

حل مسئله‌ی زمان‌بندی تولید کارگاهی انعطاف‌پذیر با یک مرحله‌ی مونتاژ و زمان آماده‌سازی وابسته به توالی

پرویز فتاحی* (دانشیار)

گروه مهندسی صنایع، دانشگاه الزهراء، تهران

نعمه باقری‌راد (دانشجوی کارشناسی ارشد)

فاطمه دانش‌آموز (دانشجوی دکتری)

گروه مهندسی صنایع، دانشگاه بوعلی سینا، همدان

مهندسی صنایع و مدیریت شریف، زمستان ۱۳۹۷ (۱۳-۱۹)
دوری ۱-۳۳، شماره ۲/۲، ص. ۱۳-۱۹

در این مقاله مسئله‌ی زمان‌بندی تولید کارگاهی انعطاف‌پذیر با یک مرحله‌ی مونتاژ و زمان آماده‌سازی وابسته به توالی با هدف کمیته‌سازی زمان تکمیل محصولات مورد بررسی قرار می‌گیرد. این مسئله، مدلی از سیستم‌های تولیدی است که در آن هر محصول از مونتاژ مجموعه‌یی از قطعات مختلف تولید می‌شود. در ابتدا، یک مدل برنامه‌ریزی خطی عدد صحیح مختلط توسعه داده شده است. اعتبارسنجی مدل ریاضی پیشنهادی با استفاده از نرم‌افزار گمز و به‌ازای مسائل کوچک و متوسط انجام شده است. سپس با توجه به این‌که مسئله‌ی مورد بررسی NP-hard است، الگوریتم بهینه‌سازی ازدحام ذرات و دو الگوریتم فراابتکاری ترکیبی برای حل مسائل در ابعاد متوسط و بزرگ پیشنهاد شده است. نتایج عددی الگوریتم‌های پیشنهادی با الگوریتم ترکیبی لی و گائو مقایسه شده است. نتایج محاسباتی نشان می‌دهد که در ابعاد متوسط و بزرگ مسئله، الگوریتم ترکیبی بهینه‌سازی ازدحام ذرات و جست‌وجوی همسایگی متغیر نسبت به سایر الگوریتم‌ها عملکرد بهتری دارد.

واژگان کلیدی: زمان‌بندی تولید کارگاهی انعطاف‌پذیر، مونتاژ، زمان آماده‌سازی، الگوریتم‌های فراابتکاری.

۱. مقدمه

یکی از عوامل کلیدی موفقیت در هر سازمان تولیدی، تعیین برنامه‌ی زمان‌بندی و توالی عملیات در مسائل برنامه‌ریزی تولید است که نقش مهم و مؤثری در عملکرد واحد تولیدی و غیر تولیدی دارد. در نظر گرفتن هم‌زمان مراحل مختلف تولید در برنامه‌ریزی و زمان‌بندی انجام امور از اهمیت بالایی برخوردار است. چرا که بهینه‌کردن توالی انجام کارها در یک مرحله از تولید، بدون در نظر گرفتن پارامترها و ظرفیت‌های مرحله‌ی بعدی، تضمینی برای کوتاه شدن زمان کل تولید و استفاده بهینه از ماشین‌آلات و سایر ظرفیت‌های موجود نخواهد بود. این مهم موجب شده است تا طی دو دهه‌ی اخیر محققین در ارائه و توسعه‌ی مدل‌های زمان‌بندی، مراحل تولید را - که عموماً شامل دو مرحله پردازش و مونتاژ است - با هم در نظر بگیرند. مسائل زمان‌بندی که در آن‌ها مراحل ساخت قطعات و مونتاژ محصولات هم‌زمان با هم در نظر گرفته می‌شود، کاربرد زیادی در شرکت‌های بزرگ تولیدی، نظیر صنایع خودروسازی، تولید لوازم خانگی و هواپیما دارد. معمولاً در این صنایع، خطوط مونتاژی برای تولید

* نویسنده مسئول

تاریخ: دریافت ۱۳۹۵/۱۲/۱۱، اصلاحیه ۱۳۹۶/۴/۲۷، پذیرش ۱۳۹۶/۵/۱۷.

DOI: 10.24200/J65.2018.20092

محصول نهایی بعد از پردازش قطعات وجود دارد. به‌عنوان مثال، لی و همکاران^[۱] یک کاربرد از مسائل دومرحله‌یی، در تولید خودروهایی با موتور احتراقی را نشان دادند. پاتس و همکاران^[۲] نیز کاربرد دیگری از این سیستم‌ها را در کارگاه تولیدی رایانه توضیح دادند. زمان‌بندی سیستم‌های تولید کارگاهی انعطاف‌پذیر جایگاه ویژه‌یی در حوزه مدیریت تولید و مسائل بهینه‌سازی ترکیبی است. انعطاف‌پذیری در سیستم‌های تولیدی منجر به رفع مشکلات گلوگاهی، افزایش تولید، بهبود عملکرد سیستم و ایجاد مزیت رقابتی می‌شود. در مطالعات انجام شده در زمینه‌ی تولید کارگاهی انعطاف‌پذیر مرحله‌ی مونتاژ نادیده گرفته شده است. لذا در این نوشتار مسئله‌ی تولید کارگاهی انعطاف‌پذیر با عملیات مونتاژ^۱؛ که یک سیستم تولید ترکیبی دومرحله‌یی^۲ محسوب می‌شود، مورد مطالعه قرار می‌گیرد. همچنین یکی از مسائل حائز اهمیت در سیستم‌های تولید کارگاهی انعطاف‌پذیر، زمان راه‌اندازی عملیات‌هاست که در اغلب تحقیقات، زمان آماده‌سازی یا نادیده گرفته شده است یا به‌عنوان بخشی از زمان فرایند لحاظ شده است. منفک کردن زمان آماده‌سازی از زمان پردازش، باعث بهبود عملکرد در تصمیم‌گیری‌ها خواهد بود.^[۳] زمان راه‌اندازی معمولاً به‌صورت‌های مستقل

یا وابسته به عملیات قبلی ماشین در نظر گرفته می‌شود. در این نوشتار فرض شده است که زمان راه‌اندازی وابسته به عملیات قبلی ماشین باشد. در این حالت میزان زمان راه‌اندازی ماشین برای عملیات جدید، به عملیات پیشین آن ماشین بستگی دارد. در ادامه‌ی این نوشتار، در بخش ۲ به مرور ادبیات و پیشینه تحقیق مسئله پرداخته می‌شود. در بخش ۳ مسئله‌ی مورد بررسی تعریف و مفروضات، پارامترها، متغیرهای تصمیم و مدل ریاضی مسئله ارائه می‌شود. در بخش ۴ الگوریتم‌های پیشنهادی و ساختار آن‌ها شرح داده می‌شود. در بخش ۵ آزمایشات عددی و ارزیابی عملکرد الگوریتم‌های پیشنهادی ارائه می‌شود. نتیجه‌گیری از کار و تحقیقات آتی نیز در بخش ۶ ارائه می‌شود.

۲. مرور ادبیات

انواع مختلفی از مسائل زمان‌بندی در زمینه سیستم‌های تولید دومرحله‌ی وجود دارد. در ذیل، برخی از مقالات مربوط به مسائل زمان‌بندی دومرحله‌ی مورد بررسی قرار گرفته است. لی و همکاران^[۱] در سال ۱۹۹۳ برای اولین بار، مسئله‌ی زمان‌بندی تولید دومرحله‌ی با هدف کمینه‌سازی دوره‌ی ساخت را مورد مطالعه قرار دادند. آن‌ها نشان دادند که مسئله‌ی مورد نظر کاملاً NP-complete است. پاتس و همکاران^[۲] نیز دو سال بعد این مسئله را بررسی کرده و آن را به حالت عمومی تری تعمیم دادند. آن‌ها m ماشین پردازش در مرحله‌ی ساخت و یک ماشین برای مونتاژ در نظر گرفتند. حریری و پاتس^[۳] مسئله‌ی ارائه شده توسط لی و همکاران^[۱] را مورد مطالعه قرار دادند و یک حد پایین و چندین رابطه‌ی چیرگی ارائه کردند.

مسئله‌ی تولید کارگاهی با عملیات مونتاژ یکی از انواع سیستم‌های تولید دومرحله‌ی است. در این سیستم‌ها، قطعات در مرحله‌ی اول پردازش شده و بعد از آماده‌سازی قطعات، در مرحله‌ی مونتاژ با یکدیگر مونتاژ می‌شوند. ژانگ^[۴] مسئله‌ی زمان‌بندی تولید کارگاهی با عملیات مونتاژ با هدف کمینه‌سازی مجموع حداکثر زمان تکمیل و نرخ تحویل به‌موقع را مورد مطالعه قرار داد. ایشان یک الگوریتم ژنتیک^۳ با یک قاعده مبتنی بر رویه‌ی کدگذاری را برای حل مسئله ارائه داد. دانش‌آموز^[۵] با بررسی مسئله‌ی تولید کارگاهی همراه با یک مرحله‌ی مونتاژ موازی با هدف کمینه‌سازی زمان ساخت، دو الگوریتم فراابتکاری پیشنهاد کرد.

امروزه بقای هر سیستم تولیدی و خدماتی در گرو میزان قابلیت آن سیستم در سازگاری با نظر و سلیقه‌ی مشتری است. به‌منظور مطابقت با نیازهای بازار، سیستم‌های تولیدی باید انعطاف‌پذیر و کارا باشد. بنابراین انعطاف‌پذیری و زمان آماده‌سازی دو عامل مهم در مسائل زمان‌بندی هستند. فتاحی و همکاران^[۶] مسئله‌ی زمان‌بندی جریان کارگاهی ترکیبی مونتاژ دومرحله‌ی با عملیات مونتاژ و زمان آماده‌سازی با هدف کمینه‌سازی زمان تکمیل تمام محصولات را مورد مطالعه قرار دادند. آن‌ها یک الگوریتم شاخه و کران سلسله‌مراتبی و یک حد بالا براساس الگوریتم جست‌وجوی حریمانه^۴ را به‌منظور بهبود بهره‌وری ارائه کردند. کمکی و همکاران^[۸] مسئله‌ی زمان‌بندی جریان کارگاهی ترکیبی مونتاژ را مورد بررسی قرار دادند که در آن، چندین ماشین در مرحله‌ی اول و یک ماشین مونتاژ در مرحله‌ی دوم در نظر گرفته شد. آن‌ها الگوریتم سیستم ایمنی مصنوعی^۵، سیستم ایمنی مصنوعی بهبودیافته و الگوریتم‌های ابتکاری را پیشنهاد کردند. فتاحی و همکاران^[۹] مسئله‌ی زمان‌بندی جریان کارگاهی انعطاف‌پذیر مونتاژ را مورد بررسی قرار دادند. مسئله‌ی مورد نظر شامل دومرحله‌ی است که مرحله‌ی اول، یک جریان کارگاهی انعطاف‌پذیر و مرحله‌ی دوم یک ماشین مونتاژ است. آن‌ها چند الگوریتم مبتنی بر الگوریتم جانسون را پیشنهاد کردند. نجاتی و همکاران^[۱۰] مسئله‌ی جریان کارگاهی ترکیبی مونتاژ با در نظر گرفتن جریان

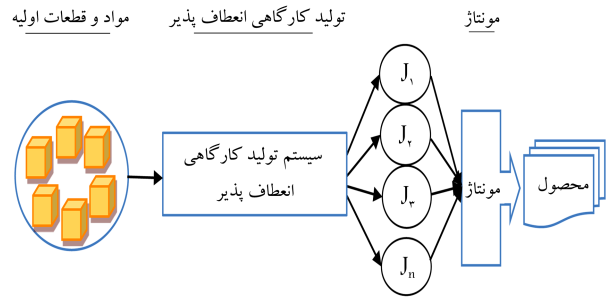
محموله، زمان آماده‌سازی وابسته به توالی و محدودیت شیفت کاری را مورد بررسی قرار دادند. آن‌ها m ماشین در مرحله‌ی اول و n ماشین در مرحله‌ی مونتاژ در نظر گرفتند. یوکویاما^[۱۱] یک مدل ریاضی برای مسئله‌ی زمان‌بندی جریان کارگاهی با عملیات آماده‌سازی و مونتاژ پیشنهاد کرد، و تابع هدف را متوسط زمان اتمام کار برای تمام محصولات در نظر گرفت. در این مسئله عملیات‌ها به بلوک‌هایی تقسیم‌بندی می‌شوند. هر بلوک شامل عملیات ماشین‌کاری، آماده‌سازی و مونتاژ برای یک یا چند محصول است. الله‌وردی و الازنی^[۱۲] مسئله‌ی زمان‌بندی مونتاژ دومرحله‌ی با هدف کمینه‌سازی زمان اتمام کارها را مورد بررسی قرار دادند. آن‌ها فرض کردند که زمان‌های آماده‌سازی وابسته به توالی است. آن‌ها یک الگوریتم شاخه و کران و سه روش ابتکاری ارائه کردند. نوایی و همکاران^[۱۳] جریان کارگاهی مونتاژ با ماشین‌های مونتاژ موازی غیریکسان و توالی وابسته به زمان آماده‌سازی را مورد بررسی قرار دادند. آن‌ها چهار الگوریتم فراابتکاری ترکیبی را توسعه دادند و از یک الگوریتم هیوریستیک جدید و الگوریتم شبیه‌سازی تیرید (SA)^۶ برای حل مرحله‌ی مونتاژ استفاده کردند. آزاده و همکاران^[۱۴] مسئله‌ی زمان‌بندی جریان کارگاهی مونتاژ دومرحله‌ی احتمالی همراه با زمان‌های آماده‌سازی با اهداف کمینه‌سازی مجموع وزنی زمان تکمیل آخرین کار و میانگین زمان تکمیل کارها را مورد بررسی قرار دادند.

براساس مرور ادبیات انجام شده در زمینه‌ی سیستم‌های تولید دومرحله‌ی، مسئله‌ی تولید کارگاهی انعطاف‌پذیر با یک مرحله‌ی مونتاژ و با در نظر گرفتن زمان آماده‌سازی برای اولین بار در این مقاله مورد مطالعه قرار می‌گیرد. مسئله‌ی تولید کارگاهی انعطاف‌پذیر توسعه یافته مسئله‌ی تولید کارگاهی است. به‌طور کلی مسئله‌ی تولید کارگاهی NP-hard است.^[۱۵] بنابراین مسئله‌ی مورد مطالعه ضمن برخورداری از تمامی پیچیدگی‌های مسئله‌ی تولید کارگاهی، پیچیدگی‌های جدیدی شامل مسئله‌ی تخصیص ماشین‌ها به عملیات‌ها و مونتاژ را نیز دارد. لذا مسئله‌ی مورد مطالعه قویاً NP-hard است. بنابراین به دست آوردن راه حل بهینه توسط روش‌های دقیق و در ابعاد واقعی برای مسائل با ابعاد متوسط و بزرگ امکان‌پذیر نیست. لذا باید از الگوریتم‌های ابتکاری یا فراابتکاری استفاده شود.

در این مقاله، سه الگوریتم فراابتکاری مبتنی بر الگوریتم بهینه‌سازی ازدحام ذرات (PSO)^۷ پیشنهاد شده است. الگوریتم بهینه‌سازی ازدحام ذرات به آسانی قابل اجراست و دارای حافظه است. در این الگوریتم، اعضای جمعیت با هم ارتباط دارند و از طریق تبادل اطلاعات به حل مسئله می‌پردازند؛ لذا این الگوریتم از سرعت همگرایی بالایی برخوردار است. این الگوریتم به دلیل قابلیت جست‌وجوی سراسری در فضای پیوسته بسیار مورد توجه است. اما این الگوریتم در شمار کمی از مقالات برای حل مسائل زمان‌بندی گسسته استفاده شده است.^[۱۶] همچنین در سال‌های اخیر محققان تلاش بسیاری برای بهبود الگوریتم بهینه‌سازی ازدحام ذرات انجام داده‌اند.^[۱۷-۲۱] به دلیل توانایی‌های الگوریتم بهینه‌سازی ازدحام ذرات و نتایج مطلوبی که از خود نشان داده است، سه الگوریتم فراابتکاری مبتنی بر این الگوریتم پیشنهاد شده است.

۳. تعریف مسئله و مدل ریاضی

مسئله‌ی مورد بررسی در این مقاله مدل‌سازی و تعیین توالی انجام کارها در یک سیستم تولید ترکیبی دومرحله‌ی است. مرحله‌ی اول، مسئله‌ی تولید کارگاهی انعطاف‌پذیر است که به‌صورت زمان‌بندی n قطعه از p محصول روی m ماشین تعریف می‌شود. هر قطعه دارای تعدادی عملیات است و برای هر عملیات، مجموعه‌ی از ماشین‌ها وجود دارد که توانایی انجام آن عملیات را دارند. بنابراین برای هر عملیات امکان انتخاب یک



شکل ۱. شمای مسئله‌ی تولید کارگاهی انعطاف‌پذیر با یک مرحله مونتاژ.

ماشین i از بین چند ماشین اختصاص داده شده وجود دارد و مسئله به دو زیرمسئله‌ی تخصیص و توالی تقسیم می‌شود. در این حالت هر قطعه دارای h_{jz} عملیات است که باید به ترتیب انجام شود. اندیس z نشان دهنده‌ی قطعه، اندیس h نشان دهنده‌ی عملیات و اندیس i نشان دهنده‌ی ماشین است. همچنین عملیات h از قطعه j از محصول p با نماد $O_{j,p,h}$ ($j = 1, \dots, n, h = 1, \dots, h_j, p = 1, \dots, P$) نشان داده می‌شود. انجام هر عملیات $O_{j,p,h}$ توسط یک ماشین i از مجموعه ماشین‌های اختصاص یافته به آن عملیات نیاز به زمان پردازش $Pt_{i,j,p,h}$ دارد. فرض شده است که عملیات $O_{j',p',h'}$ بلافاصله بعد از عملیات $O_{j,p,h}$ انجام می‌شود. بنابراین زمان مورد نیاز برای راه‌اندازی عملیات $O_{j',p',h'}$ ، $O_{j',p',h'}$ ، $O_{j',p',h'}$ ، $O_{j',p',h'}$ در نظر گرفته شده است. در مدل ارائه شده فرض بر آن است که زمان آماده‌سازی وابسته به توالی، مستقل از شماره عملیات‌های قطعات است ($s_{i,j,p,h,j',p',h'} = s_{i,j',p',h',j,p,h}$). مرحله‌ی دوم، مسئله‌ی زمان‌بندی مونتاژ است که به صورت زمان‌بندی p محصول با زمان مونتاژ A_p ، تعریف می‌شود. شمای کلی این مسئله در شکل ۱ نمایش داده شده است.

برای روشن شدن مسئله، یک مثال کاربردی از آن ارائه شده است. فرض کنید یک شرکت لوازم خانگی، دو محصول جاروبرقی و ماشین لباسشویی تولید می‌کند. به طور کلی، تولید بدنه و موتور دو زیرفرآیند اصلی هر دو محصول محسوب می‌شود. تولید محصولات مورد نظر شامل دو مرحله است: ۱. پردازش، ۲. تولید بدنه و موتور. که در یک سیستم تولید کارگاهی انعطاف‌پذیر صورت می‌گیرد. این سیستم تولیدی شامل چندین عملیات مانند برش، شکل‌دهی، رنگ‌آمیزی و... است. در این مثال زمان راه‌اندازی ماشین، به عملیات پیشین آن بستگی دارد. مثلاً زمان آماده‌سازی عملیاتی که از مرحله‌ی شکل‌دهی به رنگ آمیزی می‌رود با عملیاتی که از مرحله‌ی برش به رنگ آمیزی می‌رود متفاوت است. در هر دو مورد اپراتور زمانی را صرف تنظیم قطعه می‌کند. به علاوه، در عملیات شکل‌دهی مدت زمانی طول می‌کشد تا بدنه سرد شود؛ ولی در عملیات برش نیازی به این زمان نیست. در مرحله‌ی دوم، پس از تکمیل قطعات مورد نیاز هر محصول، قطعات به همراه سایر تجهیزات و متعلقات با یکدیگر مونتاژ می‌شوند.

فرضیاتی که در مراحل مدل‌سازی در نظر گرفته شده‌اند، عبارت‌اند از:

۱. همه‌ی ماشین‌ها در افق برنامه‌ریزی در دسترس‌اند.
۲. تقاضای محصولات نهایی مشخص و همه‌ی قطعات در زمان صفر آماده‌ی پردازش‌اند.
۳. شکست در عملیات مجاز نیست؛ یعنی هر نوع عملیات شروع شده، باید بدون وقفه به اتمام برسد.
۴. تعدادی محصول باید تولید شود و هر محصول برای تکمیل شدن نیازمند یک سری قطعات است.

۵. زمان پردازش قطعات و مونتاژ محصولات قطعی و معین است.

۶. وقتی که تولید همه‌ی اجزاء یک محصول به پایان رسید، وارد مرحله‌ی مونتاژ می‌شوند.

۷. ماشین‌ها نمی‌توانند در یک زمان بیش از یک عملیات انجام دهند و قطع کار مجاز نیست.

۸. در هر زمان هر قطعه فقط روی یک ماشین پردازش می‌شود.

۹. برای مشخص کردن شروع یک کار روی هر ماشین، یک کار مصنوعی صفر در نظر گرفته شده است. عملیاتی که به دنبال یک کار مصنوعی زمان‌بندی می‌شوند، اولین عملیات برای پردازش است.

۱.۳. پارامترهای مدل

P, P' : تعداد محصولات ($p = 0, 1, \dots, P, p' = 0, 1, \dots, P'$)

$n_p, n_{p'}$: تعداد زیر قطعات محصول p, p'

($j = 0, 1, 2, \dots, n_p, j' = 0, 1, 2, \dots, n_{p'}$)

$h_j, h'_{j'}$: تعداد عملیات‌های قطعه‌ی j, j'

($h = 1, 2, \dots, h_j, h' = 1, 2, \dots, h'_{j'}$)

m : تعداد کل ماشین‌ها در مرحله‌ی تولید کارگاهی انعطاف‌پذیر ($i = 1, 2, \dots, m$)

K'_A : تعداد محصولات اختصاص یافته به ماشین مونتاژ در مرحله‌ی دوم ($k' = 1, 2, \dots, K'_A$)

$O_{j,p,h}$: عملیات h از قطعه j از محصول p

$Pt_{i,j,p,h}$: زمان پردازش عملیات $O_{j,p,h}$ روی ماشین i

A_p : زمان مونتاژ محصول p

$a_{i,j,p,h}$: عملیات تخصیص یافته به ماشین i ، (اگر عملیات $O_{j,p,h}$ روی ماشین i انجام شود برابر ۱ و در غیر این صورت صفر است)

$s_{i,j,p,j',p'}$: زمان مورد نیاز برای راه‌اندازی عملیات $O_{j',p',h'}$ ، در صورتی که

عملیات $O_{j,p,h}$ بلافاصله بعد از عملیات $O_{j,p,h}$ انجام شود؛

L : یک عدد بزرگ.

۲.۳. متغیرهای مدل

C_{max} : زمان حداکثر دوره ساخت؛

$t_{j,p,h}$: زمان شروع عملیات $O_{j,p,h}$

$f_{j,p,h}$: زمان پایان عملیات $O_{j,p,h}$

St_p : زمان شروع عملیات مونتاژ محصول p ؛

$Sm_{k'}$: زمان شروع به کار ماشین مونتاژ در مرحله‌ی دوم در نوبت k' ؛

E_p : زمان ساخت محصول p در مرحله‌ی اول؛

C_p : زمان تکمیل محصول p ؛

$y_{i,j,p,h}$: یک متغیر صفر و ۱ است، به طوری که اگر ماشین i برای عملیات $O_{j,p,h}$ انتخاب شود برابر ۱ و در غیر این صورت ۰ است؛

$x_{i,j,p,h,j',p',h'}$: یک متغیر صفر و ۱ است، به طوری که اگر عملیات $O_{j',p',h'}$ روی ماشین i بلافاصله بعد از عملیات $O_{j,p,h}$ انجام شود برابر ۱ و در غیر این صورت صفر است؛

$Z_{p,k'}$: یک متغیر صفر و ۱ است به طوری که اگر محصول p روی ماشین مونتاژ در نوبت k' مونتاژ شود برابر ۱ و در غیر این صورت صفر است.

$$\sum_{k'=1}^{k'_A} Z_{p,k'} = 1, \quad p = 1, 2, \dots, P \quad (17)$$

$$x_{i,j,p,h,j',p',h'} = 0, \quad i = 1, 2, \dots, m, \quad j = 0, 1, \dots, n_p, \quad p = 0, 1, \dots, P, \quad h = 1, 2, \dots, h_j \quad (18)$$

$$x_{i,j,p,h,j',p',h'} \in \{0, 1\}, \quad i = 1, 2, \dots, m, \quad j = 0, 1, \dots, n_p, \quad p = 0, 1, \dots, P, \quad h = 1, 2, \dots, h_j, \quad j' = 1, \dots, n_{p'}, \quad p' = 1, \dots, P', \quad h' = 1, 2, \dots, h'_{j'} \quad (19)$$

$$y_{i,j,p,h} \in \{0, 1\}, \quad i = 1, 2, \dots, m, \quad j = 0, 1, \dots, n_p, \quad p = 0, 1, \dots, P, \quad h = 1, 2, \dots, h_j \quad (20)$$

$$Z_{p,k'} \in \{0, 1\}, \quad P = 1, 2, \dots, P, \quad k' = 1, 2, \dots, k'_A \quad (21)$$

$$C_p, St_p, E_p \geq 0, \quad P = 1, 2, \dots, P \quad (22)$$

$$t_{j,p,h}, f_{j,p,h} \geq 0, \quad j = 0, 1, \dots, n_p, \quad p = 0, 1, \dots, P, \quad h = 1, 2, \dots, h_j \quad (23)$$

$$Sm_{k'} \geq 0, \quad k' = 1, 2, \dots, k'_A \quad (24)$$

مفاهیم معادلات و محدودیت‌های این مدل به شرح زیر است: معادله ۱ نشان‌گر تابع هدف مدل است که زمان تکمیل محصولات را کمینه می‌کند. محدودیت ۲ مقدار تابع هدف را تعریف می‌کند. محدودیت‌های ۳ و ۴ توالی عملیات هر کار را مشخص می‌کنند. محدودیت ۵ بزرگ‌ترین زمان پردازش قطعات یک محصول را نشان می‌دهد. محدودیت ۶ موجب می‌شود که ماشین مورد نیاز برای هر عملیات از بین ماشین‌های تخصیص یافته به آن عملیات انتخاب شود. محدودیت ۷ موجب می‌شود که برای هر عملیات از بین ماشین‌های تخصیص یافته به آن عملیات، تنها یک ماشین انتخاب شود. محدودیت‌های ۸ و ۹ بیان می‌کنند هر ماشین در یک زمان فقط یک عملیات را، با در نظر گرفتن زمان آماده‌سازی پردازش می‌کند. محدودیت‌های ۱۰ و ۱۱ جایگشت‌های دایره‌یی (توالی) عملیات‌ها روی هر ماشین را تعریف می‌کنند. ۱ شدن مقدار سمت راست محدودیت ۱۰ بیان‌گر آن است که عملیات h ام از قطعه‌ی j از محصول p روی ماشین i پردازش می‌شود. محدودیت ۱۰ نیز عملیات قبل و بعد از عملیات h ام از قطعه‌ی j از محصول p را مشخص می‌کند. بنابراین ترتیب توالی توسط محدودیت‌های ۱۰ و ۱۱ تعیین می‌شود. محدودیت ۱۲ کم‌ترین زمان شروع مرحله‌ی مونتاژ را نشان می‌دهد. محدودیت ۱۳ بیان‌گر زمان تکمیل محصولات است. محدودیت ۱۴ این الزام را به وجود می‌آورد که در مرحله‌ی مونتاژ یک ماشین در صورتی می‌تواند به کار در نوبت $k' + 1$ پردازد که کار آن ماشین در نوبت k' به اتمام رسیده باشد. محدودیت‌های ۱۵ و ۱۶ این الزام را به وجود می‌آورند که در مرحله‌ی دوم پردازش یک محصول در صورتی می‌تواند شروع شود که هم محصول قبلی آن به اتمام رسیده باشد و هم ماشین بیکار باشد. محدودیت ۱۷ موجب می‌شود که به ماشین مونتاژ در هر نوبت، تنها یک محصول اختصاص یابد.

۴. الگوریتم‌های پیشنهادی

با توجه به پیچیدگی مسئله‌ی مورد بررسی، الگوریتم‌های جست‌وجوی دقیق برای حل آن نیازمند زمان محاسباتی زیادی هستند و این زمان با بزرگ شدن ابعاد مسئله به صورت نمایی افزایش می‌یابد و در برخی موارد نیز یافتن جواب بهینه عملاً امکان‌پذیر نیست. لذا الگوریتم‌های فراابتکاری که در پی به دست آوردن جواب با

با توجه به توضیحات فوق و با در نظر گرفتن پارامترها و متغیرهای تعریف شده، مدل ریاضی مسئله به صورت زیر تعریف می‌شود. مدل برنامه‌ریزی خطی عدد صحیح مختلط ارائه شده توسعه‌یافته‌ی مدل ریاضی سعیدی مهرآباد و فتاحی^[۲۲] است.

$$\text{Min } Z = (C_{\max}) \quad (1)$$

Subject to:

$$C_{\max} \geq C_p, \quad \forall p \quad (2)$$

$$t_{j,p,h} + y_{i,j,p,h} \cdot P t_{i,j,p,h} \leq f_{j,p,h}, \quad i = 1, 2, \dots, m, \quad j = 1, 2, \dots, n_p, \quad p = 1, 2, \dots, P, \quad h = 1, 2, \dots, h_j \quad (3)$$

$$f_{j,p,h} \leq t_{j,p,h+1}, \quad j = 1, 2, \dots, n_p, \quad p = 1, 2, \dots, P, \quad h = 1, 2, \dots, h_j - 1 \quad (4)$$

$$f_{j,p,h} \leq E_p, \quad j = 1, 2, \dots, n_p, \quad p = 1, 2, \dots, P, \quad h = 1, 2, \dots, h_j \quad (5)$$

$$y_{i,j,p,h} \leq a_{i,j,p,h}, \quad i = 1, 2, \dots, m, \quad j = 0, 1, \dots, n_p, \quad p = 0, 1, \dots, P, \quad h = 1, 2, \dots, h_j \quad (6)$$

$$\sum_{i=1}^m y_{i,j,p,h} = 1, \quad j = 0, 1, 2, \dots, n_p, \quad p = 0, 1, \dots, P, \quad h = 1, 2, \dots, h_j \quad (7)$$

$$t_{j,p,h} + P t_{i,j,p,h} + s_{i,j,p,j',p'} \leq t_{j',p',h'} + (\lambda - x_{i,j,p,h,j',p',h'}) \cdot L, \quad i = 1, 2, \dots, m, \quad j = 0, 1, \dots, n_p, \quad p = 0, 1, \dots, P, \quad h = 1, 2, \dots, h_j, \quad j' = 1, \dots, n_{p'}, \quad p' = 1, \dots, P', \quad h' = 1, 2, \dots, h'_{j'} \quad (8)$$

$$f_{j,p,h} + s_{i,j,p,j',p'} \leq t_{j,p,h+1} + (\lambda - x_{i,j',p',h',j,p,h+1}) \cdot L, \quad i = 1, 2, \dots, m, \quad j = 1, \dots, n_p, \quad p = 1, \dots, P, \quad h = 1, 2, \dots, h_j - 1, \quad j' = 0, 1, \dots, n_{p'}, \quad p' = 0, 1, \dots, P', \quad h' = 1, 2, \dots, h'_{j'} \quad (9)$$

$$\sum_{p=0}^P \sum_{j=0}^{n_p} \sum_{h=1}^{h_j} x_{i,j,p,h,j',p',h'} = y_{i,j',p',h'}, \quad i = 1, 2, \dots, m, \quad j' = 1, \dots, n_{p'}, \quad p' = 1, \dots, P', \quad h' = 1, \dots, h'_{j'} \quad (10)$$

$$\sum_{p'=1}^{P'} \sum_{j'=1}^{n_{p'}} \sum_{h'=1}^{h'_{j'}} x_{i,j,p,h,j',p',h'} = y_{i,j,p,h}, \quad i = 1, \dots, m, \quad j = 0, 1, \dots, n_p, \quad p = 0, 1, \dots, P, \quad h = 1, \dots, h_j \quad (11)$$

$$E_p \leq St_p, \quad \forall p \quad (12)$$

$$A_p + St_p \leq C_p, \quad \forall p \quad (13)$$

$$Sm_{k'} + A_p \cdot Z_{p,k'} \leq Sm_{k'+1}, \quad \forall p, \quad k' = 1, 2, \dots, k'_A - 1 \quad (14)$$

$$Sm_{k'} \leq St_p + (\lambda - Z_{p,k'}) \cdot L, \quad \forall p, k' \quad (15)$$

$$Sm_{k'} + (\lambda - Z_{p,k'}) \cdot L \geq St_p, \quad \forall p, k' \quad (16)$$

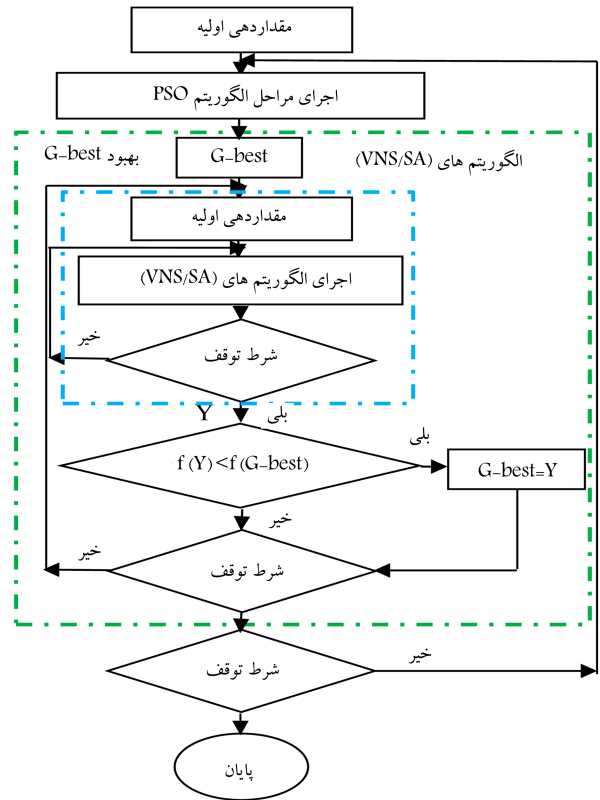
که در آن، $V_i(t)$ سرعت ذره i ام در مرحله t جدید (t) ، $X_i(t)$ موقعیت ذره i ام در مرحله t جدید (t) ، $P\text{-Best}_i(t)$ بهترین موقعیت فردی ذره i ام، $G\text{-Best}$ بهترین موقعیت جمعی ذرات، w وزن اینرسی، r_1 و r_2 دو عدد تصادفی که دارای توزیع یکنواخت در بازه $(0, 1)$ هستند. c_1 و c_2 به ترتیب ضریب یادگیری فردی و ضریب یادگیری اجتماعی هستند که به ترتیب باعث می‌شوند ذره به سمت بهترین موقعیت فردی و بهترین موقعیت جمعی حرکت کند. ساختار کلی الگوریتم پیشنهادی همانند الگوریتم بهینه‌سازی ازدحام ذرات است. در این الگوریتم پس از تولید جمعیت اولیه، برای هر کدام از ذرات یک سرعت اولیه در نظر گرفته می‌شود و بنابر فرمول عنوان شده به جست‌وجو در فضا می‌پردازد.

۲.۴. الگوریتم ترکیبی بهینه‌سازی ازدحام ذرات و جست‌وجوی همسایگی متغیر (HPSOVNS)^{۱۰}

الگوریتم بهینه‌سازی ازدحام ذرات از سرعت همگرایی بالایی برخوردار است اما در نزدیکی نقطه بهینه‌ی فرایند جست‌وجو کند می‌شود و بقیه‌ی فضا را مورد کاوش قرار نمی‌دهد. لذا به منظور جلوگیری از افتادن در بهینه‌ی محلی و همگرایی زودرس، الگوریتم جست‌وجوی همسایگی متغیر^{۱۱} پیشنهاد شده است. این الگوریتم روشی مناسب برای بهبود فرایند جست‌وجو است که می‌تواند از بیشترین اطلاعات مربوط به مسئله برای بهبود کیفیت راه‌حل‌ها استفاده کند.

الگوریتم جست‌وجوی همسایگی متغیر توسط ملادنیچ و هانس در سال ۱۹۹۷ مطرح شد.^[۲۴] این الگوریتم یکی از الگوریتم‌های فرایندکاری بر پایه‌ی تغییرات سیستماتیک در طول فرایند جست‌وجو است.^[۲۵،۲۴] تغییرات سیستماتیک ساختار همسایگی به منظور فرار از نقطه بهینه محلی انجام می‌گیرد. این الگوریتم با یک راه‌حل اولیه کار خود را آغاز می‌کند و تا زمانی که معیار خاتمه (حداکثر تعداد تکرار) برقرار شود، حلقه‌ی اصلی الگوریتم تکرار می‌شود. حلقه‌ی اصلی الگوریتم شامل دو فاز آشفستگی و جست‌وجوی محلی است. در فاز آشفستگی الگوریتم از جواب فعلی (y) با استفاده از ساختار همسایگی به جواب همسایه (y') حرکت می‌کند. در فاز جست‌وجوی محلی نیز بر روی جواب y' با استفاده از روش‌های جست‌وجوی محلی، جست‌وجو انجام می‌گیرد و بهینه‌ی محلی y'' به دست می‌آید. سپس، بهینه‌ی محلی y'' با راه حل فعلی y از نظر کیفیت راه حل مقایسه می‌شود. اگر y'' بهتر از y باشد، یعنی در صورتی که بهبودی در راه حل ایجاد شود، y'' جایگزین y می‌شود و جست‌وجو دوباره با ساختار همسایگی اول با y به‌روز شده آغاز می‌شود. در غیر این صورت، حلقه‌ی جست‌وجو توسط ساختار همسایگی بعدی تکرار می‌شود. در این مورد، ساختار محلی به صورت سیستماتیک تغییر می‌کند و روش آشفستگی برای کاوش یک منطقه‌ی دیگر از فضا و انجام جست‌وجوی محلی جدید صورت می‌گیرد. پس از این که همه‌ی ساختارهای همسایگی بررسی شد و هیچ بهبودی برای راه‌حل y به دست نیامد، تکرار بعدی الگوریتم آغاز می‌شود.

در الگوریتم ترکیبی پیشنهادی از مزایای هر دو الگوریتم بهینه‌سازی ازدحام ذرات و جست‌وجوی همسایگی متغیر استفاده می‌شود. در این روش ترکیبی در ابتدا، بهترین راه‌حل سراسری توسط الگوریتم بهینه‌سازی ازدحام ذرات به دست می‌آید. سپس، الگوریتم جست‌وجوی همسایگی متغیر بر روی بهترین راه‌حل سراسری پیدا شده در هر تکرار، به منظور بهبود بهترین راه‌حل سراسری، اعمال می‌شود. فلوجارت الگوریتم ترکیبی پیشنهادی در شکل ۲ نشان داده شده است.



شکل ۲. فلوجارت الگوریتم‌های ترکیبی پیشنهادی.

کیفیت در زمان معقول باشند، در حل این مسئله کاربرد بسیاری دارد. به همین منظور در این نوشتار، الگوریتم بهینه‌سازی ازدحام ذرات و دو الگوریتم فرایندکاری ترکیبی پیشنهاد شده است. نحوه‌ی عملکرد الگوریتم‌های ترکیبی پیشنهادی در شکل ۲ نشان داده شده است.

۱.۴. الگوریتم بهینه‌سازی ازدحام ذرات

الگوریتم بهینه‌سازی ازدحام ذرات یکی از مهم‌ترین الگوریتم‌هایی است که در حوزه‌ی هوش جمعی جای می‌گیرد. این الگوریتم برای اولین بار توسط جیمز کندی و راسل سی ابرهارت در سال ۱۹۹۵ مطرح و با الهام از رفتار اجتماعی حیواناتی چون ماهی‌ها و پرندگان که در گروه‌های کوچک و بزرگ کنار هم زندگی می‌کنند، طراحی شده است.^[۲۳] در الگوریتم بهینه‌سازی ازدحام ذرات هر جواب مسئله، موقعیت یک ذره در فضای جست‌وجو را نشان می‌دهد. تغییر مکان هر ذره در فضای جست‌وجو تحت تأثیر بهترین جوابی است که تاکنون برای هر ذره به‌طور جداگانه به دست آمده است، و آن را «بهترین موقعیت فردی»^۸ می‌نامند؛ و آن بهترین مقداری است که تاکنون تحت عنوان «بهترین موقعیت جمعی»^۹ و توسط تمام ذره‌ها در میان جمعیت به دست آمده است. معادلات مربوط به موقعیت و سرعت ذرات چنین تعریف می‌شوند:

$$V_i(t) = W \times V_i(t-1) + c_1 \times r_1 \times (P - \text{Best}_i - X_i(t-1)) + c_2 \times r_2 \times (G - \text{Best} - X_i(t-1)) \quad (25)$$

$$X_i(t) = X_i(t-1) + V_i(t) \quad (26)$$

۱.۲.۴. ساختار کلی الگوریتم‌های پیشنهادی

الف) نمایش راه حل

هر عضو از جمعیت دارای دو بخش است: ۱. تخصیص ماشین‌ها به عملیات و توالی عملیات روی ماشین‌ها در زیرمسئله‌ی تولید کارگاهی انعطاف‌پذیر؛ ۲. توالی محصولات روی ماشین مونتاژ. در بخش اول موقعیت هریک از ذرات توسط دو بردار نشان داده شده است: ۱. بردار تخصیص ماشین؛ ۲. بردار توالی عملیات. بردار توالی عملیات شامل دو بردار است. در بردار اول، قطعه، از نمایش جن و همکاران^[۲۶] استفاده شده است. هر عنصر از بردار، یک عملیات از قطعه ز را نشان می‌دهد. در این نحوه‌ی نمایش تمام عملیات یک قطعه با نماد یکسان نمایش داده می‌شوند. بردار دوم نمایان‌گر شماره‌ی محصول قطعات است. در بردار تخصیص، هر عدد نشان‌دهنده‌ی ماشین انتخاب شده برای عملیات‌های متناظر در بردار توالی است. در این نحوه‌ی نمایش طول رشته‌ی بردار توالی و تخصیص برابر با تعداد کل عملیات است. در بخش دوم، توالی محصولات توسط یک بردار نمایش داده شده است. هر عدد در این بردار نشان‌دهنده‌ی شماره‌ی محصول است. این نحوه‌ی نمایش شدنی بودن راه‌حل‌های به‌روزرسانی شده را به‌صورت خودکار انجام می‌دهد. شکل ۳ یک راه‌حل شدنی برای مسئله‌ی زمان‌بندی تولید کارگاهی انعطاف‌پذیر با یک مرحله‌ی مونتاژ با ۲ محصول و ۳ ماشین را نشان می‌دهد. در این مثال محصول ۱ و ۲ به ترتیب ۳ و ۲ قطعه دارند. قطعات ۱، ۲، ۳ از محصول ۱ به ترتیب دارای ۱، ۱، ۲ عملیات و قطعات ۱ و ۲ از محصول ۲ به ترتیب دارای ۱ و ۲ عملیات هستند. به‌عنوان مثال قطعه‌ی ۲ از محصول ۲ دارای ۲ عملیات است که به ترتیب در خانه‌های ۱ و ۴ قرار گرفته‌اند.

ب) مقداردهی اولیه

جمعیت اولیه یکی از ویژگی‌های الگوریتم‌های تکاملی است که بر سرعت همگرایی و کیفیت راه‌حل نهایی تأثیر می‌گذارد. در مسئله‌ی مورد مطالعه، تولید جمعیت اولیه در مرحله‌ی اول دارای دو بخش است: ۱. تخصیص ماشین‌ها به عملیات؛ ۲. توالی عملیات. برای تولید تخصیص‌های اولیه از دو قاعده‌ی پزلا و همکاران^[۲۷] استفاده شده است: جست‌وجو برای کمینه‌ی سراسری در جدول زمان پردازش؛ و تغییر دادن مکان کارها و ماشین‌ها به‌صورت تصادفی در جدول زمان پردازش. ترکیبی از این دو قانون مجموعه‌ی اولیه از تخصیص‌ها را تولید می‌کند. هر دو قاعده نسخه‌های اصلاح شده‌ی رویکرد محلی‌سازی پیشنهادی توسط کاکم و همکاران^[۲۸] هستند. در این مقاله ۲۰ درصد از تخصیص‌های اولیه توسط قاعده‌ی اول و ۸۰ درصد توسط قاعده‌ی دوم تولید می‌شوند. در این نوشتار، توالی اولیه‌ی عملیات در مرحله‌ی

توالی عملیات	قطعه	۲	۳	۱	۲	۱	۲	۳
	محصول	۲	۱	۲	۲	۱	۱	۱
تخصیص ماشین		۱	۳	۳	۲	۱	۳	۲

(الف)

توالی محصول	۲	۱
-------------	---	---

(ب)

شکل ۳. نحوه‌ی نمایش راه حل.

اول و توالی محصولات در مرحله‌ی دوم از قاعده‌ی تصادفی به دست می‌آید. در این قاعده به‌طور تصادفی نماد اعداد مرتب می‌شوند.

ج) ساختار همسایگی

روش انتقال از یک راه‌حل به راه‌حل همسایه آن توسط یک عامل کلیدی شناخته شده تحت عنوان ساختار همسایگی صورت می‌گیرد. در این بخش پنج نوع از ساختارهای همسایگی به‌کاررفته در الگوریتم جست‌وجوی همسایگی متغیر معرفی می‌شوند.^[۲۹]

- **ساختار همسایگی ۱.** توالی عملیات و محصولات در رشته مربوط به راه‌حل کاندید را تغییر می‌دهد. در این ساختار یک سلول به‌صورت تصادفی انتخاب می‌شود، سپس عملیات/محصول از سلول انتخاب شده با عملیات/محصول از سلول قبلی جایگزین می‌شود.
- **ساختار همسایگی ۲.** در این ساختار ابتدا یک عملیات به‌صورت تصادفی که قادر به پردازش روی بیش از یک ماشین است، انتخاب می‌شود. سپس یک ماشین جدید به‌طور تصادفی انتخاب شده و این عملیات به آن تخصیص داده می‌شود.
- **ساختار همسایگی ۳.** ساختار ۱ و ۲ به‌طور هم‌زمان روی راه‌حل کاندید اجرا می‌شود.
- **ساختار همسایگی ۴.** در این ساختار ابتدا ماشین‌هایی با بیش‌ترین و کم‌ترین بار کاری از رشته راه‌حل کاندید انتخاب می‌شود. سپس در صورت امکان یک عملیات به‌طور تصادفی از ماشین با بیش‌ترین بار کاری انتخاب می‌شود و به ماشین با کم‌ترین بار کاری تخصیص داده می‌شود. بار کاری ماشین برابر است با مجموع زمان‌های پردازش عملیات‌های تخصیص یافته به ماشین.
- **ساختار همسایگی ۵.** در این ساختار ابتدا دو قطعه از دو یا یک محصول به‌صورت تصادفی انتخاب می‌شوند. اگر تعداد عملیات از دو قطعه انتخاب شده برابر باشند، مکان عملیات‌های دو قطعه در رشته راه‌حل کاندید به‌صورت یک به یک، با توجه به شماره عملیات‌ها عوض می‌شوند. در مقابل، اگر تعداد عملیات‌ها در دو قطعه برابر نباشند، عملیات‌های مربوط به قطعه با تعداد عملیات کم‌تر، با رعایت محدودیت‌های توالی به سلول‌هایی از قطعه با تعداد عملیات بیشتر منتقل می‌شوند؛ و عملیات‌های مربوط به قطعه با تعداد عملیات بیشتر، سلول‌های باقی‌مانده خود و تمام سلول‌های قبلی از قطعه با تعداد عملیات کم‌تر را با رعایت محدودیت‌های توالی پر می‌کنند. این ساختار همسایگی با ایجاد تغییر نسبتاً بزرگ در راه‌حل اولیه، خطر گرفتار شدن در بهینه محلی را کاهش می‌دهد. یک مثال از نحوه اجرا شدن این ساختار همسایگی در شکل ۴ ارائه شده است. در این مثال قطعه‌ی ۱ از محصول ۳ و قطعه‌ی ۳ از محصول ۲ برای جایگزینی انتخاب شده است.

د) **انطباق الگوریتم بهینه‌سازی ازدحام ذرات با جواب مسئله (فضای گسسته)**
جواب مسئله‌ی تولید کارگاهی انعطاف‌پذیر با یک مرحله‌ی مونتاژ به‌صورت توالی از اعداد گسسته است درحالی‌که الگوریتم بهینه‌سازی ازدحام ذرات برای حل مسائل با فضای پیوسته طراحی شده است. اما با توجه به کارایی بالای این الگوریتم، تاکنون مسائل گسسته‌ی زیادی با آن حل شده است. در الگوریتم پیشنهادی بعد از هر بار به‌روزرسانی موقعیت ذرات، هریک از درایه‌های بردار به نزدیک‌ترین عدد صحیح تبدیل می‌شود. بردار به دست آمده در این مرحله جواب نشدنی از مسئله را نشان

تابع هدف آن محاسبه می‌شود. در این گام دمای اولیه (T_i) به‌عنوان دمای جاری (T_c) در نظر گرفته می‌شود. در گام دوم، در هر دما، یک زنجیره مارکف ایجاد می‌شود. بدین منظور، یک راه‌حل همسایه در همسایگی راه‌حل جاری پیدا می‌شود. در صورتی که همسایه‌ی جدید مقدار تابع را بهبود بدهد یا برابر آن باشد، الگوریتم از یک نقطه به نقطه‌ی دیگر حرکت می‌کند. در غیر این صورت، همسایه با تابع احتمال $e^{-\frac{\Delta F}{T_c}}$ ارزیابی می‌شود. ΔF ، اختلاف بین مقادیر تابع هدف جواب راه‌حل جاری و همسایه جدید است. برای این منظور مقدار تصادفی $p \sim U(0, 1)$ تولید شده و سپس با تابع احتمال مقایسه می‌شود. در صورتی که مقدار p کم‌تر از تابع احتمال باشد، راه‌حل همسایه به‌عنوان راه‌حل جاری پذیرفته می‌شود. علاوه بر به‌هنگام‌سازی راه‌حل جاری، بهترین راه‌حل نیز در این گام به‌نگام می‌شود. بدین منظور، هر راه‌حل جدید که منجر به بهبود راه‌حل جاری شود، آن راه‌حل با بهترین راه‌حل مقایسه می‌شود. اگر این راه‌حل از بهترین راه‌حلی که تا به حال پیدا شده است، بهتر باشد، جایگزین راه‌حل بهتر می‌شود. این گام تا زمانی ادامه می‌یابد که زنجیره‌ی مارکوف به انتها برسد. برای زنجیره‌ی مارکوف یک طول معین مشخص شده است. زمانی که تعداد جست‌وجو (تمامی نقاط پذیرفته شده و رد شده) به طول مشخص برسد، زنجیره پایان می‌یابد. در طول زنجیره، دما ثابت می‌ماند. در گام سوم، بعد از اتمام یک زنجیره، کاهش دما اتفاق خواهد افتاد. این کاهش موجب می‌شود تعداد کم‌تری از جواب‌های نامطلوب پذیرفته شود. تعیین قانون و تابع کاهش دما و حرکت به سمت سرد شدن سیستم، نیازمند ضابطه‌ی است که به شکل زیر ارائه شده است:

$$T_i = \alpha (T_{i-1}) \quad \forall i = 1, 2, \dots, n \quad (27)$$

α ضریب تبرید یا ضریب کاهش دماست که ثابتی کم‌تر از ۱ است. در گام چهارم، شرط توقف الگوریتم بررسی می‌شود. در صورتی که شرایط توقف برآورده شود، الگوریتم متوقف و در غیر این صورت به گام دوم برمی‌گردد.

الگوریتم شبیه‌سازی تبرید از توقف در ناحیه‌ی بهینه‌ی محلی اجتناب کرده و توانایی گذر از آن را دارد. امروزه انجماد تدریجی به‌عنوان یک تکنیک قوی جست‌وجوی همسایگی شناخته شده است که توانایی آن در حل مسائل ترکیبی مانند مسیریابی و مسائل نقلیه، زمان‌بندی تولید و جدول زمانی اثبات شده است.^[۲۲] به‌منظور بهبود فرایند جست‌وجوی الگوریتم بهینه‌سازی ازدحام ذرات، الگوریتم شبیه‌سازی تبرید پیشنهاد شده است. در هر تکرار از الگوریتم ترکیبی پیشنهادی، بهترین موقعیت جمعی به دست آمده توسط الگوریتم شبیه‌سازی تبرید بهبود داده می‌شود. در این الگوریتم از ساختارهای همسایگی ۲ و ۴ که در بخش قبل ارائه شد، استفاده شده است. سایر جزئیات این الگوریتم از جمله مقاردهای اولیه، نحوه‌ی نمایش راه‌حل‌ها و سایر موارد، مشابه الگوریتم پیشنهادی PSO و HPSOVNS است.

۵. نتایج محاسباتی

در این قسمت نتایج محاسباتی به دست آمده از اجرای مدل ریاضی و الگوریتم‌های پیشنهادی مورد بررسی و مقایسه قرار می‌گیرد. برای این هدف، مدل ریاضی و الگوریتم‌های پیشنهادی، به ترتیب در نرم‌افزارهای GAMS و (MAT-R2011a) LAB کدنویسی شده است. با توجه به این‌که اطلاعات مربوط به مسائل حل شده در مقالات مشابه برای مسئله‌ی مورد نظر وجود ندارد. از نمونه‌های تصادفی، به‌منظور بررسی و اعتبار سنجی مدل ریاضی پیشنهادی و همچنین کیفیت نتایج الگوریتم‌های

توالی	قطعه	۳	۱	۱	۱	۲	۱	۳
	عملیات	۲	۲	۳	۱	۳	۲	۳
تخصیص ماشین		۲	۲	۳	۴	۲	۲	۴

توالی	قطعه	۱	۱	۳	۱	۳	۲	۱
	عملیات	۳	۲	۲	۱	۲	۲	۳
تخصیص ماشین		۳	۲	۲	۴	۱	۲	۴

توالی محصول	۲	۱	۳
-------------	---	---	---

شکل ۴. ساختار همسایگی ۵.

حالت پیوسته	توالی عملیات	۰/۳۹	۲/۱۸	۱/۰۷	۲/۸۶	۱/۴۱	۳/۲۷
	توالی محصول	۲/۰۷	۳/۱۹	۱/۶۸			

حالت گسسته (نشدنی)	توالی عملیات	۰	۲	۱	۳	۱	۳
	توالی محصول	۲	۳	۲			

حالت گسسته (شدنی)	توالی عملیات	۱	۲	۱	۲	۱	۳
	تخصیص ماشین	۱	۲	۳	۳	۲	۲
	توالی محصول	۲	۲	۲	۱	۲	۳

توالی محصول	۲	۳	۱
-------------	---	---	---

شکل ۵. تبدیل جواب پیوسته به گسسته.

می‌دهد. لذا برای تولید یک جواب شدنی، هر یک از درایه‌های بردار که مقدار آن‌ها با شماره قطعات یا تعداد عملیات قطعات برابر نیستند را مشخص و به نزدیک‌ترین عدد صحیح مورد نیاز در بردار تبدیل می‌کنند.^[۶] شکل ۵ موقعیت به‌روزرسانی شده‌ی ذره (پیوسته) و همچنین چگونگی تبدیل آن به یک جواب گسسته را نشان می‌دهد. در شکل ۵ نزدیک‌ترین عدد صحیح به عدد ۰/۳۹ عدد صفر است؛ اما چون قطعه‌ی صفر نداریم، عدد ۰/۳۹ به ۱ تبدیل می‌شود. همچنین نزدیک‌ترین عدد صحیح به عدد ۲/۸۶ عدد ۳ است؛ اما چون عدد ۳ دو بار ایجاد شده و قطعه‌ی ۳ نیز فقط یک عملیات دارد، عدد ۲/۸۶ به نزدیک‌ترین عدد صحیح ممکن یعنی ۲ تبدیل می‌شود. در بردار توالی محصول، نزدیک‌ترین عدد صحیح به عدد ۱/۶۸ عدد ۲ است، اما چون هر محصول فقط یک بار مونتاز می‌شود، عدد ۱/۶۸ به نزدیک‌ترین عدد صحیح یعنی ۱، تبدیل می‌شود. در انتها بردار دوم توالی (محصول) و تخصیص ماشین به‌روزرسانی می‌شود.

۳.۴ الگوریتم ترکیبی بهینه‌سازی ازدحام ذرات و شبیه‌سازی تبرید (HPSOSA)^{۱۲}

الگوریتم شبیه‌سازی تبرید، ایده و سازوکار موجود در تبرید فیزیکی یک جسم جامد را شبیه‌سازی می‌کند. این ایده اولین بار در سال ۱۹۵۳ توسط متروپولیس مطرح شد.^[۲۰] این الگوریتم در سال ۱۹۸۳ توسط کریک پاتریک و همکاران توسعه داده شد. آن‌ها الگوریتمی به نام شبیه‌سازی تدریجی برای حل مسائل بهینه‌سازی ترکیبی ارائه کردند.^[۲۱] در این الگوریتم در ابتدا، یک راه‌حل اولیه ایجاد و مقدار

مدت زمان اجرای یکسان به جواب بهینه، مشابه راه‌حل بهینه‌ی به دست آمده توسط گمز دست می‌یابند. با این تفاوت که با افزایش ابعاد مسئله، زمان محاسباتی حاصل از نرم‌افزار گمز بیشتر از الگوریتم‌های فراابتکاری است. برای مسائل با ابعاد متوسط نیز جواب بهینه توسط نرم‌افزار گمز به دست آمد. به دلیل زیاد بودن زمان محاسباتی توسط نرم‌افزار گمز، مسائل با ابعاد بزرگ نیز توسط الگوریتم‌های HPSOVNS, HPSOSA, PSO و HGATS حل شده‌اند. همچنین به منظور بررسی عملکرد الگوریتم‌ها در ابعاد متوسط و بزرگ، از ضریب انحراف معیار استفاده شده است. نتایج به دست آمده برای مسائل با ابعاد متوسط در جدول ۶ نشان داده شده است. نمودار متوسط مقادیر درصد انحراف نسبی با بازه اطمینان ۹۵٪ برای مسائل با ابعاد متوسط در شکل ۶ ترسیم شده است. چنان‌که در جدول ۶ و شکل ۶ نشان داده شده است مقادیر درصد انحراف نسبی و ضریب انحراف معیار الگوریتم HPSOVNS کم‌تر از سه الگوریتم دیگر است. بنابراین، برتری الگوریتم HPSOVNS نسبت به سایر الگوریتم‌ها قابل مشاهده است. به دلیل ماهیت پیچیده‌ی مسئله، مسائل با ابعاد بزرگ فقط توسط الگوریتم‌های HPSOVNS, HPSOSA, PSO و HGATS حل شده است. با توجه به مقادیر درصد انحراف نسبی ارائه شده در جدول ۷، نمودار متوسط مقادیر درصد انحراف نسبی در شکل ۷ ترسیم شده است. با توجه به مقادیر درصد انحراف نسبی و ضریب انحراف معیار در مسائل با ابعاد بزرگ، الگوریتم HPSOVNS نسبت به سایر الگوریتم‌ها عملکرد بهتری دارد. علاوه بر این، مشاهده

پیشنهادی استفاده شده است. نحوه‌ی تولید نمونه‌های تصادفی در جدول ۱ آورده شده است. اعتبارسنجی مدل ریاضی پیشنهادی با استفاده از نرم‌افزار گمز و به‌ازای مسائل کوچک و متوسط انجام شده است. به منظور اعتبارسنجی الگوریتم‌های پیشنهادی، نتایج به دست آمده با الگوریتم ترکیبی ژنتیک و جست‌وجوی ممنوع^[۳۲] لی و گائو^[۳۳] مورد مقایسه قرار گرفته است.

تنظیم مناسب پارامترها و انتخاب درست عملگرها در الگوریتم‌های فراابتکاری تأثیر به‌سزایی بر عملکرد آن‌ها دارد که می‌بایست با توجه به شرایط مسئله‌ی مورد بررسی، تعیین شود. در این مقاله، به منظور تنظیم پارامترها از روش تاگوشی استفاده شده است. بدین منظور نمونه‌ها به سه گروه با اندازه کوچک، متوسط و بزرگ تقسیم شده‌اند و مقادیر مختلف فاکتورهای تأثیرگذار بر عملکرد الگوریتم‌ها مورد آزمایش قرار گرفته است. مقادیر مناسب پارامترهای الگوریتم‌های HPSOVNS, HPSOSA, PSO و به ترتیب در جداول ۲ تا ۴ نشان داده شده است.

به منظور بررسی عملکرد الگوریتم‌ها و مدل ریاضی پیشنهادی، از معیار عملکرد درصد انحراف نسبی، استفاده شده است. این معیار از رابطه‌ی ۲۸ محاسبه می‌شود.

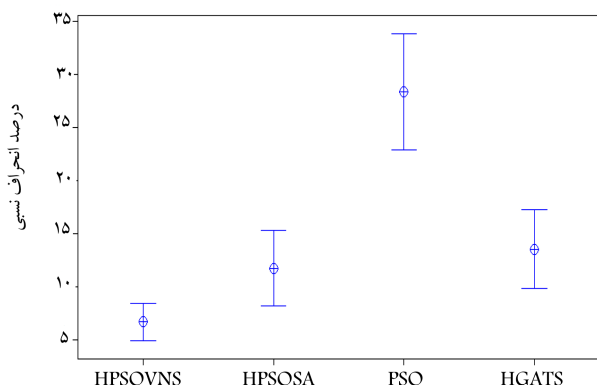
$$RPD = \frac{\text{Algorithm}_{\text{solution}} - \text{Minimum}_{\text{solution}}}{\text{Minimum}_{\text{solution}}} * 100 \quad (28)$$

که در آن $\text{Algorithm}_{\text{solution}}$ تابع هدف حاصل از حل یک نمونه با استفاده از الگوریتم مورد نظر و $\text{Minimum}_{\text{solution}}$ کم‌ترین مقدار تابع هدف حاصل از حل آن نمونه با استفاده از روش‌های حل مورد مقایسه است.

۱.۵. تجزیه و تحلیل

برای مسائل با ابعاد کوچک جواب بهینه توسط نرم‌افزار گمز به دست آمد. نتایج به دست آمده در جدول ۵ نشان داده شده است. با توجه به مقادیر تابع هدف و درصد انحراف نسبی، الگوریتم‌های HPSOVNS, HPSOSA, PSO و HGATS در

نمودار فاصله بی‌الگوریتم‌های HPSOVNS, HPSOSA, PSO, HGATS
فاصله اطمینان ۹۵٪



شکل ۶. نمودار میانگین مقادیر درصد انحراف نسبی با بازه اطمینان ۹۵٪ برای الگوریتم‌های پیشنهادی در مسائل متوسط.

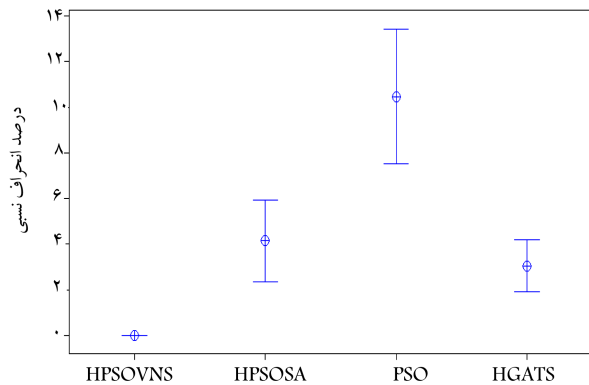
جدول ۱. مقادیر داده‌های تصادفی.

مقادیر	نمادها	پارامترها
{۳, ۲, ..., ۵۰}	P	تعداد محصولات
{۳, ۲, ..., ۱۲}	n	تعداد قطعات
{۳, ۲, ..., ۱۷}	m	تعداد ماشین‌ها در مرحله‌ی اول
عدد صحیح	P_s	زمان پردازش
عدد صحیح	A_p	زمان مونتاژ

جدول ۲. فاکتورها و مقادیر مناسب پارامترهای الگوریتم HPSOVNS.

فراابتکاری	پارامترها	طبقه‌بندی مسائل		
		کوچک	متوسط	بزرگ
PSO	جمعیت اولیه	۴۵	۴۵	۴۵
	ضریب یادگیری فردی	۱٫۵	۰٫۵	۱٫۵
	ضریب یادگیری اجتماعی	۱٫۵	۰٫۵	۰٫۵
	وزن اینرسی	۱	۰٫۸	۱
	بیشینه‌ی تعداد تکرار الگوریتم	۷۰	۷۰	۷۰
VNS	بیشینه‌ی تعداد تکرار الگوریتم جست‌وجوی همسایگی متغیر	۶۰	۵۰	۶۰
	تعداد جست‌وجوی همسایگی در رویه جست‌وجوی محلی	۴۰	۵۰	۴۰
	بیشینه‌ی تعداد تکرار حلقه اصلی	۴	۴	۴

نمودار فاصله بی الگوریتم های HPSOVNS, HPSOSA, PSO, HGATS
فاصله اطمینان ۹۵٪



شکل ۷. نمودار میانگین مقادیر درصد انحراف نسبی با بازه اطمینان ۹۵٪ برای الگوریتم‌های پیشنهادی در مسائل بزرگ.

جواب بهینه، از آزمون آماری استفاده شده است. برای انتخاب آزمون آماری مناسب (پارامتریک یا ناپارامتریک)، در ابتدا فرض‌های بنیادی تحلیل واریانس بررسی می‌شود. آمار پارامتریک مستلزم پیش‌فرض‌هایی است در مورد جامعه‌ی که از آن نمونه‌گیری صورت گرفته است. به عنوان مهم‌ترین پیش‌فرض‌ها، فرض می‌شود که توزیع داده‌ها نرمال باشد و واریانس فاکتورهای مورد بررسی برابر باشند. با توجه به برابری مقادیر تابع هدف الگوریتم‌ها در ابعاد کوچک، تحلیل آماری بر روی مسائل با ابعاد متوسط و بزرگ انجام شده است. برای بررسی نرمال بودن داده‌ها و برابری واریانس الگوریتم‌ها، به ترتیب از آزمون‌های اندرسون - دارلینگ و لون استفاده شده است. این آزمون‌ها در نرم‌افزار Minitab ۱۶ با سطح معناداری ۰/۰۵ اجرا شده و نتایج حاصل از آزمون نرمال بودن الگوریتم‌های پیشنهادی در حل مسائل با ابعاد متوسط و بزرگ، به ترتیب در شکل‌های ۸ و ۹ نشان داده شده است. چنان که مشاهده می‌شود، مقادیر P-value هر چهار الگوریتم در ابعاد متوسط و بزرگ

جدول ۳. فاکتورها و مقادیر مناسب پارامترهای الگوریتم HPSOSA.

فراابتکاری	پارامترها	طبقه‌بندی مسائل		
		کوچک	متوسط	بزرگ
PSO	جمعیت اولیه	۴۵	۴۵	۲۵
	ضریب یادگیری فردی	۱/۵	۰/۵	۰/۵
	ضریب یادگیری اجتماعی	۰/۵	۰/۵	۱/۵
	وزن اینرسی	۰/۸	۰/۸	۱
SA	بیشینه‌ی تعداد تکرار الگوریتم	۵۰	۵۰	۷۰
	ضریب کاهش دما	۰/۹	۰/۶	۰/۶
	دمای اولیه	۱۰۰	۱۰۰	۱۳۰
	تعداد تکرار در هر دما	۳۰	۳۰	۸۰

جدول ۴. فاکتورها و مقادیر مناسب پارامترهای الگوریتم PSO.

فراابتکاری	پارامترها	طبقه‌بندی مسائل		
		کوچک	متوسط	بزرگ
PSO	جمعیت اولیه	۴۵	۴۵	۴۵
	ضریب یادگیری فردی	۱/۵	۰/۵	۱
	ضریب یادگیری اجتماعی	۰/۵	۱/۵	۰/۵
	وزن اینرسی	۱	۰/۸	۰/۸
PSO	حداکثر تعداد تکرار الگوریتم	۱۰۰	۱۰۰	۱۰۰

می‌شود که الگوریتم PSO در هر دو بعد، عملکرد ضعیفی نسبت به سایر الگوریتم‌ها را دارد. نتایج به دست آمده نشان می‌دهد که در کل، الگوریتم HPSOVNS عملکرد بهتری نسبت به دو الگوریتم پیشنهادی و الگوریتم HGATS را دارد. همچنین الگوریتم PSO نسبت به سایر الگوریتم‌ها از کارایی بسیار کم‌تری برخوردار است. بنابراین می‌توان گفت که رویکرد ترکیبی موجب بهبود راه‌حل‌ها می‌شود. در این تحقیق برای مقایسه‌ی توانایی الگوریتم‌های فراابتکاری در پیدا کردن

جدول ۵. نتایج محاسباتی مسائل نمونه در ابعاد کوچک.

نمونه	سایز	زمان محاسباتی (ثانیه)	Gams		HPSOVNS		HPSOSA		PSO		HGATS	
			میانگین	درصد انحراف	میانگین	درصد انحراف	میانگین	درصد انحراف	میانگین	درصد انحراف	میانگین	درصد انحراف
			Cmax	نسبی	Cmax	نسبی	Cmax	نسبی	Cmax	نسبی	Cmax	نسبی
S _۱	۲, ۳, ۲, ۲	۲	۴۰	۰	۴۰	۰	۴۰	۰	۴۰	۰	۴۰	۰
S _۲	۲, ۴, ۴, ۳	۵	۴۱	۰	۴۱	۰	۴۱	۰	۴۱	۰	۴۱	۰
S _۳	۳, ۳, ۶, ۳	۹	۵۰	۰	۵۰	۰	۵۰	۰	۵۰	۰	۵۰	۰
S _۴	۳, ۳, ۶, ۵	۴۵	۴۵	۰	۴۵	۰	۴۵	۰	۴۵	۰	۴۵	۰
S _۵	۴, ۲, ۴, ۴	۲۵	۶۱	۰	۶۱	۰	۶۱	۰	۶۱	۰	۶۱	۰
S _۶	۴, ۳, ۴, ۵	۸۵	۳۹	۰	۳۹	۰	۳۹	۰	۳۹	۰	۳۹	۰
S _۷	۵, ۲, ۴, ۵	۶۹	۵۵	۰	۵۵	۰	۵۵	۰	۵۵	۰	۵۵	۰
S _۸	۵, ۴, ۴, ۶	۱۳۲	۶۷	۰	۶۷	۰	۶۷	۰	۶۷	۰	۶۷	۰
S _۹	۸, ۲, ۴, ۴	۳۱۷	۶۱	۰	۶۱	۰	۶۱	۰	۶۱	۰	۶۱	۰
S _{۱۰}	۶, ۳, ۴, ۶	۴۵۲	۶۳	۰	۶۳	۰	۶۳	۰	۶۳	۰	۶۳	۰

* تعداد (محصول، بیشینه تعداد قطعات، بیشینه تعداد عملیات، بیشینه تعداد ماشین).

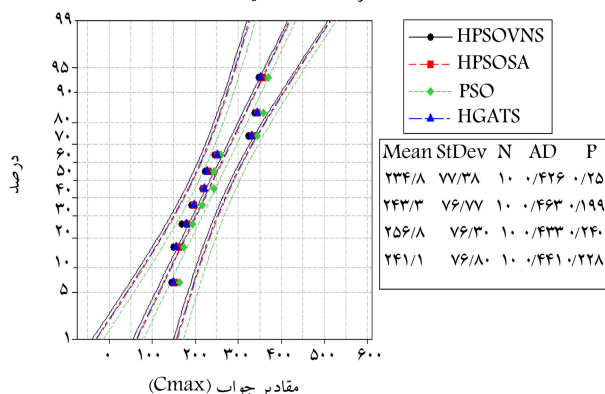
جدول ۶. نتایج محاسباتی مسائل نمونه در ابعاد متوسط.

HGATS			PSO			HPSOSA			HPSOVNS			Gams		نمونه
درصد انحراف نسبی	انحراف معیار	میانگین مقادیر C_{max}	درصد انحراف نسبی	انحراف معیار	میانگین مقادیر C_{max}	درصد انحراف نسبی	انحراف معیار	میانگین مقادیر C_{max}	درصد انحراف نسبی	انحراف معیار	میانگین مقادیر C_{max}	میانگین مقادیر C_{max}	سایز	
۱۰٫۵۹	۱٫۴۸	۹۴	۲۹٫۴۱	۲٫۵۹	۱۱۰	۵٫۸۸	۱٫۱۲	۹۰	۹٫۴۱	۱٫۴۸	۹۳	۸۵	۸٫۲٫۶٫۶	M_1
۱۱٫۷۶	۱٫۵۸	۱۱۴	۲۷٫۴۵	۲٫۳۸	۱۳۰	۱۳٫۷۳	۲٫۲۴	۱۱۶	۷٫۸۴	۰٫۴۳	۱۱۰	۱۰۲	۸٫۲٫۶٫۸	M_2
۱۰٫۲۰	۱٫۹۲	۱۰۸	۲۸٫۵۷	۱٫۱۲	۱۲۶	۶٫۱۲	۱٫۴۸	۱۰۴	۸٫۱۶	۲٫۱۲	۱۰۶	۹۸	۹٫۲٫۶٫۹	M_3
۹٫۵۲	۲٫۵۵	۱۶۱	۲۱٫۷۷	۲٫۱۷	۱۷۹	۸٫۱۶	۱٫۴۱	۱۵۹	۴٫۰۸	۱٫۴۸	۱۵۳	۱۴۷	۱۰٫۵٫۶٫۶	M_4
۲۰٫۷۳	۲٫۷۴	۹۹	۴۲٫۶۸	۲٫۱۷	۱۱۷	۲۰٫۷۳	۲٫۷۴	۹۹	۱۰٫۹۸	۱٫۶۴	۹۱	۸۲	۹٫۳٫۶٫۱۱	M_5
۲۴٫۳۲	۲٫۶۸	۹۲	۳۶٫۴۹	۲٫۳۸	۱۰۱	۱۴٫۸۶	۱٫۵۸	۸۵	۶٫۷۶	۱	۷۹	۷۴	۶٫۵٫۷٫۱۲	M_6
۱۶٫۶۷	۱٫۹۲	۱۰۵	۲۵٫۵۶	۳٫۶۴	۱۱۳	۱۷٫۷۸	۲٫۹۶	۱۰۶	۶٫۶۷	۱٫۷۹	۹۶	۹۰	۸٫۴٫۶٫۱۲	M_7
۱۰٫۲۹	۱٫۸	۱۵۰	۳۳٫۰۹	۲٫۱۲	۱۸۱	۱۱٫۰۳	۱٫۹۲	۱۵۱	۵٫۱۵	۲٫۵۹	۱۴۳	۱۳۶	۱۳٫۲٫۶٫۹	M_8
۱۰٫۱۵	۲٫۲۹	۲۱۷	۱۶٫۷۵	۳٫۷	۲۳۰	۸٫۶۳	۱٫۸۷	۲۱۴	۴٫۰۶	۱٫۹۲	۲۰۵	۱۹۷	۱۳٫۳٫۵٫۵	M_9
۱۱٫۲۸	۲٫۵	۱۴۶	۲۱٫۸۰	۳٫۳۲	۱۶۲	۱۰٫۵۳	۱٫۴۸	۱۴۷	۳٫۷۶	۱٫۸	۱۳۸	۱۳۳	۱۲٫۲٫۶٫۱۰	M_{10}

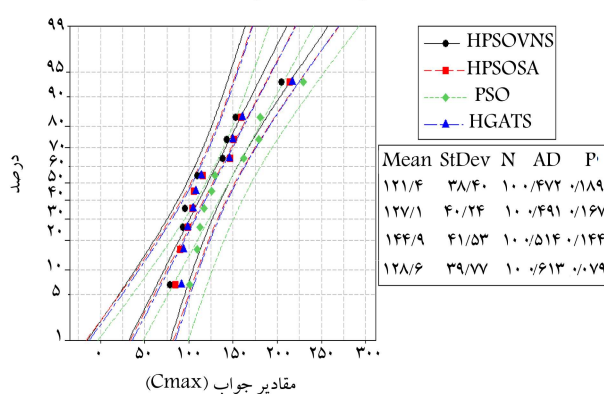
جدول ۷. نتایج محاسباتی مسائل نمونه در ابعاد بزرگ.

HGATS			PSO			HPSOSA			HPSOVNS			نمونه	
درصد انحراف نسبی	انحراف معیار	میانگین مقادیر C_{max}	درصد انحراف نسبی	انحراف معیار	میانگین مقادیر C_{max}	درصد انحراف نسبی	انحراف معیار	میانگین مقادیر C_{max}	درصد انحراف نسبی	انحراف معیار	میانگین مقادیر C_{max}		سایز
۴٫۱۷	۱٫۸۷	۱۵۰	۱۳٫۸۹	۲٫۶۹	۱۶۴	۶٫۲۵	۳٫۱۱	۱۵۳	۰	۱٫۱۲	۱۴۴	۹٫۴٫۶٫۱۴	L_1
۶٫۵۱	۱٫۴۱	۱۸۰	۱۴٫۷۹	۲٫۶۸	۱۹۴	۷٫۶۹	۲٫۵۵	۱۸۲	۰	۱٫۴۸	۱۶۹	۱۳٫۳٫۶٫۱۷	L_2
۳٫۱۴	۰٫۸۳	۱۹۷	۱۳٫۶۱	۱٫۵۸	۲۱۷	۳٫۶۶	۱٫۵۸	۱۹۸	۰	۱٫۳	۱۹۱	۱۱٫۱۲٫۳٫۶	L_3
۲٫۳۱	۲٫۵۹	۲۲۱	۱۲٫۵	۲٫۶۹	۲۴۳	۱٫۸۵	۲٫۱۷	۲۲۰	۰	۱٫۰۹	۲۱۶	۱۷٫۴٫۹٫۹	L_4
۴٫۷	۱٫۸	۱۵۶	۱۶٫۱۱	۲٫۱۷	۱۷۳	۸٫۷۲	۲٫۳۸	۱۶۲	۰	۰٫۷۱	۱۴۹	۲۵٫۲٫۱۰٫۱۴	L_5
۲٫۲۴	۵٫۸	۲۲۸	۹٫۴۲	۲٫۲۸	۲۴۴	۲٫۵۹	۱٫۸۷	۲۳۱	۰	۱٫۶۸	۲۲۳	۳۰٫۶٫۴٫۸	L_6
۲٫۰۳	۲٫۲۹	۲۵۱	۵٫۶۹	۳٫۳۴	۲۶۰	۱٫۶۳	۱٫۵۸	۲۵۰	۰	۱٫۰۹	۲۴۶	۳۳٫۴٫۷٫۶	L_7
۲٫۱۶	۲٫۲۴	۳۳۱	۶٫۱۷	۳٫۱۱	۳۴۴	۳٫۰۹	۱٫۴۸	۳۳۴	۰	۰٫۸۷	۳۲۴	۳۸٫۳٫۹٫۷	L_8
۱٫۴۴	۲٫۷۴	۳۵۳	۶٫۳۲	۱٫۸۷	۳۷۰	۲٫۸۷	۱٫۹۲	۳۵۸	۰	۱٫۶۴	۳۴۸	۴۲٫۲٫۶٫۷	L_9
۱٫۷۸	۱٫۴۸	۳۴۴	۶٫۲۱	۲٫۹۵	۳۵۹	۲٫۰۷	۱٫۸	۳۴۵	۰	۱٫۹۲	۳۳۸	۵۰٫۳٫۶٫۹	L_{10}

نمودار احتمال الگوریتم های HPSOVNS, HPSOSA, PSO, HGATS
نرمال - فاصله اطمینان ۹۵٪



نمودار احتمال الگوریتم های HPSOVNS, HPSOSA, PSO, HGATS
نرمال - فاصله اطمینان ۹۵٪



شکل ۹. آزمون نرمال بودن مقادیر الگوریتم‌ها برای مسائل با ابعاد بزرگ.

شکل ۸. آزمون نرمال بودن مقادیر الگوریتم‌ها برای مسائل با ابعاد متوسط.

جدول ۸. نتایج آنالیز واریانس طرح بلوکی تصادفی برای مسائل با ابعاد متوسط.

منبع	مجموع مربعات	درجه آزادی	میانگین مربعات	آماره F	P-value
فاکتور	۳۰۵۳٫۴	۳	۱۰۱۷٫۸	۷۷٫۵	۰
بلوک	۵۷۲۴۶	۹	۶۳۶۰٫۷	۴۸۴٫۳	۰
خطا	۳۵۴	۲۷	۱۳٫۱۳		
کل	۷۴۱۸۶۴	۴۰			

جدول ۹. نتایج آنالیز واریانس طرح بلوکی تصادفی برای مسائل با ابعاد بزرگ.

منبع	مجموع مربعات	درجه آزادی	میانگین مربعات	آماره F	P-value
فاکتور	۲۵۷۳٫۸	۳	۸۵۷٫۹۳	۱۶۰٫۸۴	۰
بلوک	۲۱۲۲۶۱٫۵	۹	۲۳۵۸۴٫۶۱	۴۴۰۰٫۷	۰
خطا	۱۴۴٫۷	۲۷	۵٫۳۶		
کل	۲۵۹۶۴۲۰	۴۰			

جدول ۱۰. نتایج آزمون حداقل اختلاف معنادار برای مسائل با ابعاد متوسط.

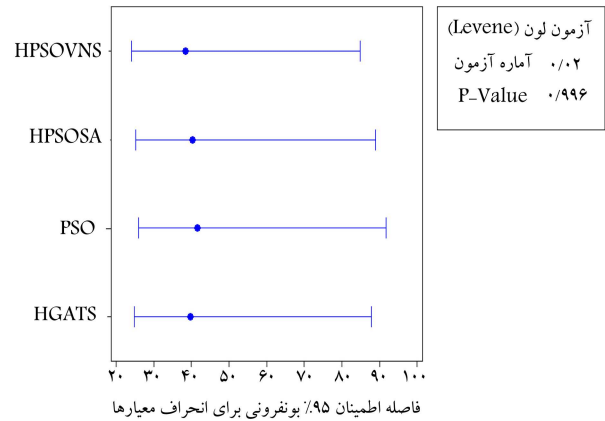
الگوریتم	الگوریتم‌ها	P-value
HPSOSA		۰٫۰۰۲
HPSOVNS	PSO	۰
HPSOVNS	HGATS	۰
HPSOSA	PSO	۰
HPSOSA	HGATS	۰٫۳۶۳
PSO	HGATS	۰

جدول ۱۱. نتایج آزمون حداقل اختلاف معنادار برای مسائل با ابعاد بزرگ.

الگوریتم	الگوریتم‌ها	P-value
HPSOSA		۰
HPSOVNS	PSO	۰
HPSOVNS	HGATS	۰
HPSOSA	PSO	۰٫۰۴۳
HPSOSA	HGATS	۰
PSO	HGATS	۰

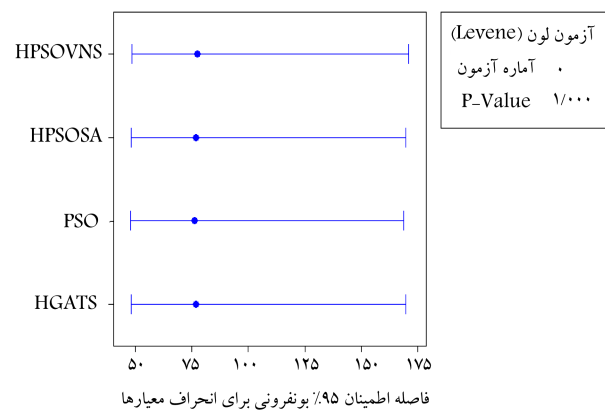
الگوریتم‌ها کم‌تر از سطح معناداری ۰٫۰۵ است. لذا نتایج نشان می‌دهد که سایر الگوریتم‌ها در بعد متوسط، از نظر توانایی در رسیدن به جواب بهینه، با یکدیگر اختلاف معنادار دارند. چنان که در جدول ۱۱ مشاهده می‌شود، مقادیر P-value مقایسه‌ی دوبه‌دو همه الگوریتم‌ها کم‌تر از سطح معناداری ۰٫۰۵ است. لذا نتایج نشان می‌دهد که هر چهار الگوریتم در بعد بزرگ، از نظر توانایی در رسیدن به جواب بهینه، با یکدیگر اختلاف معنادار دارند. براساس میانگین مقادیر الگوریتم‌ها می‌توان بیان کرد که میانگین الگوریتم HPSOVNS در هر دو بعد کم‌تر از سه الگوریتم دیگر است، لذا این الگوریتم عملکرد بهتری نسبت به سایر الگوریتم‌ها دارد.

آزمون برابری واریانس‌ها



شکل ۱۰. آزمون برابری واریانس‌های مقادیر الگوریتم‌ها برای مسائل با ابعاد متوسط.

آزمون برابری واریانس‌ها



شکل ۱۱. آزمون برابری واریانس‌های مقادیر الگوریتم‌ها برای مسائل با ابعاد بزرگ.

بیشتر از ۰٫۰۵ است. بنابراین هر چهار الگوریتم در ابعاد متوسط و بزرگ دارای توزیع نرمال هستند. نتایج آزمون برابری واریانس الگوریتم‌ها در حل مسائل با ابعاد متوسط و بزرگ به ترتیب در شکل‌های ۱۰ و ۱۱ نشان داده شده است. مقادیر P-value این آزمون در هر دو بعد بیشتر از سطح معناداری ۰٫۰۵ است. بنابراین، واریانس الگوریتم‌ها با یکدیگر برابرند. با توجه به نتایج آزمون‌های انجام شده در فوق، فرض‌های استفاده از آزمون پارامتریک برقرار است.

در این تحقیق، از تحلیل واریانس طرح بلوکی تصادفی استفاده شده است. در این آزمون، الگوریتم‌ها به‌عنوان فاکتور و سایر عوامل تأثیرگذار به‌عنوان بلوک در نظر گرفته شده‌اند. آنالیز واریانس طرح بلوکی تصادفی در نرم‌افزار Spss انجام و نتایج این آزمون برای ابعاد متوسط و بزرگ مسئله محاسبه و به ترتیب در جداول ۸ و ۹ نشان داده شده است. در این جداول، مقادیر P-value الگوریتم‌ها و بلوک‌ها کم‌تر از سطح معناداری ۰٫۰۵ است، لذا فرض صفر رد می‌شود. برای این که مشخص شود کدام الگوریتم‌ها با یکدیگر اختلاف معنادار دارند، از آزمون حداقل اختلاف معنادار^{۱۴} استفاده شده است. نتایج این آزمون برای ابعاد متوسط و بزرگ به ترتیب در جداول ۱۰ و ۱۱ نشان داده شده است. در جدول ۱۰ مقادیر P-value مقایسه‌ی دوبه‌دو الگوریتم‌های HPSOSA و HGATS بیشتر از سطح معناداری ۰٫۰۵ است؛ بنابراین می‌توان گفت اختلاف معناداری بین این دو الگوریتم در بعد متوسط وجود ندارد و از نظر عملکردی یکسان‌اند. مقادیر P-value مقایسه‌ی دوبه‌دو سایر

۶. نتیجه‌گیری

در نوشتار حاضر مسئله‌ی زمان‌بندی تولید کارگاهی انعطاف‌پذیر با عملیات مونتاژ و زمان آماده‌سازی وابسته به توالی مورد بررسی قرار گرفت. در این سیستم تولیدی، قطعات در مرحله‌ی تولید کارگاهی انعطاف‌پذیر تولید شده و بعد از آماده‌سازی قطعات، در مرحله‌ی مونتاژ با یکدیگر مونتاژ می‌شوند. به منظور حل مسئله یک مدل ریاضی توسعه داده شده که توانایی دست‌یابی به جواب بهینه برای مسائل با ابعاد کوچک و متوسط را دارد. با توجه به پیچیدگی بالای مسئله‌ی مورد بررسی، الگوریتم بهینه‌سازی ازدحام ذرات و دو الگوریتم فراابتکاری ترکیبی برای حل مسائل با ابعاد متوسط و بزرگ پیشنهاد شد. در رویکردهای ترکیبی پیشنهادی، الگوریتم بهینه‌سازی ازدحام ذرات به کاوش در فضای جست‌وجو و الگوریتم‌های جست‌وجوی همسایگی متغیر و شبیه‌سازی تبرید به تبعیت از بهترین راه‌حل به دست آمده در هر تکرار می‌پردازند. به منظور اعتبارسنجی الگوریتم‌های پیشنهادی، نتایج به دست آمده از الگوریتم‌ها با الگوریتم ترکیبی ژنتیک و جست‌وجوی ممنوع لی و گاتو^[۳۲] مقایسه شد. برای مسائل با ابعاد کوچک، نتایج نشان داد که الگوریتم‌های پیشنهادی و الگوریتم ترکیبی لی و گاتو^[۳۲] به جواب بهینه مشابه راه‌حل بهینه به دست آمده توسط گمز، دست می‌یابند. با این تفاوت که با افزایش ابعاد مسئله، زمان محاسباتی حاصل

از این الگوریتم‌ها کم‌تر از نرم‌افزار گمز است. برای مسائل نمونه در ابعاد متوسط نیز جواب مدل توسط نرم‌افزار گمز طی تقریباً دو ساعت اجرا محاسبه شد. به دلیل زیاد بودن زمان محاسباتی، مسائل نمونه در ابعاد متوسط توسط الگوریتم‌های پیشنهادی حل شد. به منظور مقایسه توانایی الگوریتم‌های فراابتکاری در یافتن جواب بهینه در ابعاد متوسط و بزرگ، از آنالیز واریانس طرح بلوکی تصادفی استفاده شد. نتایج این آزمون نشان داد که در هر دو بعد، عملکرد الگوریتم HPSOVNS نسبت به الگوریتم‌های پیشنهادی و الگوریتم ترکیبی لی و گاتو^[۳۲] بهتر است. از طرفی طی انجام محاسبات مشاهده شد که الگوریتم بهینه‌سازی ازدحام ذرات در هر دو بعد، به نتایج ضعیف‌تری نسبت به سایر الگوریتم‌ها دست می‌یابد. لذا می‌توان بیان کرد که رویکرد ترکیبی موجب رسیدن به جواب‌های با کیفیت و پایداری می‌شود.

در ادامه، پیشنهادات برای تحقیقات آتی عبارت است از: در نظر گرفتن جریان محموله در صورتی که بیش از یک محصول وجود داشته باشد، ارائه‌ی مدل در حالت چندهدفه و در نظر گرفتن توابع هدف دیگر، در نظر گرفتن حالت احتمالی و غیرقطعی برای پارامترهای مسئله. همچنین به منظور بالا بردن سرعت همگرایی پیشنهاد می‌شود از ترکیب سایر الگوریتم‌های فراابتکاری و جست‌وجوی محلی استفاده شود که ضمن تولید جواب‌های با کیفیت از سرعت همگرایی خوبی برخوردار باشند.

پانویس‌ها

- flexible job shop scheduling with assembly operation
- two-stage production system
- genetic algorithms (GA)
- greedy random adaptive search procedure (GRASP)
- artificial immune system (AIS)
- simulated annealing (SA)
- particle swarm optimization (PSO)
- personal-best (P-best)
- global-best (G-best)
- hybrid particle swarm optimization and variable neighborhood search (HPSOVNS)
- variable neighborhood search (VNS)
- hybrid particle swarm optimization and simulated annealing (HPSOSA)
- hybrid genetic algorithm and tabu search (HGATS)
- least significant difference (LSD)

منابع (References)

- Lee, C.Y., Cheng, T.C.E. and Lin, B.M.T. "Minimizing the makespan in the 3-machine assembly-type flow shop scheduling problem", *Management Science*, **39**(5), pp. 616-625 (1993).
- Potts, C.N., Sevast' Janov, S.V., Strusevich, V.A., Van Wassenhove, L.N. and Zwaneveld, C.M. "The two-stage assembly scheduling problem: Complexity and approximation", *Operations Research*, **43**(2), pp. 346-355 (1995).
- Allahverdi, A., Gupta, J.N.D. and Aldowaisan, T. "A review of scheduling research involving setup considerations", *OMEGA*, **27**(2), pp. 219-239 (1999).
- Hariri, A.M.A. and Potts, C.N. "A branch and bound algorithm for the two-stage assembly scheduling problem", *European Journal of Operational Research*, **103**(3), pp. 547-556 (1997).
- Zhang, R. "Simulation-based genetic algorithm for job shop scheduling with assembly operations", *International Journal of Advancements in Computing Technology*, **3**(10), pp. 132 (2011).
- Daneshamooz, F., Jabbari, M. and Fattahi, P. "A model for job shop scheduling with a parallel assembly stage to minimize makespan", *Journal of Industrial Engineering Research in Production Systems*, **2**(4), pp. 39-53 (2015).
- Fattahi, P., Hosseini, S.M.H. and Jolai, F. "Some heuristics for the hybrid flow shop scheduling problem with setup and assembly operations", *International Journal of Industrial Engineering Computations*, **4**(3), pp. 393-416 (2013).
- Komaki, G.M., Teymourian, E. and Kayvanfar, V. "Minimising makespan in the two-stage assembly hybrid flow shop scheduling problem using artificial immune systems", *International Journal of Production Research*, **54**(4), pp. 963-83 (2015).
- Fattahi, P., Hosseini, S.M.H. and Jolai, F. "A mathematical model and extension algorithm for assembly flexible flow shop scheduling problem", *The International Journal of Advance Manufacture Technology*, **65**(5), pp. 787-802 (2013).

10. Nejati, M., Mahdavi, I., Hassanzadeh, R. and Mahdavi-Amiri, N. "Lot streaming in a two-stage assembly hybrid flow shop scheduling problem with a work shift constraint", *Journal of Industrial and Production Engineering*, **33**(7), pp. 459-471 (2016).
11. Yokoyama, M. "Flow-shop scheduling with setup and assembly operations", *European Journal of Operational Research*, **187**(3), pp. 1184-1195 (2008).
12. Allahverdi, A. and Al-Anzi, F.S. "The two-stage assembly scheduling problem to minimize total completion time with setup times", *Computers & Operations Research*, **36**(10), pp. 2740-2747 (2009).
13. Navaei, J., Fatemi-Ghomi, S.M.T., Jolai, F. and Mozdgir, A. "Heuristics for an assembly flow-shop with non-identical assembly machines and sequence dependent setup times to minimize sum of holding and delay costs", *Computers & Operations Research*, **44**, pp. 52-65 (2014).
14. Azadeh, A., Jaihoonian, M., Maleki-SHoja, B. and Seyedmahmoudi, S.H. "An integrated neural network-simulation algorithm for performance optimization of the bi-criteria two-stage assembly flow-shop scheduling problem with stochastic activities", *International Journal of Production Research*, **50**(24), pp. 7271-7284 (2012).
15. Garey, M.R., Johnson, D.S. and Sethi, R. "The complexity of flow shop and job shop scheduling", *Mathematics of Operations Research*, **1**(2), pp. 117-129 (1976).
16. Lian, Z., Gu, X. and Jiao, B. "A novel particle swarm optimization algorithm for permutation flow shop scheduling to minimize makespan", *Chaos, Solitons and Fractals*, **35**(5), pp. 851-861 (2008).
17. Bo, L., Ling, W., Yi-Hui, J., Fang, T. and De-Xian, H. "Improved particle swarm optimization combined with chaos", *Chaos, Solitons & Fractals*, **25**(5), pp. 1261-1271 (2005).
18. Shi, X.H., Liang, Y.C., Lee, H.P., Lu, C. and Wang, L.M. "An improved GA and a novel PSO-GA based hybrid algorithm", *Information Processing Letters*, **93**(5), pp. 255-261 (2005).
19. Kennedy, J. and Mendes, R. "Population structure and particle swarm performance", *Proceedings of the 2002 Congress on Evolutionary Computation*, **2**, pp. 1671-1676 (2002).
20. Fan, H.Y. "A modification to particle swarm optimization algorithm", *Engineering Computations*, **19**(8), pp. 970-989 (2002).
21. Robinson, J., Sinton, S. and Rahmat-Samii, Y. "Particle swarm, genetic algorithm, and their hybrids: Optimization of a profiled corrugated horn antenna", in: *IEEE Antennas and Propagation Society International Symposium*, San Antonio, pp. 314-317 (2002).
22. Saidi-Mehrabad, M. and Fattahi, P. "Flexible job shop scheduling with tabu search algorithms", *The International Journal of Advanced Manufacturing Technology*, **32**(5), pp. 563-570 (2007).
23. Eberhart, R. and Kennedy, J. "A new optimizer using particle swarm theory", In *Conference: Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, Nagoya, Japan, pp. 39-43 (1995).
24. Mladenovic, N. and Hansen, P. "Variable neighborhood search", *Computers and Operations Research*, **24**(11), pp. 1097-1100 (1997).
25. Hansen, P. and Mladenovic, N. "Variable neighborhood search: Principles and applications", *European Journal of Operational Research*, **130**(3), pp. 449-467 (2001).
26. Gen, M., Tsujimura, Y. and Kubota E. "Solving job-shop scheduling problem using genetic algorithms", In: *Proceedings of the 16th International Conference on Computer and Industrial Engineering*, Ashikaga, Japan, pp. 576-579 (1994).
27. Pezzella, F., Morganti, G. and Ciaschetti, G. "A genetic algorithm for the flexible job-shop scheduling problem", *Computers & Operations Research*, **35**(10), pp. 3202-3212 (2008).
28. Kacem, I., Hammadi, S. and Borne, P. "Approach by localization and multi objective evolutionary optimization for flexible job-shop scheduling problems", *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, **32**(1), pp. 1-13 (2002).
29. Yazdani, M., Amiri, M. and Zandieh, M. "Flexible job-shop scheduling with parallel variable neighborhood search algorithm", *Expert Systems with Applications*, **37**(1), pp. 678-687 (2010).
30. Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H. and Teller, E. "Equation of state calculation by east computing machines", *The Journal of Chemical Physics*, **21**, pp. 1087-1091 (1953).
31. Kirkpatrick, S., Gelatt, C.D. and Vecchi, M.P. "Optimization by simulated annealing", *Science, New Series*, **220**, pp. 671-680 (1983).
32. Varadharajan, T.K. and Rajendran, C. "A multi-objective simulated-annealing algorithm for scheduling in flow shops to minimize the makespan and total flow time of jobs", *European Journal of Operational Research*, **167**(3), pp. 772-795 (2005).
33. Li, X. and Gao, L. "An effective hybrid genetic algorithm and tabu search for flexible job shop scheduling problem", *International Journal of Production Economics*, **174**, pp. 93-110 (2016).