m st_{j,i}$)	sc_j

۵- الگوریتم‌های فراابتکاری

در این بخش، الگوریتم‌های فراابتکاری مورد استفاده به‌طور مختصر توضیح داده می‌شود. برای حل مدل در حالت غیرخطی، از الگوریتم ترکیبی بهینه‌سازی میرایی ارتعاش با الگوریتم‌های رقابت امپریالیستی و شبیه‌سازی تبرید و همچنین از الگوریتم ژنتیک و الگوریتم بهینه‌سازی میرایی ارتعاش ساده برای مقایسه روش‌های حل استفاده شده است.

طراحی کروموزوم مناسب، اولین قدم برای داشتن یک الگوریتم فراابتکاری موفق است؛ زیرا از قوانین احتمالی در هر کروموزوم برای ایجاد جمعیتی از کروموزوم‌ها استفاده می‌شود که بیانگر یک جواب احتمالی خوب است. در این مدل، هر کروموزوم نشان‌دهنده یک جواب است؛ به‌صورت:

(۱) تخصیص محصولات به کارخانه‌ها؛

(۲) اولویت تولید هر محصول روی هر ماشین.

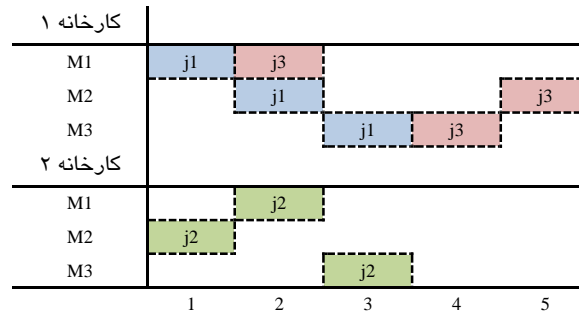
یک کروموزوم ترکیبی از ژن‌هاست. هر ژن شامل سه پارامتر تعداد کارخانه (f)، تعداد ماشین (i) و تعداد محصول (j) است. شکل ۱ نشان‌دهنده یک نمونه کروموزوم برای تخصیص و زمان‌بندی ۳ محصول با ۲ کارخانه است که هر کارخانه دارای ۳ ماشین است. نتایج زمان‌بندی در شکل ۲ نشان داده شده است.

۱۱۱	۱۲۳	۱۲۱	۱۳۱	۲۲۲	۲۱۲	۲۳۲	۱۱۳	۱۳۳
-----	-----	-----	-----	-----	-----	-----	-----	-----

شکل ۱ نمونه کروموزوم A

در شکل ۱، ژن اول (۱۱۱) نشان‌دهنده این است که محصول ۱ (j1) به ماشین ۱ در کارخانه ۱ (i1 و f1) تخصیص داده شده است.

قدم دوم، کپی‌کردن اولین و دومین موقعیت از ژن کروموزوم A در ژن کروموزوم B و تغییر آخرین موقعیت کروموزوم A در آخرین موقعیت ژن کروموزوم B از طریق تولید عدد صحیح تصادفی بین ۱ و تعداد محصولات است. اگر دو ژن همسایه جدید در کروموزوم B دقیقاً شبیه کروموزوم B باشد، موقعیت این دو ژن تغییر می‌کند و عدد صحیح تصادفی دیگری تولید می‌شود.



شکل ۲ نتایج زمان‌بندی کروموزوم نمونه A

شکل ۳ نمونه کروموزوم B را نشان می‌دهد. هدف، حداقل کردن هزینه کل سیستم است که مطابق با تابع هدف اصلی مدل است.

۱۱۳	۱۲۲	۱۲۱	۱۳۲	۲۲۳	۲۱۲	۲۳۳	۱۱۱	۱۳۱
-----	-----	-----	-----	-----	-----	-----	-----	-----

شکل ۳ نمونه کروموزوم B

در این مقاله، تقاطع یکنواخت و جهش یکنواخت بکار برده شده است. با توجه به پیچیدگی استفاده از روش‌های دقیق به دلیل وجود متغیرهای گسسته و همچنین داشتن جواب نزدیک‌تر به بهینه با استفاده از الگوریتم‌های فراابتکاری، در این مقاله مقایسه الگوریتم‌های HVDO، GA و VDO مطرح شده است.

۵-۱- الگوریتم ژنتیک

الگوریتم ژنتیک یک روش جستجوی فراابتکاری بر اساس فرآیند طبیعی تکامل است که توسط هلند [۱۶] ارائه شده است. این الگوریتم بر پایه سازوکار انتخاب طبیعی و ژنتیک استوار است. مراحل الگوریتم ژنتیک به صورت زیر است:

- ۱- پارامترهای الگوریتم شامل احتمال تقاطع (Pc)، احتمال جهش (Pm)، اندازه جمعیت (Npop) و حداکثر تکرار (Nit) را مشخص کنید؛
- ۲- جمعیت اولیه را به صورت تصادفی و برحسب اندازه جمعیت به دست آورید؛
- ۳- تابع هدف را محاسبه کنید؛

- ۴- نفرات را با استفاده از روش چرخه رولت^۱، انتخاب کنید؛
- ۵- عملگر تقاطع را برای هر زوج کروموزوم با احتمال تقاطع مشخصی بکار برید؛
- ۶- عملگر جهش را برای هر کروموزوم با احتمال جهش مشخصی بکار برید؛
- ۷- جمعیت جدید را با توجه به نتایج جمعیت فعلی جایگزین نمایید؛
- ۸- اگر شرط خاتمه (حداکثر تعداد تکرار) برآورده شد، متوقف شوید و در غیر این صورت به مرحله ۳ برگردید.

۲-۵- الگوریتم بهینه‌سازی میرایی ارتعاش ترکیبی (HVDO)

بسیاری از مسائل مهم در دینامیک را حرکت‌هایی تشکیل می‌دهند که پیرامون وضع تعادل یک زره، یک جسم یا یک سیستم به وقوع می‌پیوندند. زره، جسم یا سیستم به حرکتی وادار می‌شود که به آن حرکت ارتعاشی می‌گویند. تمام اجسامی که دارای جرم و خاصیت کشسانی هستند، می‌توانند حرکت ارتعاشی داشته باشند. هر حرکت ارتعاشی تحت تأثیر یک نیروی درون سیستم یا نیروی محرکه ایجاد می‌گردد. وقتی منبع انرژی یک نوسانگر قطع شود، بخشی از انرژی تلف می‌شود، دامنه نوسان^۲ رفته‌رفته کم می‌شود و بالاخره نوسانگر از نوسان بازمی‌ماند. این فرآیند را میرا کردن ارتعاش یا فرآیند میرایی می‌گویند. شناسایی عوامل میرایی در ارتعاش یک سیستم اهمیت ویژه‌ای دارد. در مواردی نیاز است اثر میرایی چنان زیاد شود که از ارتعاشات ویرانگر جلوگیری شود. از طرفی در بسیاری از موارد برای جلوگیری از میرا شدن نوسانات یک سیستم باید انرژی تلف‌شده به‌گونه‌ای جبران شود؛ اغلب به سیستم یک نیروی خارجی اعمال می‌شود که یک نیروی تناوبی است. در هر دوره، این نیرو به‌گونه‌ای عمل می‌کند که دامنه نوسان مقداری ثابت بماند. در این حالت به نوسان سیستم، نوسان واداشته^۳ گفته می‌شود. در حالتی که بسامد نیروی خارجی وارد بر سیستم با بسامد طبیعی و ذاتی آن مساوی شود، دامنه نوسان به بیشترین مقدار خود می‌رسد. در این حالت با شرایطی مواجه می‌شویم که به آن تشدید^۴ گفته می‌شود. هرچقدر دامنه نوسان بیشتر شود، فراوانی پاسخ بیشتر می‌شود و هرچقدر دامنه نوسان کمتر شود، فراوانی پاسخ کمتر خواهد بود. به عبارت دیگر، در دامنه‌های نوسان بالا به دلیل بزرگ‌تر بودن دامنه،

1. Roulette Wheel
2. Amplitude
3. Forced
4. Resonance

محدوده یا فضای بزرگ‌تری در اختیار بوده و احتمال رخ دادن پاسخ‌های جدیدتر زیادتر است و برعکس در دامنه‌های نوسان پایین احتمال رخ دادن یک پاسخ جدید کمتر خواهد بود. به نحوی که وقتی دامنه نوسان به صفر میل می‌کند، احتمال رخ دادن یک پاسخ جدید به صفر میل می‌کند و سیستم از حالت نوسان بازمی‌آید. در این الگوریتم که توسط مهدی‌زاده و همکاران [۱۷] ارائه شده است، ماکزیمم دامنه نوسان را در نظر می‌گیریم و در تناوب زمانی مورد مطالعه هر بار یک پاسخ را انتخاب می‌کنیم و مقدار انرژی را محاسبه می‌کنیم و به صورت تصادفی یک پاسخ دیگر انتخاب می‌کنیم، آنگاه انرژی پاسخ‌های جدید و فعلی را محاسبه می‌کنیم. اگر تفاوت انرژی کوچک‌تر یا مساوی صفر بود، سیستم از سطح بالاتر انرژی به سطح پایین‌تر انرژی منتقل می‌شود و در نتیجه، پاسخ جدید پذیرفته می‌شود. در غیر این صورت، پاسخ جدید با احتمال ریلی^۱ پذیرفته شده و با ارتعاش واداشته، دامنه نوسان ثابت نگه‌داشته می‌شود تا جواب جدید دیگری تولید شود. در نهایت به شرط توقف مراجعه می‌شود. اگر ضابطه توقف برقرار نباشد، از برنامه میرا نمودن جهت کاهش دامنه نوسان بهره گرفته و مجدداً قدم‌های فوق در دامنه نوسان جدید که کمتر از دامنه نوسان قبلی است، تکرار می‌شوند. در جدول ۳، مفاهیم بهینه‌سازی با مفاهیم الگوریتم VDO با هم مقایسه می‌شوند. الگوریتم VDO با تولید حل‌های تصادفی در فضای جستجو شروع می‌شود. سپس پارامترهای الگوریتم شامل دامنه اولیه (A_0)، بیشترین تعداد تکرار در هر دامنه (L)، ضریب میرایی (γ) و انحراف استاندارد (σ) مقداردهی می‌شوند. سپس جواب‌ها به وسیله مقادیر تابع هدف (E) ارزیابی می‌شوند [۱۷].

جدول ۳ مقایسه مفاهیم بهینه‌سازی و الگوریتم VDO

میرایی ارتعاش	بهینه‌سازی
وضعیت پاسخ سیستم	جواب‌های مسئله
تغییرات انرژی	تغییرات تابع هدف
حالت حداقل انرژی	حل بهینه
تغییر وضعیت پاسخ	حل‌های همسایه
دامنه نوسان	پارامتر کنترلی
میرایی صفر	حداقل تابع هدف
درجه آزادی	تعداد متغیر تصمیم

1. Rayleigh distribution

حلقه اول، یک جواب تصادفی تولید می‌کند و سپس با استفاده از ساختار همسایگی یک جواب جدید به دست می‌آید و بهترین آن‌ها انتخاب می‌شود. جواب با کمترین مقدار می‌تواند به عنوان تابع توزیع رایلی انتخاب شود. در حقیقت، جواب جدید پذیرفته می‌شود، اگر [۱۷]:

$$\Delta = E(\text{جواب جدید}) - E(\text{فعلی جواب}) < 0 \quad (30)$$

اگر $\Delta > 0$ باشد، یک عدد تصادفی r بین (۰ و ۱) تولید کنید. حل جاری با شرط زیر پذیرفته می‌شود [۱۷]:

$$r < 1 - \exp\left(-\frac{A^2}{2\sigma^2}\right) \quad (31)$$

در حلقه دوم، دامنه تنظیم می‌شود؛ به طوری که در هر تکرار کاهش می‌یابد. الگوریتم زمانی متوقف می‌شود که شرط توقف برقرار باشد:

$$A_t = A_0 \exp\left(-\frac{t}{Q}\right) \quad (32)$$

با توجه به نوع حرکت گام‌های الگوریتم VDO و وجود مرحله ایجاد همسایگی، الگوریتم ترکیبی پیشنهادی در حرکت حلقه کلی از ترکیب الگوریتم VDO و SA و در ایجاد همسایگی از الگوریتم ICA استفاده می‌کند. ایجاد همسایگی در الگوریتم ICA با عنوان حرکت مستعمره‌های یک فرمانروایی به سوی امپریالیست مطرح می‌شود. یک مستعمره به اندازه x واحد به سمت امپریالیست حرکت می‌کند. x یک متغیر تصادفی بین صفر و $(\beta \times d)$ است که β یک عدد بزرگ‌تر از ۱ و d فاصله بین مستعمره و امپریالیست است. به این دلیل $\beta > 1$ است که مستعمره‌ها از هر دو جهت به یک امپریالیست نزدیک می‌شوند. برای توضیحات بیشتر در مورد الگوریتم‌های ICA و SA به علاقه‌بندها و حاجی‌پور [۱۸] رجوع شود. شبکه‌کد و فلوچارت الگوریتم HVDO در زیر نشان داده شده است:

مرحله اول: تولید حل اولیه شدنی؛

مرحله دوم: تعیین پارامترهای الگوریتم شامل:

A_0 : دامنه اولیه ارتعاشات (مقادیر بالا، کاهش سرعت همگرایی ولی امکان

جستجوی بیشتر)؛

t : شمارنده تعداد دفعات کاهش دامنه؛

γ : ضریب میرایی (کاهش دامنه نوسان)؛

L : تعداد تکرار حلقه واداشته (پارامتر کنترل زمان دستیابی به جواب نهایی)؛

$$p(A) = \frac{A}{\sigma^2} \cdot \exp\left(-\frac{A^2}{2\sigma^2}\right); A > 0$$

Q : دو برابر عکس ضریب میرایی ($Q = 2/\gamma$)؛

مرحله سوم: محاسبه تابع هدف برای حل اولیه (E_0)؛

مرحله چهارم: انجام حلقه داخلی از L تا رسیدن به $L < L_{max}$ ؛

مرحله پنجم: تولید همسایگی جدید از حل موجود با استفاده از تابع یکنواخت و

محاسبه تابع هدف جدید (E) با استفاده از ICA

مرحله ششم: پذیرش حل جدید: اگر $\Delta > 0$ باشد و $\Delta = E - E_0$ حل جدید پذیرفته شده و

در غیر این صورت، عدد تصادفی r بین صفر و ۱ تولید می‌شود که اگر کمتر از $1 - r$

باشد، حل جدید قبول می‌شود و در غیر این صورت، حل قبلی موردقبول

خواهد بود و اگر شرایط خروج از حلقه فراهم نشده باشد، مجدداً وارد مرحله پنجم

می‌شویم.

مرحله هفتم: تنظیم دامنه. در این مرحله، از $A_t = A_0 \exp(-\frac{t}{Q})$ برای کاهش دامنه

ارتعاش در هر مرحله استفاده می‌شود. اگر شرایط خروج از حلقه کلی فراهم شده باشد،

به مرحله هشتم می‌رویم؛ در غیر این صورت، به مرحله چهارم باز می‌گردیم.

مرحله هشتم: معیار توقف ($0 \geq A$).

۳-۵- تنظیم پارامتر

در این بخش به دلیل اینکه همه الگوریتم‌های فراابتکاری به مقادیر پارامترهای اصلی

آن بستگی دارند، روی تنظیم پارامترهای ورودی دو الگوریتم تمرکز شده است. برای

این کار از روش رویه پاسخ (RSM) استفاده شده است. RSM مجموعه‌ای از

روش‌های آماری و ریاضی است که برای بهینه‌سازی در شرایطی مفید است که

چندین متغیر ورودی به‌طور بالقوه بر اندازه‌گیری عملکرد یا ویژگی کیفی که پاسخ نامیده می‌شود، تأثیرگذار باشد [۱۹]. هدف این است که سطح پارامترهای الگوریتم (به‌عنوان متغیرهای ورودی) را پیدا کنیم تا متغیر پاسخ (تابع هدف) بهینه شود. به‌عنوان مثال، اندازه جمعیت، تعداد تکرار، احتمال تقاطع و احتمال جهش در GA و اندازه جمعیت، دامنه اولیه ارتعاشات، ضریب میرایی، تعداد تکرار حلقه واداشته و پارامتر توزیع رایلی برای HVDO در نظر گرفته شده است که هرکدام دارای مقادیر ۱، صفر و ۱- در نرم‌افزار MINITAB بوده و به ترتیب نشان‌دهنده سطوح بالا، متوسط و پایین هستند. در این روش، ما از طرح مربع مرکزی با چهار عامل برای سه اندازه مسائل کوچک (یعنی $2 \times 2 \times 3$ ، $2 \times 4 \times 3$ و $4 \times 5 \times 5$)، مسائل متوسط (یعنی $5 \times 8 \times 5$ ، $6 \times 10 \times 5$ و $10 \times 10 \times 10$) و مسائل بزرگ (یعنی $4 \times 15 \times 5$) استفاده کرده‌ایم که این اعداد نشان‌دهنده $n \times m \times g$ هستند. بر اساس روش RSM استفاده‌شده، از آنجایی که مقادیر کمبود برآزندگی برای همه اندازه‌های مسائل، بزرگ‌تر از ۰/۰۵ است، بنابراین نشان می‌دهد که توابع رگرسیون برآزش شده با سطح اطمینان ۹۵٪ درست هستند.

جدول ۴ مقادیر بهینه پارامترهای ورودی الگوریتم‌های GA و HVDO

الگوریتم	پارامتر	مسئله کوچک	مسئله متوسط	مسئله بزرگ
GA	N_{pop}	۱۰	۲۲	۳۶
	N_{it}	۳۰۰	۶۸۰	۹۵۰
	P_c	۰/۹	۰/۹	۰/۸
	P_m	۰/۴	۰/۵	۰/۵
HVDO	N_{pop}	۵	۱۶	۲۵
	A_0	۶	۸	۱۰
	L	۴۰	۶۰	۸۰
	σ	۱	۱/۵	۱/۵
	γ	۰/۰۵	۰/۰۵	۱/۵

بر اساس جواب به‌دست‌آمده و با استفاده از حل توابع رگرسیون توسط LINGO، اندازه بهینه مقادیر پارامترهای GA و HVDO به‌دست آمد که در جدول ۴ ارائه شده

است. شکل ۴ نمودار همگرایی برای یک مسئله نمونه در اندازه بزرگ را برای دو الگوریتم GA و HVDO نشان می‌دهد.

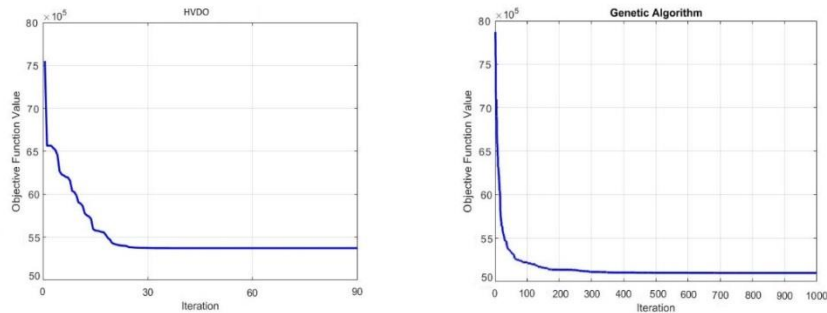
۴-۵- نتایج محاسباتی و ارزیابی الگوریتم‌ها

در این بخش، ما ۹ مسئله مختلف با سه اندازه کوچک، متوسط و بزرگ را برای مثال‌های عددی بکار برده‌ایم. برای هر اندازه ۲۰ مسئله به‌طور تصادفی تولید شده است؛ به‌منظور نمایش عملکرد الگوریتم‌های پیشنهادی، نتایج مورد تحلیل قرار گرفته است. به‌منظور کاهش عدم اطمینان برای هر مثال، ۱۰ اجرا به‌وسیله الگوریتم‌های GA، VDO و HVDO و همچنین LINGO انجام شده است. نتایج این اجراها در جداول ۵ بر اساس معیارهای زمان اجرا (CPU TIME)، حداقل مقدار تابع هدف (MIN) و شاخص انحراف نسبی (RDI^۱) ارائه شده است.

جدول ۵ مقایسه معیارهای الگوریتم‌های GA، VDO، HVDO و LINGO

CPU TIME				MIN				RDI				معیار	
HVDO	VDO	GA	LINGO	HVDO	VDO	GA	LINGO	HVDO	VDO	GA	LINGO	مسئله	
۱۴	۱۰۳	۳۲	۱۰	۸۰۲۶۲۴	۸۰۲۶۲۴	۸۰۲۶۲۴	۸۰۲۶۲۴	۰	۰	۰	۰	۳×۳×۲	کوچک
۱۷	۱۱۴	۴۸	۱۰	۱۰۴۵۰۴۱	۱۰۴۶۷۰۰	۱۰۴۵۷۲۲	۱۰۲۵۵۰۸	۰/۸۵	۱	۰/۹۱	۰	۳×۴×۲	
۲۹	۱۲۹	۷۳	۱۵	۱۷۸۴۴۳۸	۱۷۸۷۵۲۸	۱۷۸۶۰۰۳	۱۷۷۲۲۸۲	۰/۷۹	۱	۰/۹	۰	۵×۵×۴	
۹۹	۳۳۴	۱۶۷	۱۸	۴۲۱۲۱۷۱	۴۴۴۱۰۰۷	۴۳۲۲۲۰۴	۴۱۲۱۸۰۶	۰/۲۸	۱	۰/۶۲	۰	۵×۸×۵	متوسط
۹۷	۳۲۰	۱۷۳	۲۵	۴۵۲۳۸۸۹	۴۷۵۵۸۰۹	۴۶۰۴۸۲۲	۴۴۰۹۴۵۸	۰/۳۳	۱	۰/۵۶	۰	۵×۱۰×۶	
۲۶۶	۴۰۰	۳۱۶	۳۴	۴۷۳۴۰۱۷	۴۸۷۱۹۵۱	۴۷۸۱۰۱۳	۴۶۸۷۵۹۸	۰/۲۵	۱	۰/۵۴	۰	۱۰×۵×۵	
۱۲۵۲	۱۳۷۰	۱۰۸۲	۳۴۵	۵۹۳۵۱۶۱	۶۰۵۵۸۵۳	۵۹۸۳۵۷۹	۵۸۶۱۳۰۳	۰/۳۷	۱	۰/۶۲	۰	۵×۱۵×۴	بزرگ
۲۱۳۴	۱۹۲۵	۱۷۴۱	۱۰۱۰	۱۴۸۴۴۶۵۹	۱۵۶۱۸۲۸۰	۱۵۳۲۳۶۵۹	۱۴۳۰۷۳۹۳	۰/۴	۱	۰/۷۷	۰	۱۰×۱۰×۱۰	
۳۷۱۲	۲۷۸۵	۲۳۸۳	۱۱۱۸	۲۳۳۹۲۹۵۱	۲۳۳۹۸۱۶۰	۲۳۳۶۰۳۷۵	۲۱۳۹۱۰۵۹	۰/۴۷	۱	۰/۶	۰	۱۵×۱۵×۱۰	

1. Relative deviation index



شکل ۴ نمودار همگرایی GA و HVDO برای مسئله با اندازه بزرگ

نتایج الگوریتم‌ها بر اساس معیارهای زیر با یکدیگر مقایسه می‌شوند:
 ۱- RDI: این معیار برای اندازه‌گیری عملکرد نسبی الگوریتم‌های حل است (چوی و کیم [۲۰]) و از رابطه زیر به دست می‌آید:

$$RDI(x) = \frac{OB_x - OB_{best}}{OB_{worst} - OB_{best}} \quad (۳۳)$$

RDI(x)

بیانگر شاخص انحراف نسبی الگوریتم x ، OB_x مقدار تابع هدف با استفاده از الگوریتم x ، OB_{best} و OB_{worst} به ترتیب نشان‌دهنده بهترین و بدترین جوابی است که توسط الگوریتم‌ها برای هر مسئله به دست آمده است. مشخص است که الگوریتم با کمترین RDI به عنوان بهترین الگوریتم انتخاب می‌شود.

۲- بهترین جواب (MIN): الگوریتم با کمترین مقدار تابع هدف به عنوان بهترین الگوریتم انتخاب می‌شود.

۳- CPU TIME: زمان محاسباتی الگوریتم‌ها تا رسیدن به بهترین جواب است. برای تمامی مسائل با اندازه‌های مختلف، جواب الگوریتم‌ها با جواب LINGO که از حل مدل خطی‌سازی شده به دست آمده، مقایسه شده‌اند. به‌طور کلی و بر اساس نتایج جدول ۵ و شکل ۵، واضح است که الگوریتم HVDO عملکرد بهتری نسبت به سایر الگوریتم‌ها بر اساس ۳ معیار مطرح شده دارد. با استفاده از میانگین جواب‌ها،

الگوریتم HVDO در معیار MIN نسبت به GA دو درصد و نسبت به VDO چهار درصد بهبود در پیدا کردن جواب بهینه دارد. همچنین با استفاده از میانگین زمان‌های حل، الگوریتم GA در معیار CPU TIME نسبت به HVDO ۱۰ درصد و نسبت به VDO ۲۰ درصد بهبود دارد. شاخص RDI نیز نشان می‌دهد که الگوریتم HVDO جواب‌های نزدیک‌تر به بهینه را به دست آورده است. به‌طور خلاصه، جدول ۶ مقایسه این الگوریتم‌ها را نسبت به LINGO (به‌عنوان مرجع) برای میانگین مقادیر مثال‌های مدل پیشنهادی نشان می‌دهد. همان‌طور که مشخص است، HVDO در معیارهای MIN و RDI عملکرد بهتری نسبت به سایر الگوریتم‌ها داشته است، ولی از نظر زمان به دست آوردن جواب در رتبه پایین‌تری نسبت به GA قرار دارد.

جدول ۶ مقایسه میانگین جواب‌های الگوریتم‌های حل

معیار	LINGO	GA	VDO	HVDO
CPU Time	مرجع	رتبه ۱	رتبه ۲	رتبه ۲
MIN	مرجع	رتبه ۲	رتبه ۲	رتبه ۱
RDI	مرجع	رتبه ۲	رتبه ۲	رتبه ۱

۶- نتیجه‌گیری

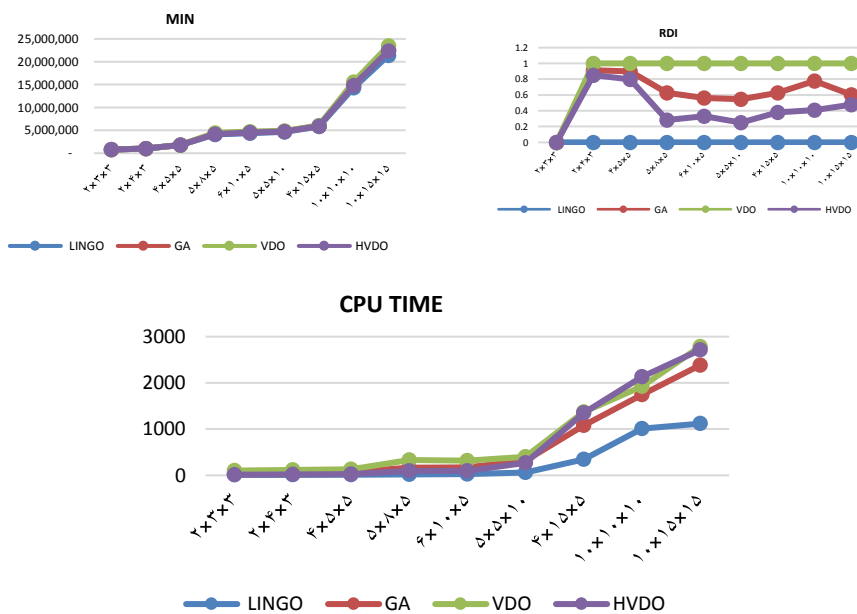
این مقاله یک مدل غیرخطی برای مسئله زمان‌بندی و اندازه انباشته اقتصادی در محیط جریان کارگاهی جایگشتی توزیع شده با چندین کارخانه و ماشین متفاوت را طراحی و ارائه کرده است. در این مسائل، دو تصمیم مرتبط در نظر گرفته می‌شود: تخصیص محصولات به کارخانه‌ها و توالی محصولات تخصیصی به هر کارخانه در شرایطی که سرعت ماشین‌ها و به‌تبع آن، کارخانه‌ها نیز متفاوت هستند. همچنین خطی‌سازی مدل پیشنهادی نیز به‌منظور به دست آوردن جواب بهینه در این مقاله توضیح داده شده و مدل‌های برنامه‌ریزی خطی و غیرخطی عدد صحیح مختلط مورد تحلیل قرار گرفته است. به دلیل اینکه مدل پیشنهادی NP-hard است، از الگوریتم‌های فراابتکاری GA، VDO و HVDO برای حل آن استفاده شده و با استفاده از سه معیار CPU Time، MIN و RDI مقایسه شده‌اند.

بر اساس رویکرد تنظیم پارامتری که در جدول ۴ ارائه شد و همچنین تحلیل

معیارهای مقایسه و نمودارهای مربوطه در شکل ۵ و جداول ۵ و ۶ نشان داده شد که مدل خطی‌سازی شده بر اساس سه معیار مقایسه، نسبت به الگوریتم‌های فراابتکاری برای کلیه مسائل و معیارها بهتر عمل کرده است. همچنین HVDO عملکرد بهتری در معیارهای MIN و RDI داشته است و GA در مورد CPU Time بهتر عمل کرده است.

۶-۱- پیشنهاد برای تحقیقات آتی

پیشنهاد می‌شود برای تحقیقات آتی، این مدل برای حالتی که بافر بین ماشین‌ها محدود بوده یا تابع هدف به صورت چندهدفه است و اثر پیروی و یادگیری در سیستم وجود داشته باشد، توسعه داده شود. همچنین برای روش‌های حل می‌توان از سایر الگوریتم‌های فراابتکاری یا روش‌های دقیق، مانند روش تجزیه و شاخه و حد، استفاده نمود.



شکل ۵ مقایسه نموداری روش‌های حل برای ۹ مسئله مختلف بر اساس معیارهای MIN و RDI و CPU TIME

۷- منابع

- [1] Olfat, L. Total Tardiness Minimization in Flow Shop with Intermediate Due Dates, *Journal of Modern Researches in Decision Making*, Vol. 2 No. 3 (in Persian), 2017, pp. 25-47,
- [2] Naderi, B., Ruiz, R. The distributed permutation flowshop problem, *Computers and Operations Research*, 2009, Volume 37, Issue 4, pp 754-768, doi:10.1016/j.cor.2009.06.019.
- [3] Ounniche, J, Boctor, F.F., Martel, A. The impact of sequencing decisions on multi-item lot sizing and scheduling in flow shops, *International Journal of Production Research*, 37(10), 1999, 2253–2270.
- [4] R. M'Hallah. Minimizing total earliness and tardiness on a permutation flow shop using VNS and MIP, *Computers & Industrial Engineering*, 2014, Volume 75, Pages 142-156. <http://dx.doi.org/10.1016/j.cie.2014.06.011>.
- [5] He, H. Minimization of maximum lateness in an m-machine permutation flow shop with a general exponential learning effect, *Computers & Industrial Engineering*, 2016, Volume 97, Pages 73-83, <http://dx.doi.org/10.1016/j.cie.2016.04.010>
- [6] Li, X., Yin, M. A discrete artificial bee colony algorithm with composite mutation strategies for permutation flow shop scheduling problem, *Scientia Iranica E*, 2012, Volume 19, pp.1921–1935, doi:10.1016/j.scient.2012.10.034
- [7] Behnamian, J., Fatemi Ghomi, S.M.T. The heterogeneous multi-factory production network scheduling with adaptive communication policy and parallel machine, *Information Sciences*, 219, 2013, 181–196.
- [8] G hafoori S. and M. taghizadeh yazdi, "Proposing a Multi- Objective Mathematical Model for RCPSP and Solving it with Firefly and

- Simulated Annealing algorithms," *Journal of Modern Researches in Decision Making*, vol. vol 1 no4 (in Persian), pp.117-142,2017.
- [9] Wang, S., Wang, L., Liu, M., Xu, Y. An effective estimation of distribution algorithm for solving the distributed permutation flow-shop scheduling problem, *Int. J. Production Economics* 145, 2013, 387–396. <http://dx.doi.org/10.1016/j.ijpe.2013.05.004>
- [10] Naderi, B., Ruiz, R. A scatter search algorithm for the distributed permutation flowshop scheduling problem, *European Journal of Operational Research*, 2014, <http://dx.doi.org/10.1016/j.ejor.2014.05.024>
- [11] Lin, J., Zhang, S. An effective hybrid biogeography-based optimization algorithm for the distributed assembly permutation flow-shop scheduling problem, *Computers & Industrial Engineering*, Volume 97, Pages 128-136, 2016, <http://dx.doi.org/10.1016/j.cie.2016.05.005>
- [12] Deng, J., Wang, L. A competitive memetic algorithm for multi-objective distributed permutation flow shop scheduling problem, *Swarm and Evolutionary Computation*, 2016, <http://dx.doi.org/10.1016/j.swevo.2016.06.002>
- [13] Rodriguez, M.A., Vecchiotti, A.R., Harjunoski, I., Grossmann, I.E. Optimal supply chain design and management over a multi-period horizon under demand uncertainty. Part I: MINLP and MILP models, *Computers and Chemical Engineering* 62, 2014, 194–210.
- [14] You, F., Grossmann, I.E. *Integrated Multi-Echelon Supply Chain Design with Inventories under Uncertainty: MINLP Models, Computational Strategies*, Carnegie Mellon University, USA, 2008.
- [15] Pakzad-Moghaddam, S.H., Tavakkoli-Moghaddam, R., Mina, H. An approach for modeling a new single machine scheduling problem with deteriorating and learning effects, *Computers & Industrial Engineering*, 2014, <http://dx.DOI.org/10.1016/j.cie.2014.09.021>.

- [16] Holland, J.H. Adaptive in Natural and Artificial Systems, Ann Arbor: University of Michigan Press, 1975.
- [17] Mehdizadeh, E., Tavakkoli-Moghaddam, R., Yazdani, M. A Vibration Damping Optimization Algorithm for a Parallel Machines Scheduling Problem with Sequence-independent Family Setup Times, Applied mathematical modeling, 2015, Volume 39, Issue 22, Pages 6845-6859, Doi: <http://dx.doi.org/10.1016/j.apm.2015.02.027>
- [18] Alaghebandha, M., Hajipour, V. A soft computing-based approach to optimise queuing inventory control problem. International Journal of Systems Science, 2013, DOI:10.1080/00207721.2013.809614.
- [19] Pasandideh, SHR., Akhavan Niaki, ST., Hajipour, V. A multi objective facility location model with batch arrivals: two parameter tuned meta heuristic algorithms, J Intell Manuf, 24, 2013, 331-348.
- [20] Choi, S.-W., Kim, Y.-D. Minimizing total tardiness on a two-machine re-entrant flowshop, Eur. J. Oper. Res. 199, 2009, 375–384.