

ارائه مدل زمان‌بندی چندعاملی در محیط جریان کارگاهی با فرض زوال‌پذیری کارها، زمان‌های آماده‌سازی وابسته به توالی و زمان آزادسازی کارها با استفاده از الگوریتم ازدحام ذرات چندهدفه

سمیه قندی بیدگلی^{۱*}، مرضیه امینی رارانی^۲

۱. استادیار گروه مهندسی صنایع، دانشکده مهندسی، دانشگاه کاشان، کاشان، ایران.

۲. دانشجوی کارشناسی‌ارشد، دانشکده مهندسی صنایع، دانشگاه علم و صنعت، تهران، ایران.

خلاصه

زمان‌بندی چندعاملی در شرایط زوال‌پذیری کارها در سال‌های اخیر مورد توجه روزافزون جوامع دانشگاهی و صنعتی قرار گرفته‌است. مسأله زمان‌بندی چندعاملی، زیرمجموعه‌ای از مسائل زمان‌بندی چندهدفه است که در آن هر عامل، دارای مجموعه‌ای از کارها است و هدف آن، بهینه کردن تابع هدف مربوط به خود است. در این پژوهش یک مسأله زمان‌بندی سه‌عاملی در محیط جریان کارگاهی در شرایط زوال‌پذیری کارها مورد بررسی قرار گرفته است. در مسأله در نظر گرفته شده زمان پردازش واقعی کارها تابع خطی از زمان پردازش نرمال و زمان شروع پردازش کار مربوطه می‌باشد. جهت واقعی‌تر کردن مسأله، دو فرض کاربردی «زمان‌های آماده‌سازی وابسته به توالی» و «زمان آزادسازی کاره» نیز در نظر گرفته شده‌اند. همچنین یک مدل برنامه‌ریزی عدد صحیح مختلط برای مسأله توسعه داده شده که برای حل آن از روش حل دقیق محدودیت جزئی تعمیم‌یافته استفاده شده است. با توجه به پیچیدگی مدل و عدم توانایی روش محدودیت جزئی تعمیم‌یافته در حل مسائل با ابعاد بزرگ، الگوریتم فراابتکاری ازدحام ذرات چندهدفه پیشنهاد شده است. به منظور ارزیابی عملکرد الگوریتم پیشنهادی، به حل مسائل نمونه عددی در اندازه‌های مختلف با استفاده از این الگوریتم و الگوریتم ژنتیک مبتنی بر مرتب‌سازی نامغلوب و روش حل دقیق محدودیت اپسیلون تعمیم‌یافته پرداخته شده است. سپس جهت انتخاب الگوریتم برتر از روش تصمیم‌گیری چندمعیاره ویکور استفاده شد. نتایج محاسباتی، مؤید همگرایی قابل قبول و پراکندگی خیلی خوب راه‌حل‌های الگوریتم MOPSO و همچنین عملکرد بهتر این الگوریتم نسبت به روش محدودیت اپسیلون تقویت‌شده و الگوریتم NSGA-II می‌باشد.

اطلاعات مقاله

تاریخچه مقاله:

دریافت ۱۳۹۹/۱۰/۱

پذیرش ۱۴۰۰/۴/۵

(مقاله پژوهشی)

کلمات کلیدی:

زمان‌بندی جریان کارگاهی

زوال‌پذیری کارها

زمان‌بندی چندعاملی

الگوریتم فراابتکاری ازدحام

ذرات چندهدفه

زمان آماده‌سازی وابسته به

توالی

زمان آزادسازی کارها

۱. مقدمه

زمان‌بندی و توالی از کارها به منظور بهینه کردن توابع هدف موردنظر است. اما در مسائل زمان‌بندی کاربردی، ممکن است انواع متفاوتی از مشتریان وجود داشته باشند. برخی از مشتریان، به دلیل عدم تحمل تأخیر ممکن است هزینه بیشتری را برای تحویل به‌موقع کالای خود بپردازند. اما برای برخی مشتریان، هزینه اهمیت بیشتری نسبت به زمان دارد. بنابراین یک زمان‌بندی خوب باید به‌طور همزمان، انتظارات مختلف انواع مشتریان را برآورده سازد. به دسته‌ای از مسائل زمان‌بندی

زمان‌بندی نوعی فرآیند تصمیم‌گیری است که با هدف بهینه‌سازی یک و یا چند هدف انجام می‌گیرد. این موضوع در بازار رقابتی کنونی به یک نیاز اساسی برای بقای کارخانجات و محیط‌های تولیدی تبدیل شده است [۱].

در مسائل زمان‌بندی کلاسیک، اغلب فرض می‌گردد که همه کارها متعلق به یک مشتری (منبع) بوده و هدف مسأله، پیدا کردن یک

* نویسنده مسئول: سمیه قندی بیدگلی

تلفن: ۰۳۱-۵۵۹۱۲۴۱۶؛ پست الکترونیکی: s.ghandi@kashanu.ac.ir

می‌توان به در نظر گرفتن زمان آماده‌سازی و زمان آزادسازی اشاره نمود. زمان‌های آماده‌سازی را می‌توان به دو دسته زمان آماده‌سازی مستقل از توالی و زمان آماده‌سازی وابسته به توالی تقسیم‌بندی نمود. زمان آماده‌سازی وابسته به توالی در شرایطی در نظر گرفته می‌شود که زمان آماده‌سازی برای یک کار به‌خصوص بستگی به کاری داشته باشد که قبل از این کار بر روی منبع (ماشین) مورد نظر پردازش گردیده است. به‌عنوان کاربردی برای این زمان‌ها می‌توان به صنایع نساجی، پلاستیک‌سازی، کاشی‌سازی، صنعت چاپ، فولادسازی و ... اشاره نمود [۵].

۲. مرور ادبیات

زوال‌پذیری اولین بار توسط گوپتا و گوپتا در یک مسأله زمان‌بندی تک‌ماشین مطرح گردید [۶]. در مدل ارائه شده زمان پردازش واقعی کارها تابعی از زمان شروع پردازش آن‌ها بود. مدل مذکور به‌عنوان مبنایی جهت مدل نمودن کاربردهای دنیای واقعی مانند صنایع نورد فولاد مورد استفاده قرار گرفت [۷]. هم‌چنین در مقاله کاناتور و گوپتا مسأله حداقل‌سازی زمان انجام آخرین کار با محدودیت اثر زمانی روی تک‌ماشین بررسی گردیده است [۸]. در این تحقیق فرض شده است که زمان پردازش واقعی یک کار، تابعی از زمان شروع آن می‌باشد. اما در پژوهش چنگ و همکاران [۹] مدل جدیدی از زوال‌پذیری ارائه گردیده است که در آن زمان پردازش واقعی یک کار تابعی از زمان پردازش کارهایی می‌باشد که قبل از آن پردازش شده‌اند. پژوهش‌های ذکر شده همگی در محیط تک‌ماشینی انجام شده است. زوال‌پذیری در محیط جریان کارگاهی اخیراً در تحقیقاتی مانند [۱۰] بررسی شده که در آن دو مدل جهت حداقل نمودن زمان اتمام آخرین کار و مجموع زمان تکمیل تمام کارها ارائه گردیده‌اند. لی و همکاران یک الگوریتم شاخه و کران را جهت حل مسأله زمان‌بندی جریان کارگاهی با فرض زوال‌پذیری کارها و هدف حداقل نمودن کل دیرکرد بیان کردند [۱۱]. وانگ و وانگ مسأله زمان‌بندی جریان کارگاهی دو ماشین با فرض زوال‌پذیری کارها و هدف حداقل نمودن مجموع زمان تکمیل وزن‌دار تمام کارها را در نظر گرفته و جهت حل این مسأله یک الگوریتم شاخه و کران را طراحی نموده‌اند [۱۲]. چن و همکاران مسأله زمان‌بندی جریان کارگاهی دو ماشین با فرض زوال‌پذیری کارها را با هدف حداقل نمودن مجموع زمان تکمیل تمام کارها [۱۳] و جمع وزنی زمان اتمام آخرین کار و مجموع زمان تکمیل تمام کارها [۴] در نظر گرفته‌اند. مطالعاتی مانند [۱۴]–[۱۷] مسأله زمان‌بندی جریان کارگاهی را با دو فرض زوال‌پذیری کارها و زمان‌های آماده‌سازی وابسته به توالی مدل‌سازی کرده‌اند.

تعداد زیادی از پژوهش‌های زمان‌بندی جهت ساده‌سازی مسأله، زمان آزادسازی را برابر با صفر در نظر قرار داده‌اند. در مقاله [۱۸] برای اولین بار محدودیت زمان آزادسازی در یک محیط تولیدی تک‌ماشینی

که در آن‌ها کارها مربوط به مشتریان (منابع) مختلف باشند، زمان‌بندی چندعاملی^۱ و به هر یک از این مشتریان یک عامل گفته می‌شود [۲]. در مسأله زمان‌بندی چندعاملی، هر عامل (مشتری) دارای مجموعه‌ای از کارها بوده و به‌دنبال بهینه نمودن تابع هدف مربوط به خود می‌باشد. هدف این مسأله، یافتن توالی از کارهای همه عامل‌ها می‌باشد که در آن تعادلی بین اهداف عامل‌ها ایجاد گردد. در مسائل دنیای واقعی، معمولاً تولیدکننده با چندین مشتری مختلف با اهداف گوناگون سروکار داشته و باید در حین زمان‌بندی کارها، اهداف تمام مشتریان را لحاظ نماید. این امر دلیل و اثباتی برای اهمیت مسأله زمان‌بندی چندعاملی می‌باشد. زیرا در نظر گرفتن تنها یک یا دو عامل در مسأله نمی‌تواند پاسخ‌گوی شرایط کاربردی باشد. مسأله زمان‌بندی با عامل، زیرشاخه‌ای از زمان‌بندی چندهدفه بوده و نقش مهمی در مسائل زمان‌بندی واقعی و آکادمیک ایفا می‌نماید. از کاربردهای این زمان‌بندی می‌توان به صنعت حمل‌ونقل ریلی، هوایی و سازمان‌های پروژه‌محور اشاره نمود [۳]. جهت حل مسأله زمان‌بندی چندعاملی دو روش توسعه داده شده است. روش اول به‌دنبال بهینه نمودن جمع وزنی اهداف تمام مشتریان می‌باشد. در روش دوم هدف یک مشتری به‌خصوص منوط به قرار گرفتن میزان هدف سایر مشتریان در یک حد مشخص بهینه می‌گردد.

در زمان‌بندی جریان کارگاهی m ماشین به‌صورت سری وجود داشته و هر یک از کارها باید با ترتیب یکسان و مشخص روی تمام ماشین‌ها پردازش گردند. به‌عبارت دیگر مسأله زمان‌بندی جریان کارگاهی عبارت از یافتن توالی بهینه کارهای مختلف روی تعدادی ماشین بوده و از جمله مسائل مشکل در حوزه زمان‌بندی می‌باشد. باتوجه به اهمیت روزافزون مسائل جریان کارگاهی، در نظر گرفتن معیارهای مختلف مورد توجه محققان این علم می‌باشد. توجه به مسائل واقعی برای صنعت امری حیاتی به‌شمار می‌رود. در این پژوهش یک مدل جدید پیشنهاد شده است که برای نزدیک شدن به واقعیت، به بررسی هم‌زمان زوال‌پذیری کارها^۲، زمان‌های آماده‌سازی وابسته به توالی^۳ و زمان آزادسازی^۴ کارها پرداخته است.

در بسیاری از مطالعات، زمان پردازش قطعات به‌صورت ثابت و بدون تغییر فرض می‌گردد. در صورتی که این امر معمولاً در شرایط دنیای واقعی محقق نمی‌گردد. در برخی شرایط، قطعه با مدت زمان انتظار بیشتر در صف نیازمند زمان پردازش بیشتری نسبت به قطعه‌ای می‌باشد که مدت زمان کمتری در صف انتظار بوده است. به این وضعیت، زوال‌پذیری گفته می‌شود [۴]. به‌عنوان مثالی برای این وضعیت می‌توان به فرایند نورد داغ ورقه‌های فولادی اشاره نمود. هرچه ورقه فولادی داغ شده بیشتر در صف انتظار نورد بماند، سردتر شده و در نتیجه در مرحله بعد زمان بیشتری باید صرف عملیات نورد این ورقه گردد. اخیراً انواع مختلفی از مدل‌های زمان‌بندی برای توصیف بهتر مسائل زمان‌بندی عملی ارائه گردیده‌اند. از جمله این موارد

3. Sequence-dependent setup time

4. Release time

1. Multi agent scheduling

2. Deterioration

زمان تکمیل کارهای عامل دوم می‌باشد [۳۵]. لی مسأله زمان‌بندی جریان کارگاهی دو عاملی را با اهداف حداقل نمودن زمان اتمام آخرین کار و کل تأخیر در نظر گرفته‌است. در مسأله مذکور به منظور محدود نمودن و کاهش اندازه فضای جستجو برای مقادیر اهداف، مرزهایی تنظیم گردیده است و سپس یک الگوریتم جستجوی همسایگی متغیر^۶ (VNS) جهت یافتن جواب‌های امکان‌پذیر چندگانه استفاده گردیده‌اند [۳۶]. هم‌چنین در [۲] مسأله زمان‌بندی جریان کارگاهی دو عاملی با فرض زوال‌پذیری کارها در نظر گرفته شده است. در پژوهش مذکور، زمان پردازش نرمال کارها به صورت یک متغیر تصادفی در نظر گرفته شده است. هم‌چنین شیائو و همکاران یک مسأله زمان‌بندی تک ماشینه سه‌عاملی با زمان آزادسازی و فعالیت تعمیر و نگهداری را ارائه داده‌اند که در آن هدف عامل اول حداقل نمودن زمان اتمام کل کارها است. حداکثر تأخیر کارهای عامل دوم نباید از یک مقدار ثابت و محدود بیشتر شود و عامل سوم نیز به دنبال انجام فعالیت‌های تعمیر و نگهداری در یک فاصله زمانی مشخص با عنوان پنجره تعمیر و نگهداری می‌باشد. حد پایین برای شتاب دادن به الگوریتم شاخه و کران و جلوگیری از ایجاد شاخه‌های غیرضروری پیشنهاد شده‌است و نتایج محاسباتی بیانگر عملکرد قابل قبول این حد پایین می‌باشد. هم‌چنین در این مقاله از الگوریتم ژنتیک برای به‌دست آوردن جواب اولیه (نزدیک به بهینه) برای الگوریتم شاخه و کران استفاده کرده‌اند [۳۷]. نصرالهی و همکاران به بررسی مسأله دو عاملی زمان‌بندی جریان کارگاهی دو ماشینه پرداختند [۳۸]. هدف پژوهش به حداقل رساندن مجموع وزنی حداکثر زودکرد و حداکثر تأخیر کارهای عامل اول است. هم‌چنین کارهای عامل دوم مجاز به تأخیر نیستند. آن‌ها با توسعه الگوریتم شاخه و کران مسأله را با داده‌های عددی حل نمودند. بای و همکاران یک مدل چندعاملی جریان کارگاهی را ارائه کردند که هدف آن به حداقل رساندن مجموع وزنی زمان اتمام آخرین کار هر عامل است [۳۹]. الگوریتم شاخه و کران را برای حل مسأله در ابعاد کوچک مطرح کردند و برای ابعاد بزرگ الگوریتم مستقل کلونی زنبور عسل مصنوعی^۹ (DABCA) را ارائه دادند. نتایج نشان می‌دهد این الگوریتم از GA بهتر عمل می‌کند.

مسأله زمان‌بندی چندعاملی روی تک‌ماشین با در نظر گرفتن زوال‌پذیری کارها در مقاله وو و همکاران مورد بررسی قرار گرفته است [۴۰]. تابع هدف عامل اول، حداقل کردن حداکثر تأخیر وزنی بوده و هدف عامل دوم حداقل نمودن جمع وزنی تأخیرها می‌باشد. آن‌ها برای حل مسأله مذکور از الگوریتم دقیق شاخه و کران استفاده نموده‌اند. برای کم کردن تعداد تکرارهای این الگوریتم و به‌دست آوردن جواب اولیه آن از الگوریتم‌های ACO و SA استفاده شده‌است. هم‌چنین در پژوهش بین و همکاران ترکیب‌های مختلفی از اهداف عامل‌ها در

در نظر گرفته شد. پس از آن تعدادی مطالعه به توسعه الگوریتم‌های حل برای مسأله زمان‌بندی تک‌ماشینه با زمان‌های آزادسازی پرداخته‌اند. به‌عنوان مثال می‌توان به الگوریتم ژنتیک^۱ (GA) [۱۹]، الگوریتم شاخه و کران مبتنی بر یادگیری [۲۰]، [۲۱]، الگوریتم شبیه‌سازی تبرید^۲ (SA) [۲۲]، الگوریتم خانه مورچگان^۳ (ACO) [۲۳] و الگوریتم ژنتیک ترکیبی [۲۴] اشاره نمود. در برخی مطالعات مسأله زمان‌بندی تک‌ماشینه با زمان‌های آزادسازی در شرایط پیچیده‌تری مثل دو عاملی بودن [۲۵]، در نظر گرفتن زمان انقضاء [۲۶] و در نظر گرفتن زمان‌های آماده‌سازی وابسته به توالی [۲۷] بررسی گردیده است. عبدالهادی و همکاران در پژوهش خود زمان‌بندی جریان کارگاهی ترکیبی را همراه با زمان آماده‌سازی وابسته به توالی در نظر گرفتند [۲۸]. جهت حل مسأله، یک الگوریتم فراابتکاری ترکیبی پیشنهاد دادند. نتایج محاسباتی نشان می‌دهد الگوریتم ارائه شده از الگوریتم‌های ازدحام ذرات^۴ (PSO)، GA، جستجوی محلی^۵ (LSA) بهتر عمل می‌کند. بلابید و همکاران زمان آماده‌سازی مستقل از توالی را در مسأله جریان کارگاهی با جایگشت لحاظ کردند [۲۹]. برای حل مسأله در ابعاد کوچک دو روش ابتکاری براساس قاعده جانسون و الگوریتم NEH و برای ابعاد متوسط و بزرگ دو الگوریتم فراابتکاری جستجوی محلی تکرار شونده^۶ (ILSA) و الگوریتم حریص تکراری^۷ (IGA) ارائه کردند. براساس نتایج محاسباتی روش ابتکاری مبتنی بر الگوریتم NEH و الگوریتم IGA بهترین عملکردها را ثبت می‌کنند. مسأله زمان‌بندی جریان کارگاهی با زمان آزادسازی در تحقیقات خیلی کمتری نسبت به مسأله تک‌ماشینه با زمان آزادسازی در نظر گرفته شده است. به‌عنوان مثال می‌توان به مقاله [۳۰] اشاره نمود که در آن هشت الگوریتم برای حل مسأله زمان‌بندی جریان کارگاهی با زمان‌های آزادسازی ارائه گردیده است. هم‌چنین امیریان و صحرایان مسأله زمان‌بندی جریان کارگاهی چندهدفه با زمان‌های آماده‌سازی وابسته به توالی و زمان‌های آزادسازی را بررسی نموده‌اند [۳۱].

لی و همکاران یک مسأله زمان‌بندی جریان کارگاهی دو ماشینه با دو عامل را در نظر گرفته‌اند که در آن اهداف شامل حداقل نمودن زمان در جریان بودن اولین عامل به شرط عدم تأخیر در کارهای عامل دوم می‌باشند [۳۲]. لوو و همکارانش یک مدل بهینه‌سازی مجموع وزنی و یک مدل بهینه‌سازی محدود شده را جهت حل مسأله جریان کارگاهی با دو عامل پیشنهاد کردند [۳۳]. مور و موشیو تعدادی الگوریتم حل با زمان چندجمله‌ای را برای حل مسأله جریان کارگاهی دو ماشینه طراحی نموده‌اند [۳۴]. فان و چنگ دو مسأله را در یک محیط جریان کارگاهی با دو عامل ارائه داده‌اند. اولین مسأله به دنبال حداقل نمودن مجموع وزنی زمان تکمیل کارهای هر دو عامل بوده و هدف مسأله دوم حداقل نمودن جمع وزنی مجموع زمان تکمیل کارهای عامل و

6. Iterative Local Search Algorithm
7. Iterated Greedy Algorithm
8. Variable Neighborhood Search
9. Discrete Artificial Bee Colony Algorithm

1. Genetic Algorithm
2. Simulated Annealing
3. Ant Colony optimization
4. Particle Swarm Optimization
5. Local Search Algorithm

اهداف همه عامل ها در نظر گرفته شود.

(۵) مقالات موجود در زمینه جریان کارگاهی چندعاملی، یا دوهدفه بوده و یا بدون در نظر گرفتن فرضیات واقعی از قبیل زمان آماده سازی وابسته به توالی، زمان آزادسازی کارها و ... انجام شده اند. در حالی که این مسأله، در محیط چندماشینی بسیار کم مورد توجه قرار گرفته است.

مسأله زمان بندی جریان کارگاهی چندعاملی با در نظر گرفتن زوال پذیری کارها، زمان های آماده سازی وابسته به توالی و زمان های آزادسازی کارها جهت بهینه نمودن عملیات تولید در صنایع تولید کاشی، صنایع فولاد و سایر صنایع مشابه از اهمیت بالایی برخوردار می باشد. هر چند مسأله ذکر شده دارای کاربردهای عملی در بسیاری از صنایع می باشد، تاکنون در هیچ تحقیقی بررسی نگردیده است. با توجه به این شکاف تحقیقاتی، نوآوری اصلی این مقاله این است که در این مقاله برای نخستین بار مسأله زمان بندی جریان کارگاهی چندعاملی در شرایط زوال پذیری کارها، زمان آماده سازی وابسته به توالی و زمان آزادسازی کارها در نظر گرفته شده است. این امر به گسترش مرزهای تحقیقاتی در حوزه تئوری زمان بندی کمک می نماید. به منظور تسهیل کاربردهای عملی مسأله مذکور، یک مدل سه هدفه برنامه ریزی عدد صحیح خطی مختلط^۲ (MILP) برای این مسأله توسعه داده شده است. جهت حل مسأله مورد نظر، دیدگاه بهینه سازی چندهدفه استفاده شده است که هدف آن، یافتن مجموعه ای از راه حل های بهینه و فراهم نمودن اطلاعات کافی برای تصمیم گیرندگان است تا قادر به اخذ تصمیم منطقی و معقول باشند. همچنین جهت حل دقیق مسأله در ابعاد کوچک، روش محدودیت اسیلون تعمیم یافته پیشنهاد شده است. با توجه به پیچیدگی ذاتی موجود در مسأله مورد نظر و چندهدفه بودن آن، یک الگوریتم ازدحام ذرات چندهدفه^۳ (MOPSO) برای حل این مسأله در ابعاد متوسط و بزرگ به صورت موثر و کارآمد ارائه گردیده و با به کارگیری الگوریتم ارائه شده برای حل مسائل مختلفی در اندازه های کوچک، متوسط و بزرگ و مقایسه نتایج با دو روش محدودیت اسیلون تقویت شده و الگوریتم -NSGA II، کارایی و امکان پذیری به کارگیری الگوریتم MOPSO ارائه شده برای حل مسأله مورد بررسی اعتبارسنجی گردیده است.

ساختار مقاله به این صورت است که ابتدا در بخش ۳، مسأله زمان بندی جریان کارگاهی سه عاملی با فرض زوال پذیری کارها، زمان آماده سازی وابسته به توالی و زمان آزادسازی کارها به صورت کامل معرفی گردیده است. در بخش ۴ ابتدا روش محدودیت اسیلون تعمیم یافته مطرح شده و سپس اصول و اجزای الگوریتم ازدحام ذرات چندهدفه برای مسأله و نحوه ایجاد جواب با استفاده از این الگوریتم توضیح داده شده است. نتایج حاصل از اجرای الگوریتم مذکور برای مسائل نمونه عددی در اندازه های مختلف در بخش ۵ به صورت کامل ارائه و تحلیل گردیده و در بخش ۶، نکات مهم انجام شده در مقاله

مسأله زمان بندی چندعاملی با در نظر گرفتن زوال پذیری مورد پژوهش قرار گرفته است [۴۱]. برای هر کدام از مسائل طرح شده، پیچیدگی محاسباتی و قابلیت حل آن برای پیدا کردن جواب بهینه عامل اول تحت این شرط که حداکثر زمان زودکرد عامل دوم از یک حد بالا کمتر باشد، بررسی شده است. لی و همکاران نیز مسأله زمان بندی چندعاملی را در شرایط زوال پذیری مورد بررسی قرار داده اند [۹]. در این پژوهش تابع هدف عامل اول، حداقل کردن جمع وزنی زمان اتمام کارها با مجاز نبودن هیچ کار به تأخیر افتاده برای عامل دوم است. همچنین یک الگوریتم شاخه و کران و سه الگوریتم ابتکاری برای پیدا کردن جواب بهینه و نزدیک به بهینه پیشنهاد شده است. همچنین در [۴۲] یک مدل برنامه ریزی عدد صحیح مختلط برای مسأله زمان بندی جریان کارگاهی چندعاملی با زوال پذیری کارها ارائه شده است. همچنین با توجه به پیچیدگی مدل و عدم توانایی روش های حل دقیق در حل مسائل با ابعاد بزرگ، الگوریتم فراابتکاری ژنتیک مبتنی بر مرتب سازی نامغلوب^۱ (NSGA-II) پیشنهاد شده است.

لازم به ذکر است که مسأله زمان بندی جریان کارگاهی چندعاملی با فروض زوال پذیری کارها، زمان های آماده سازی وابسته به توالی و زمان های آزادسازی که برای برخی کاربردهای عملی مانند محیط های ساخت و تولید پیشرفته نیاز می باشد، تاکنون در هیچ تحقیقی در نظر گرفته نشده است. با تجزیه و تحلیل مطالعات انجام شده در زمینه زمان بندی جریان کارگاهی چندعاملی با فرض زوال پذیری کارها، موارد زیر را می توان بیان نمود:

(۱) مسأله جریان کارگاهی چندعاملی با در نظر گرفتن زوال پذیری کارها در تحقیقات بسیار کمی بررسی گردیده است. در حالی که مدل سازی این مسأله برای محیط های تولیدی و خدماتی امری ضروری می باشد.

(۲) طی مطالعات انجام شده این نتیجه حاصل شد که مسأله زمان بندی جریان کارگاهی چندعاملی با در نظر گرفتن زمان آماده سازی وابسته به توالی تاکنون مورد پژوهش قرار نگرفته است. البته لازم به ذکر است که زمان و تعداد دفعات آماده سازی ماشین ها در محیط جریان کارگاهی چندعاملی در مقاله [۴۳] در نظر گرفته شده است.

(۳) اکثر مطالعات موجود در زمینه زمان بندی جریان کارگاهی با فرض زوال پذیری حالات خاصی از مسأله مانند سیستمی با دو ماشین را در نظر گرفته و مدل نموده اند.

(۴) در بیشتر موارد، تابع هدف مربوط به یک عامل یا هر دو عامل محدود شده است. به بیان بهتر، هدف این مسائل، عبارت از حداقل کردن تابع هدف یکی از عامل ها با این شرط می باشد که تابع هدف عامل دیگر از یک حد بالا تجاوز نکند. در اکثر شرایط کاربردی و مسائل دنیای واقعی، تمامی مشتریان از اهمیت یکسانی برخوردار بوده و بهینه نمودن اهداف تنها یک مشتری معمول نمی باشد. بنابراین در این پژوهش، هدف به دست آوردن راه حل یا راه حل هایی است که در آن

است که در این مسأله، تصمیم‌گیرنده می‌تواند کارها را با پرداخت هزینه مربوط به تولید آن‌ها به شرکتی دیگر واگذار (رد) نماید. بنابراین زمان‌بندی جریان کارگاهی سه‌عاملی با سه فرض زوال‌پذیری کارها، زمان آماده‌سازی وابسته به توالی و زمان آزادسازی کارها برای مدل نمودن چنین شرایطی مناسب می‌باشد. هم‌چنین تصمیم‌گیرنده بایستی تمام کارها را به‌گونه‌ای زمان‌بندی نماید که انتظارات تمام مشتریان برآورده گردد. جهت کمک به تصمیم‌گیرنده جهت اخذ یک تصمیم مناسب، یک انتخاب خوب فراهم نمودن اطلاعات تصمیم‌گیری مناسب (مجموعه‌ای از راه‌حل‌های مناسب) برای وی است تا بتواند براساس اولویت‌های خود راه‌حل یا راه‌حلهایی را انتخاب نماید که توازن مناسبی بین اهداف تمام مشتریان برقرار نماید.

۳-۱. مدل ریاضی مسأله زمان‌بندی جریان کارگاهی سه‌عاملی با فرض زوال‌پذیری کارها، زمان آماده‌سازی وابسته به توالی و زمان آزادسازی کارها

– مفروضات

- توالی کارها روی ماشین‌های مختلف، یکسان است.
- تا زمانی که پردازش یک کار تمام نشده است، نمی‌توان آن را قطع کرد.
- در هر زمان، هر ماشین می‌تواند فقط یک کار را پردازش کند و یک کار نمی‌تواند هم‌زمان روی دو ماشین انجام شود.
- کارها در ابتدا همگی در دسترس نیستند.
- زمان آماده‌سازی وابسته به توالی است.
- تا زمانی که پردازش عملیات یک کار روی ماشین قبلی تمام نشود، عملیات بعدی آن نمی‌تواند شروع شود.
- برای تعریف بهتر مدل، یک کار مجازی \cdot تعریف می‌شود که مدت زمان پردازش آن روی همه ماشین‌ها صفر است و حتماً دارای یک کار پس‌نیازی خواهد بود.
- هر واحد تأخیر در کارهای عامل اول دارای هزینه است. یعنی برای مشتری تأخیر به قدری اهمیت دارد که تولید کننده به ازای هر واحد تأخیر هزینه‌ای لحاظ می‌کند.
- کارهای عامل دوم قابل رد شدن نیستند.
- در کارهای عامل سوم فقط تأخیر یا عدم آن اهمیت دارد و تعداد واحد تأخیر مهم نیست. بنابراین برای کارهای عامل سوم، هزینه هر واحد کار به تأخیر افتاده (صرف‌نظر از میزان این تأخیر) اهمیت دارد.

• اثر زمانی کارها به‌صورت رابطه زیر در مدل مسأله لحاظ می‌شود:

$$p'_{ji} = p_{ji} + \alpha \times ST_{ji} \quad (1)$$

که در آن ST_{ji} و p_{ji} به ترتیب زمان پردازش نرمال و زمان شروع کار i روی ماشین j بوده و α نرخ زوال‌پذیری کارها می‌باشد.

– اندیس و مجموعه‌های مدل

اندیس و مجموعه‌های مدل به‌صورت زیر تعریف می‌شوند:

$$Q \quad \text{مجموعه عامل‌ها:} \quad q = 1, 2, 3$$

$$j, k \quad \text{زیروندکارها:} \quad j = \{1, \dots, n\} \quad \text{و} \quad k = \{1, \dots, n\}$$

به‌صورت خلاصه مرور و نتایج به‌دست آمده توضیح داده شده‌اند. هم‌چنین در این بخش پیشنهادهاتی برای تحقیقات آتی بیان گردیده است.

۳. تعریف مسأله زمان‌بندی جریان کارگاهی سه‌عاملی با فرض زوال‌پذیری کارها، زمان آماده‌سازی وابسته به توالی و زمان آزادسازی کارها

در این پژوهش یک مسأله زمان‌بندی جریان کارگاهی سه‌عاملی با زوال‌پذیری کارها در نظر گرفته شده است که در آن زمان آماده‌سازی کارها وابسته به توالی است. هم‌چنین زمان دسترسی کارها با یکدیگر متفاوت بوده و در ابتدا همگی آزاد نیستند.

در مسأله مورد بررسی هر مشتری (عامل) هدف متفاوتی داشته و تصمیم‌گیرنده به‌دنبال حداقل نمودن اهداف هر سه مشتری با استفاده از زمان‌بندی کارهای آن‌ها می‌باشد. یک زمان‌بندی ایجادشده ترکیبی از n کار مربوط به هر 3 عامل می‌باشد که بایستی روی m ماشین پردازش گردند. از طرفی تأثیرگذاری کارهای سه مجموعه از کارها یکدیگر منجر به ایجاد تضاد بین اهداف عامل‌ها گردیده و بنابراین منطقی است که مسأله موردنظر به‌صورت یک مسأله بهینه‌سازی چندهدفه فرمول‌بندی شود. باتوجه به این توضیحات، هدف از این پژوهش یافتن یک مجموعه از راه‌حلهایی می‌باشد که توازن مناسبی را بین سه عامل ایجاد نماید.

در واقع مسأله در نظر گرفته شده به‌طور گسترده‌ای در سیستم‌های تولیدی دنیای واقعی وجود دارد. فرایند نورد داغ ورقه‌های فولادی برای مشتری‌های متفاوت (که هر یک اهداف مختلفی را مدنظر دارند) مثالی واقعی از مدل مورد بررسی در این پژوهش می‌باشد. عملیات مورد نیاز برای نورد ورقه‌های مختلف بایستی به‌صورت پشت سر هم انجام گردند. این عملیات شامل گرم کردن ورقه‌های فولادی اولیه در کوره، نورد ورقه‌های پیش‌گرم شده و یک یا چند مرحله عملیات حرارتی پس از نورد می‌باشد. سرد شدن ورقه‌های فولادی پیش‌گرم شده با گذشت زمان، موجب افزایش مدت زمان نورد این ورقه‌ها می‌گردد. قبل از انجام فرایند نورد، غلتک‌های نورد بسته به اندازه و ضخامت ورقه موردنظر بایستی تنظیم گردند. زمان تنظیم (آماده‌سازی) غلتک‌ها قبل از انجام فرایند نورد یک ورقه به‌خصوص بسته به ضخامت و ابعاد آن و هم‌چنین ورقه‌ای که بلافاصله قبل از این ورق فرایند نورد بر روی آن انجام شده است ممکن است متفاوت باشد. در صورتی که ورقه اول ضخیم و پهن بوده و ورقه دوم نازک و باریک باشد زمان زیادی جهت تنظیم و تعویض غلتک‌ها بایستی صرف شود. بالعکس در صورتی که ورقه اول نازک و باریک و ورقه دوم ضخیم و پهن باشد می‌توان با اضافه نمودن چندین غلتک مناسب به غلتک‌های قبلی فرایند نورد ورقه جدید را سریع‌تر آغاز نموده و از سرد شدن آن جلوگیری نمود. هم‌چنین کلاف ورقه‌های ورودی موردنیاز برای تأمین سفارشات مشتریان مختلف ممکن است در زمان‌های متفاوتی به شرکت وارد گردیده و جهت پیش‌گرم نمودن و در نتیجه انجام فرایند نورد در دسترس (آزاد) باشند. لازم به ذکر

رد کردن کارها برای عامل سوم.
- محدودیت های مدل

$$\sum_{k=0, j \neq k}^n x_{kj} = (1 - y_j) \quad \forall j \quad (3)$$

$$\sum_{j=1, k \neq j}^n x_{jk} \leq (1 - y_j) \quad \forall k \quad (4)$$

$$\sum_{j=1}^n x_{0j} = 1 \quad (5)$$

$$x_{kj} + x_{jk} = 1 \quad \forall j, k = \{1, 2, \dots, n-1\}, j > k \quad (6)$$

$$p'_{ji} = (p_{ji} + \alpha \times ST_{ji}) \times (1 - y_j) \quad \forall j, i \quad (7)$$

$$ST_{ji} \geq C_{ki} + (x_{kj} - 1)M + x_{kj} \times S_{kji} \quad (8)$$

$\forall j, k, i, j \neq k$

$$ST_{ji} \geq C_{j, i-1} + x_{kj} \times S_{kji} \quad \forall j, k, \forall i > 1 \quad (9)$$

$$C_{j1} \geq p'_{j1} + x_{kj} \times S_{kj1} + r_j \quad \forall j, k \quad (10)$$

$$C_{ji} \geq C_{ki} + p'_{ji} + (x_{kj} - 1)M + x_{kj} \times S_{kji} \quad \forall j, k, i \quad (11)$$

$$C_{ji} \geq C_{j, i-1} + p'_{ji} + x_{kj} \times S_{kji} \quad \forall j, k, \forall i > 1 \quad (12)$$

$$T_j \geq C_{jm} - d_j \quad \forall j \in Set_1 \quad (13)$$

$$C_{max} \geq C_{jm} \quad \forall j \in Set_2 \quad (14)$$

$$C_{ji} \leq d_j + M \times U_j \quad \forall j \in Set_3, i \quad (15)$$

$$y_j = 1 \quad \forall j \in Set_2 \quad (16)$$

$$C_{ji} \geq 0 \quad \forall j, i \quad (17)$$

$$T_j \geq 0 \quad \forall j \quad (18)$$

$$x_{kj} \in \{0, 1\} \quad \forall j, k, j \neq k \quad (19)$$

$$y_j \in \{0, 1\} \quad \forall j \quad (20)$$

رابطه (۳) تضمین می کند که اگر تصمیم به پردازش کار j گرفته شود، این کار حتماً باید یک کار پیش نیاز داشته باشد که این کار پیش نیاز می تواند کار مجازی ۰ با زمان صفر باشد. هم چنین اگر تصمیم بر رد کردن کار مذکور باشد، این کار هیچ پیش نیازی هم نخواهد داشت. رابطه (۴) محدودیت پس نیازی بوده و تضمین می کند که هر کاری که اگر تصمیم به پردازش کار j گرفته شود، حداکثر یک کار پس نیاز کار مذکور می باشد. هم چنین اگر تصمیم بر رد کردن آن

i	مجموعه ماشین ها:	$i = \{1, \dots, m\}$
Set_q	مجموعه کارهای عامل q	
- پارامترهای مدل		
m	تعداد کل ماشین ها	پارامترهای مدل به صورت زیر تعریف می شوند:
n	تعداد کل کارها	
p_{ji}	مدت زمان پردازش نرمال کار j روی ماشین i	
tc	هزینه هر واحد تأخیر	
uc	هزینه هر واحد کار به تأخیر افتاده	
M	یک عدد مثبت بزرگ	
d_j	زمان تحویل کار j	
e_j	هزینه واگذاری (رد کردن) کار j	
α	نرخ زوال پذیری کارها	
F_q	تابع هدف عامل q	
S_{kji}	زمان آماده سازی کار j بلافاصله پس از کار k روی ماشین i	
r_j	زمان آزادسازی کار j	

- متغیرهای تصمیم مدل
متغیرهای تصمیم مدل به صورت زیر تعریف می شوند:
 p'_{ji} متغیر پیوسته زمان پردازش واقعی کار j روی ماشین i
 ST_{ji} متغیر پیوسته زمان شروع کار j روی ماشین i
 C_{ji} متغیر پیوسته زمان اتمام کار j روی ماشین i
 T_j متغیر پیوسته مدت زمان تأخیر کار j
 C_{max} متغیر پیوسته زمان اتمام آخرین کار ($C_{max} = \max(C_{jm})$)
 U_j متغیر صفر و یک که در صورت تأخیر کار j مقدار آن یک بوده و در غیر این صورت صفر می باشد. $U_j = \begin{cases} 1 & C_{jm} > d_j \\ 0 & o.w \end{cases}$
 y_j متغیر صفر و یک که در صورت رد کردن کار j مقدار آن یک بوده و در غیر این صورت مقدار آن صفر می باشد.
 x_{kj} متغیر صفر و یک که در صورت پردازش کار j بلافاصله بعد از کار k مقدار آن یک بوده و در غیر این صورت صفر می باشد.
- توابع هدف مدل

توابع هدف مدل به صورت زیر تعریف می شوند:

$$\text{Min } (F_1, F_2, F_3)$$

$$F_1 = \sum_{j=1}^n (tc \times T_j + y_j \times e_j) \quad (2)$$

$$F_2 = C_{max} = \max_{j \in Set_2} (C_{jm})$$

$$F_3 = \sum_{j=n_2+1}^n (uc \times U_j + y_j \times e_j)$$

این توابع عبارتند از:
 F_1 : حداقل کردن مجموع هزینه تأخیر کارهای پردازش شده و میزان هزینه رد کردن کارها برای عامل اول.
 F_2 : حداقل کردن زمان اتمام آخرین کار برای عامل دوم.
 F_3 : حداقل کردن مجموع هزینه تعداد کارهای به تأخیر افتاده و هزینه

$$p_{ji} - p_{ji} \times y_j + \alpha \times ST_{ji} - \alpha \times ST_{ji} \times y_j \quad \forall j, i \quad (21)$$

$$= p_{ji} - p_{ji} \times y_j + \alpha \times ST_{ji} - \alpha \times ST_{ji} y_j$$

$$ST_{ji} \leq ST_{ji} \quad \forall j, i \quad (22)$$

$$ST_{ji} \geq ST_{ji} - M(1 - y_j) \quad \forall j, i \quad (23)$$

$$ST_{ji} \leq My_j \quad \forall j, i \quad (24)$$

اکنون به جای معادله (۷) روابط (۲۱) تا (۲۴) جایگزین می‌شود.

۴. روش‌های حل مسأله مورد بررسی

همان‌طور که در بخش قبل نشان داده شد، مدل برنامه‌ریزی ارائه شده در این مقاله، یک مدل چندهدفه می‌باشد. بهینه‌سازی چندهدفه، فرایند بهینه‌سازی هم‌زمان دو یا چند تابع هدف متضاد با در نظر گرفتن تعدادی محدودیت می‌باشد. از آنجایی که این مسائل به دلیل متعارض بودن یا غیرقابل مقایسه بودن اهداف به ندرت دارای جواب منحصر به فرد می‌باشند، تصمیم‌گیرنده بایستی جوابی را از میان جواب‌های نامغلوب^۱ انتخاب نماید. در یک جواب نامغلوب بهتر شدن مقدار یکی از توابع هدف در ازای بدتر شدن مقدار حداقل یکی از توابع دیگر حاصل می‌گردد. در واقع جبهه پارتو^۲ شامل مجموعه‌ای از جواب‌های نامغلوب در فضای جواب می‌باشد. با توجه به این توضیحات جهت حل مسأله مورد بررسی، از دو روش محدودیت اِپسیلون^۳ تعمیم‌یافته^۳ و الگوریتم ازدحام ذرات چندهدفه استفاده گردیده است که در ادامه توضیح داده می‌شوند.

۴-۱. روش حل دقیق محدودیت اِپسیلون تقویت شده

روش محدودیت اِپسیلون یکی از رویکردهای شناخته شده برای مواجهه با مسائل چندهدفه است، که در آن یکی از توابع هدف با انتقال سایر توابع هدف به محدودیت‌ها بهینه می‌گردد. این روش مزیت‌های بسیاری نسبت به سایر روش‌های مواجهه با مسائل چندهدفه دارد. این روش اولین بار در سال ۱۹۷۱ توسط هایمس و همکاران ارائه شده است. رویکرد این روش به گونه‌ای است که مسأله چندهدفه را به مسأله تک‌هدفه تبدیل می‌نماید. به این صورت که یکی از توابع هدف موجود انتخاب و حداقل‌سازی شده و سایر توابع هدف (که به صورت حداقل‌سازی می‌باشند) به محدودیت‌هایی با حد بالا تبدیل می‌شوند. این روش، مجموعه‌ای از جواب‌های نامغلوب را برای تصمیم‌گیرنده ایجاد می‌نماید تا بتوان با توجه به شرایط، یک یا چندتا از این جواب‌ها را انتخاب نمود. مدل کلی این روش برای پژوهش حاضر با فرض در نظر گرفتن تابع هدف اول به عنوان تابع هدف اصلی به صورت زیر تعریف می‌گردد:

$$\begin{aligned} & \text{Min } F_1(x) \\ & \text{st : } g(x) \leq 0 \\ & F_2(x) \leq \varepsilon_2 \\ & F_3(x) \leq \varepsilon_3 \end{aligned} \quad (25)$$

باشد، این کار هیچ پس‌نیازی نخواهد داشت. معادله (۵) تأکید دارد که کار مجازی ۰ حتماً باید یک پس‌نیاز داشته باشد. براساس معادله (۶) یک کار نمی‌تواند هم‌زمان هم پیش‌نیاز و هم پس‌نیاز کار دیگری باشد. معادله (۷)، روش محاسبه مدت زمان پردازش واقعی یک کار را بیان می‌کند. رابطه (۸) بیانگر آن است که زمان شروع یک کار به خصوص به جز کار مجازی ۰ روی ماشین‌ها باید بزرگتر از زمان اتمام کار پیش‌نیاز این کار روی همان ماشین به علاوه زمان آماده‌سازی ماشین از کار پیش‌نیاز به کار موردنظر باشد. هم‌چنین در رابطه (۹) زمان شروع یک کار به خصوص روی ماشین‌ها به جز ماشین اول باید بزرگتر از زمان اتمام عملیات همان کار روی ماشین قبلی به علاوه زمان آماده‌سازی ماشین از کار قبلی به کار موردنظر باشد. زمان آماده‌سازی با متغیر x_{kj} لحاظ گردیده است.

رابطه (۱۰) تضمین می‌کند که زمان تکمیل هر کار در ماشین اول باید از مدت زمان پردازش واقعی آن کار روی ماشین اول به علاوه زمان آماده‌سازی از کار بلافاصله قبل از آن و زمان آزادسازی کار مذکور بیشتر باشد. با استفاده از رابطه (۱۱) این اطمینان حاصل می‌شود که زمان اتمام هر کار روی هر ماشین باید از زمان اتمام کار پیش‌نیازی آن به اضافه زمان پردازش واقعی کار مذکور و زمان آماده‌سازی از کار پیش‌نیاز به کار مذکور بیشتر باشد. رابطه (۱۲) تضمین می‌نماید که هر یک از عملیات مورد نیاز کار j در صورتی می‌تواند آغاز گردد که عملیات قبلی آن تمام شده باشد. رابطه (۱۳) مدت زما تأخیر کارهای عامل اول را حساب می‌کند. متغیر C_{max} عامل دوم با استفاده از رابطه (۱۴) تعریف می‌گردد. هم‌چنین رابطه (۱۵) متغیر U_j را برای عامل سوم محاسبه می‌نماید. در صورتی که زمان اتمام کار j متعلق به عامل سوم کمتر از زمان تحویل کار موردنظر باشد، این کار تأخیر نداشته و در نتیجه $C_{jm} < d_j$ می‌باشد. از طرفی از آنجاکه با توجه به تابع هدف F_3 برای عامل سوم، هدف حداقل‌سازی هزینه تعداد کارهای به تأخیر افتاده می‌باشد، بنابراین در شرایطی که کار j متعلق به عامل سوم تأخیر نداشته باشد، متغیر U_j حتماً مقدار ۰ گرفته و مقدار ۱ نمی‌گیرد. براساس رابطه (۱۶) کارهای عامل دوم قابل رد شدن نبوده و بایستی همگی انجام گردند. روابط (۱۷) تا (۲۰) جهت تعریف متغیرهای تصمیم مسأله ایجاد گردیده‌اند.

لازم به ذکر است که در مدل ارائه شده در بخش قبلی معادله (۷) به دلیل وجود ضرب یک متغیر صفر و یک (y_j) با یک متغیر پیوسته (ST_{ji}) غیرخطی بوده و باید خطی‌سازی گردد. طبق آنچه که در کتاب چن و همکاران گفته شده است [۴۴]، این رابطه غیرخطی به این صورت خطی‌سازی می‌گردد که به جای ضرب دو متغیر ($ST_{ji} \times y_j$) در معادله (۷) یک متغیر جدید $ST_{ji} y_j$ قرار گرفته و سه معادله جدید نیز به مدل اضافه می‌گردند. این چهار معادله به صورت زیر می‌باشند:

$$p'_{ji} = (p_{ji} + \alpha \times ST_{ji}) \times (1 - y_j) =$$

$$\text{Min } (F_1(x) - \rho \times (\frac{s_2}{r_2} + \frac{s_3}{r_3}))$$

$$\text{st: } g(x) \leq 0 \quad (26)$$

$$F_2(x) + s_2 = \varepsilon_2$$

$$F_3(x) + s_3 = \varepsilon_3$$

که در آن ρ مقداری بسیار کوچک (معمولاً بین 10^{-6} تا 10^{-3}) بوده و r_i دامنه i امین تابع هدف می‌باشد که با استفاده از جدول نتایج نهایی تعیین گردیده است.

۴-۲. الگوریتم MOPSO

از آنجاکه مسأله موردنظر در این مقاله یک مسأله سخت و چندهدفه می‌باشد، یک الگوریتم فراابتکاری ازدحام ذرات چندهدفه جهت ایجاد مجموعه‌ای از بهترین جواب‌های ممکن با در نظر گرفتن اهداف هر سه عامل توسعه داده شده است. در ابتدا توضیح کلی در خصوص الگوریتم ازدحام ذرات و الگوریتم MOPSO و تفاوت آن‌ها ارائه گردیده و سپس ساختار و اجزای الگوریتم MOPSO ارائه شده جهت حل مسأله زمان‌بندی جریان کارگاهی سه‌عاملی با در نظر گرفتن زوال‌پذیری کارها، زمان آماده‌سازی وابسته به توالی و زمان آزادسازی کارها توضیح داده خواهد شد.

۴-۲-۱. الگوریتم‌های PSO, MOPSO

PSO یک الگوریتم تکاملی مبتنی بر جمعیت می‌باشد که در آن جمعیت جواب‌ها تحت عنوان توده یا ازدحام شناخته می‌شود. این روش در سال ۱۹۹۵ توسط جیمز کندی و راسل ابرهارت معرفی گردید [۴۶]. در PSO، ذرات در فضای جست‌وجو جاری می‌شوند. تغییر مکان ذرات در فضای جست‌وجو تحت تأثیر تجربه و دانش خودشان و همسایگان‌شان می‌باشد. بنابراین موقعیت دیگر توده ذرات روی چگونگی جست‌وجوی یک ذره اثر می‌گذارد. نتیجه مدل‌سازی این رفتار اجتماعی فرآیند جست‌وجوی است که ذرات به سمت نواحی موفق میل می‌کنند. در هر لحظه t هر ذره i سرعت بعدی خود $(v_i(t+1))$ را در فضای جست‌وجو با توجه به سرعت فعلی اش $(v_i(t))$ ، مکان فعلی اش $(x_i(t))$ ، بهترین مکانی که تاکنون در آن قرار گرفته است (p_{best}) و بهترین مکانی که در کل همسایگی اش وجود دارد (g_{best}) ، تنظیم می‌کند. همچنین هر ذره موقعیت بعدی خود $(x_i(t+1))$ را با توجه به محل فعلی و سرعت بعدی خود محاسبه می‌کند. با توجه به توضیحات ذکر شده سرعت و مکان جدید هر ذره با استفاده از معادلات (۲۷) و (۲۸) محاسبه می‌گردد.

$$v_i(t+1) = wv_i(t) + c_1r_1(t)[p_{best} - x_i(t)] + c_2r_2(t)[g_{best} - x_i(t)] \quad (27)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (28)$$

که در آن w ضریب اینرسی نامیده می‌شود. چون ضریبی است که جهت حرکت کنونی را به جهت حرکت بعدی مرتبط می‌گرداند. به عبارت دیگر این ضریب نشانگر میزان تمایل ذره برای حفظ حالت حرکت کنونی خود بوده و برای همگرایی الگوریتم حیاتی می‌باشد. c_1 ، c_2 به ترتیب عبارت از ضریب شتاب شخصی c_1 و ضریب شتاب

که در آن $g(x)$ شامل همه محدودیت‌های اصلی مسأله بوده و ε_2 و ε_3 حدود بالای به دست آمده برای توابع هدف دوم و سوم هستند. با تغییر پارامترهای ε_2 و ε_3 راه‌حل‌های مناسبی برای مسأله ایجاد می‌گردند. لازم به ذکر است که این روش دارای ایرادات زیر می‌باشد:

- مشکل بودن تشخیص و تعیین دامنه تغییرات مناسب برای پارامترهای ε_2 و ε_3 . دلیل این امر این است که بهترین مقدار برای این پارامترها را می‌توان به آسانی و با در نظر گرفتن هر یک از اهداف دوم و سوم به عنوان تنها تابع هدف مسأله محاسبه نمود. اما محاسبه بدترین مقدار این پارامترها کار مشکلی می‌باشد. معمول‌ترین روش برای محاسبه این مقادیر استفاده از جدول نتایج نهایی^۱ می‌باشد. بدین صورت که بیشترین مقدار مربوط به ستون تابع هدف مربوطه (هریک از توابع هدف ۲ و ۳) به عنوان تخمینی برای بدترین مقدار پارامترهای ε_2 و ε_3 استفاده می‌گردد.

۲- عدم وجود تضمینی برای کارا و مناسب بودن جواب‌های ایجاد شده توسط این روش.

۳- افزایش زمان حل مسأله در صورت وجود بیش از دو تابع هدف. به منظور غلبه بر ایرادات نام‌برده، روش محدودیت افسیلون تقویت شده معرفی گردیده است [۴۵]. در این روش، جهت محاسبه مقادیر جدول نتایج نهایی برای هر تابع هدف از روش لکسیکوگراف استفاده می‌گردد. یک راه‌حل ساده جهت رفع مشکل تخمین بدترین مقادیر پارامترهای ε_2 و ε_3 ، تعریف مقادیر رزرو برای توابع هدف می‌باشد. مقادیر رزرو مانند یک حد پایین برای توابع هدف عمل نموده و مقادیر بدتر از مقادیر رزرو مجاز نمی‌باشند. روش لکسیکوگراف بدین صورت عمل می‌نماید که ابتدا اولین تابع هدف به تنهایی در نظر گرفته شده و مقدار بهینه $F_1 = z_1^*$ برای آن ایجاد می‌گردد. سپس تابع هدف دوم با اضافه نمودن محدودیت $F_1 = z_1^*$ به منظور حفظ جواب بهینه حاصل از حل تابع هدف اول در نظر گرفته شده و حل می‌گردد. با فرض دستیابی به مقدار بهینه $F_2 = z_2^*$ ، این بار دو محدودیت z_1^* و $F_1 = z_2^*$ به همراه تابع هدف سوم به عنوان تابع هدف اصلی در نظر گرفته می‌شوند تا دو جواب بهینه قبلی حفظ گردند. با این کار جواب بهینه یافت شده در واقع یک جواب کارآمد نیز می‌باشد زیرا در ایجاد آن محدودیت مربوط به دو تابع هدف دیگر نیز در نظر گرفته شده است. در شرایطی که چندین جواب مختلف برای مسأله وجود داشته باشد، به منظور عملکرد بهتر روش، محدودیت‌های مربوط به توابع هدف فرعی با قرار دادن متغیرهای کمکی به تساوی تبدیل می‌گردد. همچنین مقادیر نرمال‌سازی شده متغیرهای مازاد (متغیرهای s_2 ، s_3) به عنوان عبارت دوم در تابع هدف در نظر گرفته می‌شوند. در این صورت مدل کلی روش محدودیت افسیلون تقویت شده برای پژوهش حاضر با فرض در نظر گرفتن تابع هدف اول به عنوان تابع هدف اصلی به صورت زیر تعریف می‌گردد:

جمعی^۱ می‌باشند. هم‌چنین r_1 ، r_2 دو مقدار تصادفی در بازه صفر و یک هستند.

در الگوریتم PSO، ابتدا سیستم با استفاده از جواب‌های ایجاد شده به صورت تصادفی شروع به کار نموده و هر جواب (ذره) در فضای جست‌وجو حرکت می‌نماید. سپس برای هر ذره، مقدار تابع هدف محاسبه گردیده و مقادیر p_i و v_i به‌روزرسانی می‌گردند. سپس سرعت و مکان هر ذره با استفاده از مقدار تابع هدف به‌روزرسانی می‌گردند. در نهایت هر ذره باتوجه به سرعت و مکان فعلی جدیدش هدایت می‌گردد تا نتیجه بهتری در فضای جست‌وجو حاصل شود. این فرایند تا زمان برقراری شرط توقف ادامه می‌یابد.

محدودیت اصلی الگوریتم PSO این است که این الگوریتم برای مسائل بهینه‌سازی با یک تابع هدف طراحی گردیده است. به‌منظور رفع این محدودیت، الگوریتم MOPSO در سال ۲۰۰۲ توسط کوئلو و لچوگا ارائه گردید [۴۷] که در شرایطی که بهینه‌سازی چند تابع هدف متناقض به‌طور هم‌زمان مدنظر باشد مورد استفاده قرار می‌گیرد. در این الگوریتم از یک آرشو خارجی ذرات (A_i) جهت ذخیره نمودن جواب‌های نامغلوب یافت شده توسط الگوریتم در حین فرایند جست‌وجو تا t امین تکرار استفاده می‌گردد. این آرشو برای هدایت روند جست‌وجوی سایر ذرات و هم‌چنین ساختن جبهه پارتو مورد استفاده قرار می‌گیرد. معادلات به‌روزرسانی سرعت و مکان مشابه معادلات (۲۷) و (۲۸) می‌باشد اما با این تفاوت که تمامی اهداف جهت تعیین v_i و r_i استفاده می‌گردند. بهترین موقعیت مشاهده شده توسط یک ذره از ابتدای جست‌وجو تحت عنوان رهبر شخصی^۲ ذره نام‌گذاری شده و رهبر جمعی^۳ ذره نیز عبارت از بهترین موقعیت یافت شده توسط جواب‌هایی می‌باشد که در همسایگی جواب موردنظر قرار دارند. لازم به ذکر است که رهبر جمعی از بین جواب‌های موجود در آرشو خارجی ذرات و براساس یک معیار از پیش‌تعیین شده مانند پراکندگی تعیین می‌گردد. روش‌ها و معیارهای مختلفی برای انتخاب رهبران و کنترل اندازه آرشو خارجی در الگوریتم MOPSO معرفی شده‌اند. پس از تعیین یک رهبر شخصی و یک رهبر جمعی برای هر ذره، سرعت هر ذره در هر بعد (تابع هدف) به‌روزرسانی گردیده و مقدار سرعت جدید جهت به‌روزرسانی مکان ذره استفاده می‌گردد. الگوریتم، جواب‌های نامغلوب ایجاد شده را با ثبت بهترین جواب یافت شده توسط هر ذره از ابتدای جست‌وجو تاکنون ذخیره می‌نماید. در صورتی که تعداد جواب‌های نامغلوب موجود در آرشو به‌صورت خودکار هرس می‌گردد. به‌طور کلی هر دو الگوریتم PSO و MOPSO شامل سه مرحله زیر می‌باشند [۴۸]:

۱- مرحله ابتدایی که در آن مجموعه‌ای از ذرات ایجاد گردیده و توده یا ازدحام را تشکیل می‌دهند.

۲- مرحله ارزیابی جهت یافتن، به‌روزرسانی و انتخاب رهبران

شخصی و جمعی برای هر ذره.

۳- مرحله به‌روزرسانی جهت به‌هنگام نمودن سرعت و مکان ذرات در هر بار تکرار الگوریتم.

۴-۲-۲. ساختار الگوریتم MOPSO توسعه داده شده

الگوریتم MOPSO توسعه داده شده، همان‌طور که در شکل (۱) نشان داده شده است با ایجاد N ذره هر یک با موقعیت و سرعت مشخص به صورت تصادفی آغاز می‌گردد (خط ۱). هر ذره در واقع بیانگر رد نمودن یا انجام کار در شرکت موردنظر و توالی پردازش کارهای رد نشده روی ماشین‌ها می‌باشد. سپس هر توالی با استفاده از سه هدف موجود در معادله (۲) ارزیابی می‌گردد (خط ۲). آرشو خارجی در ابتدای جست‌وجو (در زمان $t=0$) تهی می‌باشد (خط ۴). حلقه تکرار الگوریتم تا زمان برقراری شرط توقف اجرا می‌گردد. این حلقه شامل به‌روزرسانی آرشو خارجی (خط ۶)، جدول‌بندی فضای هدف کشف شده (خط ۷) و انتخاب رهبران شخصی و جمعی از بین آرشو خارجی می‌باشد تا ذرات در فضای جست‌وجو حرکت نمایند (خط ۸). جدول‌بندی فضای هدف بدین معنی است که باتوجه به حداقل و حداکثر میزان توابع هدف به‌دست آمده، هر یک از این بازه‌ها برحسب نیاز به nGrid قسمت مساوی تقسیم می‌گردد. سپس ذرات مختلف برحسب مقادیر توابع هدف مختلف آن‌ها در یکی از خانه‌های جدول ایجاد شده قرار می‌گیرند.

در روش مورد استفاده جهت انتخاب رهبر، یکی از ذرات موجود در یک جدول به‌خصوص از فضا با استفاده از روش انتخاب چرخ رولت^۴ انتخاب می‌گردد. احتمال انتخاب یک ذره به‌خصوص i به‌عنوان رهبر (یعنی p_i) با تعداد ذرات موجود در جدول آن ذره (یعنی n_i) رابطه عکس دارد. جهت محاسبه مقدار احتمال انتخاب ذره i روش Boltzman به‌کار گرفته شده‌است. براساس این روش مقدار p_i با استفاده از رابطه زیر محاسبه گردد:

$$p_i = \frac{e^{-\beta n_i}}{\sum_j e^{-\beta n_j}} \quad (29)$$

که در آن پارامتر β عبارت از فشار انتخاب یک ذره به‌خصوص می‌باشد. هم‌چنین لازم به ذکر است که نحوه عملکرد روش مورد استفاده جهت هرس کردن آرشو خارجی کاملاً مشابه روش انتخاب رهبر می‌باشد با این تفاوت که در رابطه (۲۹) به‌جای پارامتر β ، پارامتر γ قرار می‌گیرد که بیانگر فشار حذف یک ذره به‌خصوص می‌باشد. پس از انجام مراحل فوق سرعت و مکان جدید ذرات محاسبه گردیده (خطوط ۹ و ۱۰) و عملگر جهش جهت بهبود پراکندگی جواب‌ها و جلوگیری از همگرایی زودرس الگوریتم با یک احتمال مشخص بر روی جواب‌ها اعمال گردیده (خط ۱۱) و جواب‌های جدید ارزیابی می‌گردند (خط ۱۲). پس از آن بهترین مکانی که هر یک از ذرات تاکنون در آن قرار گرفته‌اند به‌روزرسانی می‌گردد (خط ۱۳). پس

3. Social leader

4. Roulette Wheel Selection

1. Social acceleration coefficient

2. Cognitive leader

۴ رد می گردد. زیرا عدد موجود در خانه ۴ام از ماتریس برابر با صفر می باشد. همچنین برای سایر کارها توالی به دست آمده به صورت ۶، ۵، ۳، ۱ و ۲ خواهد بود. زیرا عدد متناظر با کار ۱ در ماتریس نشان داده شده، بزرگترین عدد در میان ۵ عدد بزرگتر از صفر است و به همین دلیل به عنوان اولین کار پردازش خواهد شد. به همین ترتیب توالی سایر کارها براساس مقدار عدد متناظر در ماتریس شکل (۲) به صورت غیرنرولی مشخص می گردد.

۰/۸۳	۰/۹۵	۰/۴۱	۰	۰/۲۸	۰/۱۳
------	------	------	---	------	------

شکل (۲): نحوه نمایش جواب در الگوریتم MOPSO

۴-۲-۴. عملگرهای جهش مورد استفاده در الگوریتم MOPSO توسعه داده شده

در این پژوهش، از ۴ عملگر جهش زیر با شانس مساوی برای انتخاب استفاده گردیده است:

- عملگر جابه جایی^۱: در این عملگر، دو خانه از ماتریس موقعیت ذره به تصادف انتخاب گردیده و مقادیر موجود در این دو خانه با هم جابه جا می شوند.
- عملگر وارون سازی^۲: در این عملگر، دو خانه از ماتریس موقعیت ذره به صورت تصادفی انتخاب گردیده (به عنوان مثال خانه های a_1 و a_2) و مقادیر خانه های بین دو خانه انتخابی با یکدیگر جابه جا می گردند. یعنی مقدار موجود در خانه $a_1 + 1$ با مقدار موجود در خانه $a_2 - 1$ مقدار موجود در خانه $a_1 + 2$ با مقدار موجود در خانه $a_2 - 2$ و ... جابه جا می گردند.
- عملگر جای گذاری^۳: این عملگر به این صورت عمل می نماید که دو خانه از ماتریس موقعیت ذره به تصادف انتخاب گردیده و مقدار موجود در خانه اول (خانه a_1) به یک خانه بلافاصله قبل از خانه دوم (خانه $a_2 - 1$) منتقل می گردد. همچنین مقادیر موجود در خانه های $a_1 + 1$ تا $a_2 - 2$ و مقدار قبلی موجود در خانه $a_2 - 1$ هر کدام به یک خانه عقب تر منتقل می گردند.
- عملگر رد کردن^۴: در این عملگر، دو خانه از ماتریس موقعیت که مقادیر آنها مخالف صفر بوده و کارهای متناظر با آنها مربوط به مجموعه کارهای عامل دوم نباشند، به تصادف انتخاب گردیده و مقدار آنها برابر با ۰ در نظر گرفته می شود. کارهای متناسب با خانه های دارای مقدار صفر رد می گردند. دلیل عدم انتخاب مجموعه کارهای عامل دوم این است که این کارها قابل رد شدن نبوده و بایستی همگی انجام گردند.

۵. نتایج محاسباتی

در این مقاله به منظور ارزیابی عملکرد الگوریتم ازدحام ذرات چندهدفه، نتایج حاصل از اجرای این الگوریتم برای حل مثال های عددی ایجاد شده به صورت تصادفی مطابق جدول (۱) با نتایج حاصل از اجرای روش

از اجرای حلقه تکرار به تعداد t_{max} مرتبه، الگوریتم آرشیو خارجی را به عنوان تخمینی برای جبهه پارتو برمی گرداند (خط ۱۵).

الگوریتم ۱- شبه کد الگوریتم MOPSO توسعه داده شده

ورودی: اطلاعات مربوط به کارها و ماشین ها

خروجی: جواب های نامغلوب مربوط به زمان بندی کارها روی ماشین ها

۱- ایجاد جمعیتی از ذرات با اندازه N به صورت تصادفی

۲- ارزیابی ذرات با استفاده از توابع هدف معادله (۲)

۳- $t = 0$

۴- مقداردهی اولیه به آرشیو خارجی A_0

۵- تا زمانی که $t < t_{max}$ مراحل زیر انجام می گردد:

۶- به روزرسانی آرشیو خارجی A_t شامل ذرات نامغلوب

مجموعه U_{S_t-1} و A_{t-1} و هرس کردن آن با استفاده از الگوریتم

Boltzmann در صورت لزوم

۷- جدول بندی فضای هدف کشف شده

۸- انتخاب یک رهبر شخصی و یک رهبر جمعی از بین

جواب های موجود در A_t برای هر ذره با استفاده از روش

Boltzmann

۹- محاسبه سرعت ذرات با استفاده از معادله (۲۷)

۱۰- به روزرسانی مکان ذرات با استفاده از معادله (۲۸)

۱۱- اعمال عملگر جهش

۱۲- ارزیابی ذرات

۱۳- به روزرسانی مقدار y_i برای هر یک از ذرات

۱۴- اضافه کردن یک واحد به شمارنده تکرار t

۱۵- برگرداندن $A_{t_{max}}$ شامل بهترین زمان بندی های نامغلوب

شکل (۱): شبه کد الگوریتم MOPSO توسعه داده شده

۴-۲-۳. نحوه نمایش جواب در الگوریتم MOPSO توسعه داده شده

یکی از مؤلفه های مهم در الگوریتم های فراابتکاری نحوه نمایش جواب می باشد. ساختار نمایش جواب، در مسائل مختلف، متفاوت بوده و متناسب با ویژگی های مسأله مورد بررسی، تعریف می گردد. برای مسأله مورد بررسی در این پژوهش، نحوه نمایش جواب بدین صورت است که برای هر ذره یک ماتریس سطری به اندازه تعداد کارها در نظر گرفته شده و هر خانه نشان دهنده یک کار در توالی موردنظر است. به دلیل این که در این مسأله، فرض رد کردن کارها در نظر گرفته شده، در هر خانه، عددی تصادفی بین صفر و یک تولید می شود که اگر مقدار آن صفر باشد، به این معنی است که کار مربوط به آن رد شده است و در ترتیب قرار نمی گیرد. برای کارهایی که قرار است پردازش شوند، ترتیب قرار گرفتن در توالی به صورت کوچکترین عدد بزرگتر از صفر تا بزرگترین عدد داخل ماتریس است. به عنوان مثال با فرض این که ماتریس نشان داده شده در شکل (۲) یک موقعیت برای یک ذره به خصوص باشد، می توان نتیجه گرفت که در زمان بندی ایجاد شده کار

نیز شناخته می‌شود.

۳- میانگین فاصله از جواب ایده‌آل $(MID)^3$:

این معیار میانگین فاصله و میزان نزدیکی جواب‌های پارتویی را به نقطه ایده‌آل $(f_1^{best}, f_2^{best}, f_3^{best})$ ارزیابی نموده و با استفاده از رابطه زیر محاسبه می‌گردد:

$$MID = \frac{\sum_{i=1}^n \sqrt{\left(\frac{f_{1i} - f_1^{best}}{f_{1max} - f_{1min}}\right)^2 + \left(\frac{f_{2i} - f_2^{best}}{f_{2max} - f_{2min}}\right)^2 + \left(\frac{f_{3i} - f_3^{best}}{f_{3max} - f_{3min}}\right)^2}}{n} \quad (30)$$

که در آن n بیانگر تعداد جواب‌های نامغلوب بوده و همچنین مقادیر f_j^{max} و f_j^{min} به ترتیب بیانگر بیشترین و کمترین مقدار تابع هدف f_j در بین مجموعه جواب‌های نامغلوب یافت شده توسط تمامی الگوریتم‌های مورد مقایسه می‌باشند. همچنین f_{ji} بیانگر مقدار تابع هدف f_j برای i امین جواب نامغلوب از یک الگوریتم به خصوص می‌باشد. باتوجه به رابطه (۳۰) مقدار MID کمتر، نشان‌دهنده همگرایی و نزدیکی بیشتر جواب‌ها به نقطه ایده‌آل و در نتیجه عملکرد بهتر یک الگوریتم به خصوص می‌باشد.

۴- معیار پراکندگی $(DM)^4$:

این معیار مقدار پراکندگی مجموعه جواب‌های پارتویی را نشان داده و با استفاده از رابطه (۳۱) تعیین می‌گردد. در واقع این معیار، طول قطر مکعب مربوط به فضایی را اندازه‌گیری می‌کند که توسط مقادیر انتهایی اهداف برای مجموعه جواب‌های نامغلوب یک الگوریتم به خصوص ایجاد گردیده‌اند. بنابه این مفهوم، بیشتر بودن مقدار معیار DM برای یک الگوریتم مورد بررسی بیانگر این است که جواب‌های نامغلوب یافته شده توسط الگوریتم محدوده بیشتری را پوشش داده و در نتیجه الگوریتم موردنظر از عملکرد بهتری برخوردار می‌باشد.

۵- معیار نرخ نقاط نامغلوب به نقاط جبهه پارتو بهینه $(ONVGR)$: این معیار با استفاده از رابطه (۳۲) تعیین می‌گردد.

$$ONVGR = \frac{NPS_{know}}{NPS_{true}} \quad (32)$$

که در آن NPS_{know} بیانگر تعداد جواب‌های نامغلوب به دست آمده توسط الگوریتم موردنظر بوده و NPS_{true} بیانگر تعداد نقاط موجود در جبهه پارتوی بهینه ایجاد شده توسط تمامی الگوریتم‌ها می‌باشد.

۶- فاصله عمومی $(GD)^5$:

این معیار با استفاده از رابطه (۳۳) تعیین می‌گردد که در آن d_i^2 بیانگر فاصله اقلیدسی i امین جواب نامغلوب یافت شده توسط الگوریتم موردنظر تا نزدیکترین جواب واقع بر جبهه پارتو بهینه می‌باشد.

$$GD = \frac{\sqrt{\sum_i d_i^2}}{NPS_{know}} \quad (33)$$

محدودیت اپسیلون تقویت شده و الگوریتم مبتنی بر مرتب‌سازی نامغلوب [۴۲] مقایسه می‌گردد. برنامه الگوریتم‌ها با استفاده از نرم‌افزار Matlab نوشته شده و بر روی سیستمی با حافظه Intel Pentium® 2.45 GB و رم IV 2.8 GHz هم‌چنین روش محدودیت اپسیلون تقویت شده با استفاده از نرم‌افزار بهینه‌سازی GAMS نوشته شده و اجرا گردیده است.

جدول (۱): داده‌های مسأله

پارامتر	تعریف	مقدار
M	مقدار بزرگ	۱۰۰۰۰
α	نرخ اثر زمانی	۰/۱
a_{ji}	زمان پردازش نرمال کار j روی ماشین i	$Unifrand(1, 10)$
d_j	زمان تحویل کار j	$Unifrand(15, 30)$
e_j	میزان جریمه رد کردن کار j	$Unifrand(1, 5)$
r_j	زمان آزادسازی کار j	$Unifrand(0, 3)$
S_{kji}	زمان آماده‌سازی کار j بلافاصله پس از کار k روی ماشین i	$Unifrand(2, 7)$
tc	هزینه هر واحد تأخیر	۲
uc	هزینه هر واحد کار به تأخیر افتاده	۳

۵-۱. تنظیم پارامترهای الگوریتم MOPSO

از آنجاکه خروجی الگوریتم‌های فراابتکاری به شدت به مقادیر ورودی آن‌ها وابسته است، در این مقاله برای تنظیم پارامترهای الگوریتم MOPSO از روش تاگوچی استفاده گردیده است. ۸ پارامتر قابل تنظیم در این الگوریتم عبارت از تعداد تکرار الگوریتم (t_{max}) ، اندازه جمعیت (N) ، ضریب اینرسی (w) ، ضریب شتاب شخصی (C_1) ، ضریب شتاب جمعی (C_2) ، فشار انتخاب (β) ، فشار حذف (γ) و تعداد جدول بندی فضای هدف (nGrid) می‌باشند که برای هر یک از این پارامترها، ۳ سطح از مقادیر مختلف تعریف گردیدند (جدول (۲)). بنابراین تعداد حالات ایجاد شده عبارت از 3^8 یا ۶۵۶۱ حالت می‌باشد. روش تاگوچی برای یافتن بهترین مقادیر این پارامترها، این حالت‌ها را به ۲۷ حالت کاهش می‌دهد. باتوجه به این که در مسائل چندهدفه یک جواب بهینه به‌تنهایی وجود نداشته و مجموعه‌ای از جواب‌های نامغلوب به‌عنوان بهینه در نظر گرفته می‌شوند، لذا در این مسائل از چندین معیار ارزیابی جهت تنظیم پارامترها و مقایسه کارایی الگوریتم‌ها استفاده می‌گردد. معیارهای عملکردی در نظر گرفته شده در این پژوهش عبارتند از:

۱- زمان حل: به‌علت مهم بودن زمان حل الگوریتم‌ها در مسائل زمان‌بندی، زمان نیز به‌عنوان معیار ارزیابی عملکرد در نظر گرفته می‌شود.

۲- معیار تعداد نقاط نامغلوب $(NPS)^1$:

الگوریتمی که بتواند تعداد جواب‌های نامغلوب بیشتری را حاصل نماید، از عملکرد بهتری برخوردار خواهد بود. این معیار با عنوان $ONVG^2$

4. Diversification Metric
5. General Distance

1. Number of Pareto Solutions
2. Overall Non-Dominated Vector Generation
3. Mean Ideal Distance

پس از اجرای حالت های پیشنهادی، مقادیر معیارها نرمال سازی شده و جمع وزنی آن ها به نرم افزار Minitab داده می شود. در نهایت پس از تحلیل این داده ها براساس نسبت سیگنال به نویز^۲، نرم افزار مینی تب بهترین حالت را از بین ۶۵۶۱ حالت اعلام نمود. خروجی های این نرم افزار در شکل (۳) نشان داده شده است. در تحلیل نسبت سیگنال به نویز، با استفاده از تغییر نتایج، بهترین و قوی ترین شرایط کاری تعیین می گردند. به عبارت دیگر، این نسبت پراکندگی در اطراف یک مقدار مشخص را بیان می کند. برای مسأله مورد بررسی در این مقاله با توابع هدف کمینه سازی هر چه این نسبت بیشتر باشد، پراکندگی کمتر بوده و در نتیجه اثر پارامتر مربوطه مهم تر می باشد. با توجه به این توضیحات و براساس مقادیر نشان داده شده در شکل (۳) بهترین مقادیر برای پارامترهای الگوریتم MOPSO تعیین گردیده و در جدول (۳) نشان داده شده اند.

$$d_i = \max \left\{ 0, \min \left\{ (f_1^i(x) - f_1^j(x)) + (f_2^i(x) - f_2^j(x)) + (f_3^i(x) - f_3^j(x)) \right\} \right\} \quad (34)$$

۷- فاصله نسلی معکوس (IGD):

این معیار برای تعیین میزان همگرایی جواب ها به کار برده می شود. برای داشتن IGD کوچکتر، باید نقاط به دست آمده توسط الگوریتم به جبهه پارتو نزدیک بوده و علاوه بر این، تعداد جواب های نامغلوب یافت شده توسط الگوریتم باید زیادتر بوده و نباید الگوریتم بخش هایی از جبهه را از دست داده باشد.

$$IGD = \frac{\sum d_i}{NPS_{know}} \quad (35)$$

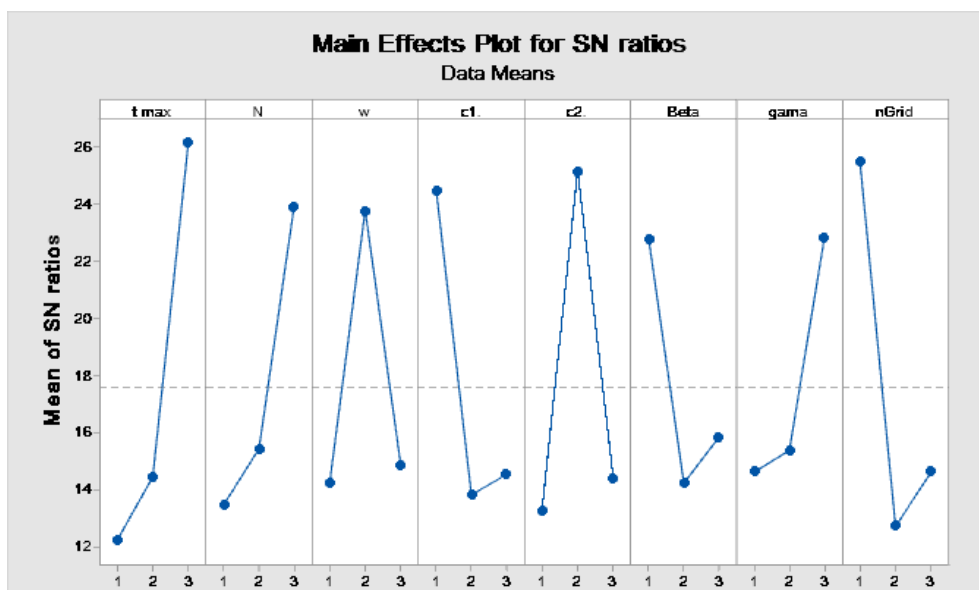
$$DM = \sqrt{\left(\frac{\max\{f_{1i}\} - \min\{f_{1i}\}}{f_1^{\max} - f_1^{\min}}\right)^2 + \left(\frac{\max\{f_{2i}\} - \min\{f_{2i}\}}{f_2^{\max} - f_2^{\min}}\right)^2 + \left(\frac{\max\{f_{3i}\} - \min\{f_{3i}\}}{f_3^{\max} - f_3^{\min}}\right)^2} \quad (31)$$

جدول (۲): پارامترهای الگوریتم MOPSO و سطوح هر یک از آن ها

سطوح	تعداد تکرار الگوریتم	اندازه جمعیت	ضریب اینرسی	ضریب شتاب شخصی	ضریب شتاب جمعی	فشار انتخاب	فشار حذف	تعداد جدول بندی
۱	۲۰	۴۰	۰/۸	۲/۰۵	۰/۵	۲	۰/۵	۷
۲	۳۰	۵۰	۱	۲/۵	۱	۳	۰/۷۵	۸
۳	۴۰	۶۰	۲	۳	۲	۴	۱	۹

جدول (۳): بهترین مقادیر ایجاد شده توسط روش تاگوچی برای پارامترهای الگوریتم MOPSO

پارامتر	تعداد تکرار الگوریتم	اندازه جمعیت	ضریب اینرسی	ضریب شتاب شخصی	ضریب شتاب جمعی	فشار انتخاب	فشار حذف	تعداد جدول بندی
سطح	۳	۳	۲	۱	۲	۱	۳	۱
مقدار بهینه	۴۰	۶۰	۱	۲/۰۵	۱	۲	۱	۷



شکل (۳): نمودار پارامترهای اصلی ارائه شده توسط روش تاگوچی برای نسبت سیگنال به نویز

۵-۲. ارزیابی عملکرد الگوریتم MOPSO توسعه داده شده

در این بخش نتایج مقایسه الگوریتم ازدحام ذرات چندهدفه با روش حل دقیق محدودیت اپسیلون تقویت شده و هم‌چنین الگوریتم فراابتکاری NSGA-II ارائه می‌گردد. نحوه عملکرد این الگوریتم، به این صورت است که ابتدا یک جمعیت اولیه به اندازه ۱۵۰ جواب به صورت تصادفی تولید گردیده و سپس رتبه‌بندی جبهه‌ها انجام می‌شود. پس از آن به کروموزوم‌های هر سطح (جبهه)، برازشی معادل رتبه آن تخصیص داده می‌شود. در نهایت با اعمال عملگرهای تقاطع و جهش هر یک با احتمال $0/6$ و $0/3$ ، جمعیت مرحله بعد تولید می‌شود. این کار تا زمان برقراری شرط توقف الگوریتم (۱۰۰ بار تکرار الگوریتم) ادامه می‌یابد. لازم به ذکر است که داده‌های مسأله با استفاده از نرم‌افزار اکسل و با استفاده از روشی مشابه مقاله [۴۲] ایجاد گردیده‌اند. دلیل این کار، نبودن داده‌های جامع در ادبیات موضوع می‌باشد. در جدول (۴) نتایج حاصل از اجرای الگوریتم پیشنهادی MOPSO، الگوریتم NSGA-II و روش محدودیت اپسیلون تقویت شده برای مثال‌های عددی با اندازه‌های مختلف آورده شده است. لازم به ذکر است که در این جدول n و m به ترتیب بیانگر تعداد کل کارها و تعداد کل ماشین‌ها می‌باشند. هم‌چنین در ستون سوم جدول دسته کارهای مربوط به هر یک از ۳ عامل نشان داده شده است. به عنوان مثال در مسأله با ۶ کار و ۲ ماشین کارهای ۱ و ۲ مربوط به عامل اول، کارهای ۳ و ۴ مربوط به عامل دوم و کارهای ۵ و ۶ مدنظر عامل سوم می‌باشند. نتایج نشان داده شده برای هر الگوریتم، متوسط مقادیر حاصل از ۵ بار اجرای آن الگوریتم می‌باشد. زمان حل نرم‌افزار GAMS بر روی ۱۰۰۰۰ ثانیه تنظیم شده است و در صورت عدم دستیابی به جواب بهینه تا زمان تنظیم شده، بهترین مقدار به دست آمده توسط این نرم‌افزار به عنوان جواب روش محدودیت اپسیلون تقویت شده در نظر گرفته می‌شود. بهترین مقادیر حاصل برای هر یک از معیارها در جدول (۴) به صورت پررنگ و با حاشیه سبز رنگ نشان داده شده‌اند. با استفاده از اطلاعات جدول (۴) نتایج زیر حاصل می‌گردد:

۱- معیار اول (Time): الگوریتم MOPSO از لحاظ این معیار در تمامی مسائل بهتر از دو روش دیگر عمل نموده و قادر به دستیابی به جواب‌های نهایی در زمانی کمتر از دو روش دیگر می‌باشد.

۲- معیار دوم (NPS): از نظر معیار تعداد نقاط غیرمغلوب نیز الگوریتم ازدحام ذرات چندهدفه بهتر از دو روش دیگر می‌باشد. زیرا همان‌طور که قبلاً اشاره گردید الگوریتمی که بتواند تعداد جواب‌های پارتویی بیشتری را حاصل نماید، از عملکرد بهتری برخوردار می‌باشد.

۳- معیار سوم (MID): از نظر این معیار که مربوط به میزان همگرایی جواب‌های ایجاد شده توسط الگوریتم‌ها می‌باشد، الگوریتم NSGA-II نسبت به دو روش دیگر بهتر می‌باشد. به دلیل زیادتر بودن تعداد نقاط پارتویی یافت شده توسط الگوریتم MOPSO نسبت به الگوریتم NSGA-II کاملاً منطقی و قابل پیش‌بینی است که به‌طور متوسط فاصله و میزان نزدیکی این تعداد بیشتر نقاط پارتویی یافت شده به

نقطه ایده‌آل زیادتر گردد. اما مزیت ارائه تعداد بیشتری نقاط پارتو این است که مجموعه‌ای از جواب‌های نامغلوب بیشتر و متنوع‌تر برای تصمیم‌گیرنده ایجاد می‌گردد تا تصمیم‌گیرنده با انعطاف بیشتر و راحت‌تر بتواند با توجه به شرایط مختلف، یک یا چندتا از این جواب‌ها را انتخاب نمود.

۴- معیار چهارم (DM): نتایج به دست آمده نشان می‌دهد که میزان پراکندگی جواب‌های الگوریتم MOPSO به‌طور قابل قبولی بیشتر از جواب‌های ایجاد شده توسط دو روش دیگر می‌باشد. در واقع جواب‌های نامغلوب یافته شده توسط الگوریتم مذکور محدوده بیشتری را پوشش داده و در نتیجه الگوریتم موردنظر از نظر این معیار از عملکرد بهتری نسبت به دو روش دیگر برخوردار می‌باشد.

۵- معیار پنجم (ONVGR): برای تعیین جبهه پارتو، نقاط نامغلوب یافت شده هر سه روش را با هم در نظر گرفته می‌شود. اینجا تعدادی نقاط مغلوب می‌شوند. از آنجاکه تعداد نقاط نامغلوب یافت شده توسط الگوریتم MOPSO برای تمامی مسائل بیشتر از دو روش دیگر می‌باشد، بنابراین الگوریتم مذکور از نظر این معیار بهتر از دو روش دیگر عمل می‌نماید.

۶- معیار ششم (GD) و معیار هفتم (IGD): روش محدودیت اپسیلون تقویت شده در معیار فاصله نسلی معکوس و فاصله عمومی عملکرد بهتری نسبت به دو روش دیگر دارد. همان‌طور که از جدول (۴) مشخص است، تعداد جواب‌های نامغلوب یافت شده توسط این روش خیلی کمتر از دو روش دیگر می‌باشد. هم‌چنین برای ۸ تا از ۹ مسأله بررسی شده این جواب‌های نامغلوب دقیقاً در جبهه پارتوی بهینه نیز حضور دارند. بنابراین بدیهی است که فاصله این جواب‌های نامغلوب تا جبهه پارتو ناچیز و در اکثر موارد برابر با ۰ باشد.

باتوجه به این توضیحات الگوریتم MOPSO یک روش مناسب برای حل مسأله زمان‌بندی چندعاملی در نظر گرفته شده در این مقاله می‌باشد. هم‌چنین لازم به ذکر است که جهت انتخاب الگوریتم برتر از روش تصمیم‌گیری چندمعیاره ویکور استفاده شد. وزن در نظر گرفته شده برای معیار $Time$ برابر با $0/1$ و برای سایر معیارها برابر با $0/15$ می‌باشد. روش ویکور یک ابزار مفید جهت تصمیم‌گیری چندمعیاره به حساب می‌آید. به‌خصوص در مواقعی که تصمیم‌گیرندگان به علت وجود شاخص‌های متناقض به راحتی نمی‌توانند ترجیحات خود را در مسأله تصمیم‌گیری ابراز کنند. راهکار سازشی به دست آمده در روش ویکور مورد توافق تصمیم‌گیرندگان خواهد بود. زیرا این راهکار مطلوبیت گروهی را حداکثر (توسط شاخص S) و تأثیرات فردی را حداقل (توسط شاخص R) می‌سازد. مراحل روش ویکور عبارتند از:

- ۱- ارزش معیار z_k برای گزینه k ام را با f_{kj} تعریف می‌کنیم. f_{ij}^* مطلوب‌ترین مقدار معیار z_k و f_{ij}^- نامطلوب‌ترین مقدار معیار z_k می‌باشد.
- ۲- برای گزینه k ام، مقدار S_k ، R_k و Q_k به صورت زیر حساب می‌گردد:

$$S_k = \sum_{j=1}^n w_j \frac{f_j^* - f_{kj}}{f_j^* - f_j^-} \quad (32)$$

بود. هم چنین در صورت عدم برقراری شرط اول، a', a'', \dots, a^z بهترین گزینه ها خواهند بود به گونه ای که

$$Q(a^z) - Q(a') < DQ$$

مطابق این روش، الگوریتم MOPSO به عنوان الگوریتم برتر انتخاب می گردد.

شکل های (۴) و (۵) نیز به صورت واضح بیانگر این هستند که راه حل های ایجاد شده توسط الگوریتم MOPSO به جبهه پارتوی اصلی نزدیک تر بوده و دارای پراکندگی یکنواخت تری در فضای هدف می باشند. دلیل این امر، عملگرهای جهش مناسب مورد استفاده و هم چنین روش مناسب به کار برده شده جهت انتخاب رهبران شخصی و جمعی در الگوریتم مذکور می باشد.

۳-۵. تحلیل حساسیت پارامترهای مهم مسأله

در این قسمت، به ارزیابی عملکرد مدل پیشنهادی و تحلیل رفتار مدل تحت مقادیر مختلف پارامترها پرداخته می شود. برای این منظور لازم است که با تغییر مقادیر پارامترهای مهم مسأله، چگونگی پاسخ مدل به این تغییرات سنجیده شود. در ادامه تحلیل حساسیت های انجام شده بر روی مدل و نتایج حاصل شده از آن ها ارائه می شود.

$$R_k = \max_j \left[w_j \frac{f_j^* - f_{kj}}{f_j^* - f_j^-} \right] \quad (33)$$

$$Q_k = v \left[\frac{S_k - S^*}{S^- - S^*} \right] + (1 - v) \left[\frac{R_k - R^*}{R^- - R^*} \right] \quad (34)$$

$$S^* = \min_k (S_k) \quad S^- = \max_k (S_k) \quad (35)$$

$$R^* = \min_k (R_k) \quad R^- = \max_k (R_k) \quad (36)$$

۳- Q و R ، S صورت صعودی مرتب می گردند.

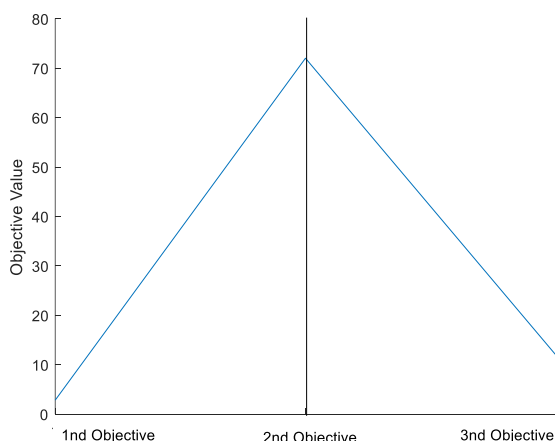
۴- گزینه a' که کمترین مقدار را در لیست Q دارد انتخاب می گردد. اگر a' دارای دو شرط زیر باشد، به عنوان بهترین گزینه انتخاب خواهد شد.

$$Q(a') - Q(a'') \geq DQ$$

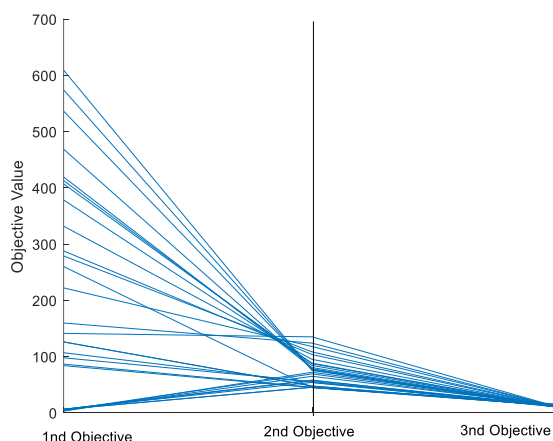
a'' دومین گزینه برتر است و $DQ = \frac{1}{m-1}$ تعداد (m گزینه هاست)

a' باید بهترین گزینه در لیست S و یا R باشد.

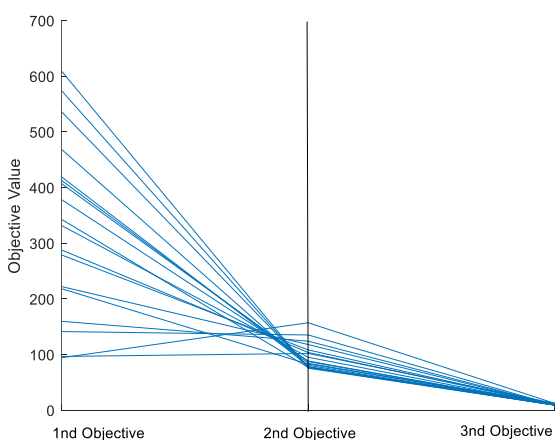
۵- در صورت عدم برقراری شرط دوم، a' و a'' بهترین گزینه خواهند



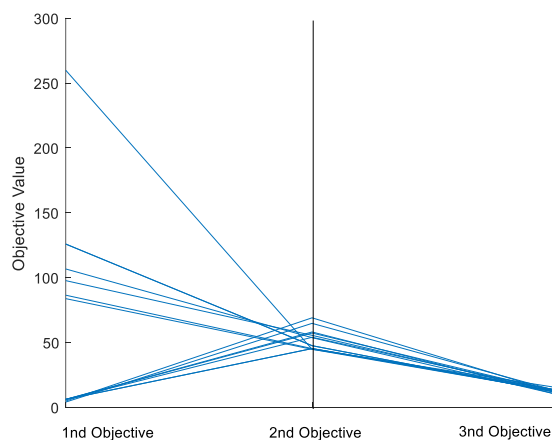
ب- روش محدودیت افسیلون تقویت شده



الف- جواب های غیرمغلوب یافت شده



د- الگوریتم MOPSO توسعه داده شده

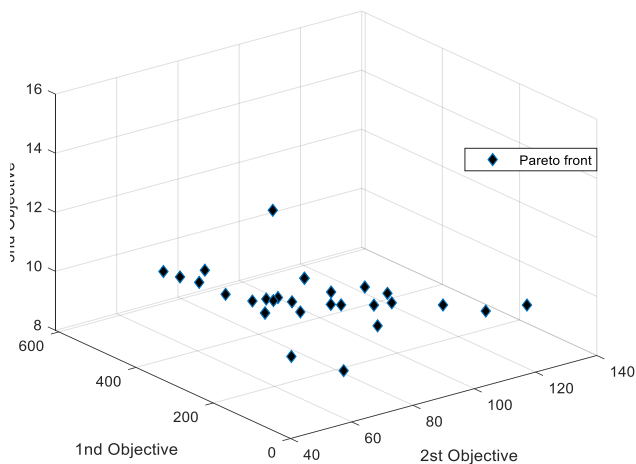


ج- الگوریتم NSGA-II

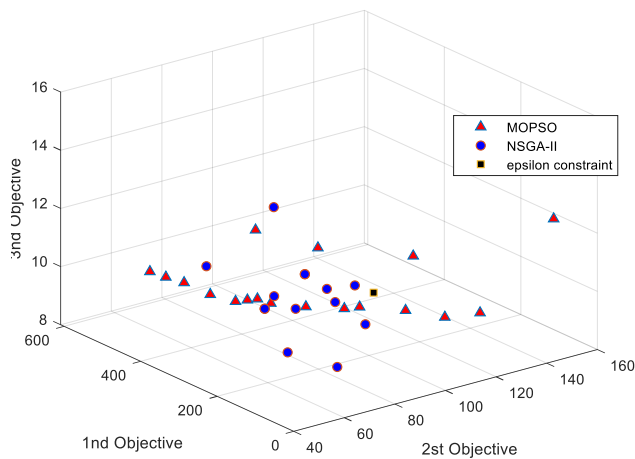
شکل (۴): نمودار محورهای موازی برای جواب های نامغلوب ایجاد شده توسط روش های مختلف برای مسأله با ۱۲ کار و ۲ ماشین

جدول (۴): مقایسه الگوریتم MOPSO با دو روش محدودیت اِپسیلون تقویت شده و NSGA-II براساس معیارهای معرفی شده

MOPSO			NAGA-II				augmented ϵ -constraint								دسته کارها	n×m	اندازه						
GD	IGD	ONVGR	DM	MID	NPS	Time	GD	IGD	ONVGR	DM	MID	NPS	Time	GD				IGD	ONVGR	DM	MID	NPS	Time
۱۳/۳۹	۳۰/۳۱	۰/۹۵	۱/۵۳	۰/۹۵	۱۹	۱۰/۵۷	۲/۶۸	۶/۶۰	۰/۴	۰/۳۵۱	۰/۶۱۶	۵	۸۵/۴۹۷	۰	۰	۰/۱	۰/۴۶	۰/۸۶	۲	۵۲/۸۷		۶×۲	
۹/۴۵	۲۵/۱۵	۰/۸۷	۱/۶۵	۰/۹۸	۱۳	۱۰/۵۹	۱/۸۹	۵/۳۰	۰/۳۳	۰/۵۳۵	۰/۳۶۲	۵	۸۷/۰۲۶	۰	۰	۰/۱۳	۰/۴۱	۰/۸۵	۲	۵۸/۳۲	1,2,3/ 4,5/ 6	۶×۲	کوچک
۱۰/۱۲	۳۱/۲۴	۰/۹	۰/۷۱۴	۱/۳۰	۹	۸/۵۳	۲/۲۴	۶/۲۶	۰/۲	۰/۰۱	۰/۱۳۶	۲	۷۵/۵۲۶	۰	۰	۰/۲	۱/۴۱	۰/۳۵	۲	۶۵/۲۴		۶×۵	
۰/۸۲	۰/۸۳	۰/۶۷	۰/۵۴	۰/۵۱	۰/۶۷	۰/۰۶	۰/۱۸	۰/۱۷	۰/۲۲۸	۰/۱۳	۰/۱۷	۰/۲۲۸	۰/۵۵	۰	۰	۰/۱۰۲	۰/۳۳	۰/۳۲	۰/۱۰۲	۰/۳۹		میانگین نرمال شده	
۱/۰۹	۲۷/۶	۰/۹۲	۱/۳۶	۰/۹۷	۲۳	۱۱/۵۳	۰/۲۲	۵/۵۲	۰/۲۸	۰/۶۹۶	۰/۷۳۶	۷	۹۹/۹۶۷	۵/۱۰	۵/۱۰	۰/۰۴	۰	۰/۷۰	۱	۱۵۵۶		۸×۲	
۲/۷۶	۵/۹۵	۱	۱/۸۴	۱/۰۸	۲۱	۱۰/۱۸	۰/۵۵	۱/۱۸	۰/۲	۰/۲۳۱	۰/۵۹۳	۴	۱۱۹/۴۸	۰	۰	۰/۱	۱/۰۸	۰/۸۵	۲	۱۶۱۲	1,2,3/ 4,5,6/ 7,8	۸×۳	متوسط
۱۳/۵۶	۳۵/۸۳	۰/۹۴	۲/۳۳	۱/۲۵	۱۶	۱۰/۱۵	۷/۴۱	۹/۷۸	۰/۱۲	۰/۴۷۱	۰/۳۱۲	۲	۱۰۳/۰۳۶	۰	۰	۰/۰۶	۰	۰/۱۵	۱	۱۱۹۷/۲۳		۸×۵	
۰/۵۸	۰/۷۸	۰/۷۸	۰/۷۹۱	۰/۵	۰/۷۸	۰/۰۱	۰/۲۶	۰/۱۷	۰/۱۷	۰/۲	۰/۲۴	۰/۱۷	۰/۰۵	۰/۱۶	۰/۰۵	۰/۰۵	۰/۱۵۵	۰/۲۶	۰/۰۵	۰/۹۴		میانگین نرمال شده	
۵/۰۹	۹/۷۴	۰/۶۷	۱/۳۷	۰/۹۴	۲۰	۱۲/۴۰	۱/۸۶	۲/۳۵	۰/۵	۰/۴۴	۰/۲۷	۱۵	۱۰۱/۹۷۵	۰	۰	۰/۰۳۳	۰	۱/۰۲	۱	۸۲۵۸		۱۲×۲	
۱۱/۵۴	۲۴/۱۹	۰/۹۲	۱/۹۲	۰/۹۹	۲۳	۱۲/۰۴	۲/۷۶	۴/۷۶	۰/۳۶	۰/۸۰۹	۰/۲۳۵	۹	۱۰۷/۴۱۱	۰	۰	۰/۰۴	۰	۰/۷۵	۱	۱۲۴۶۴	1-4/ 5-8/ 9-12	۱۲×۳	بزرگ
۸/۶۰	۱۴/۱۸	۰/۹۲	۰/۷۷	۱/۰۸	۱۲	۱۱/۲۳	۳/۲۴	۴/۵۹	۰/۲۱	۰/۱	۰/۱۴۷	۳	۱۱۱/۲۲۹	۰	۰	۰/۰۷۷	۰	۱	۱	۱۳۳۴۹		۱۲×۵	
۰/۷۶	۰/۸	۰/۶۴۷	۰/۷۵	۰/۴۷	۰/۶۴۷	۰/۰۰۱	۰/۲۴	۰/۲	۰/۳۱۸	۰/۲۵	۰/۱	۰/۳۱۸	۰/۰۰۹	۰	۰	۰/۰۳۵	۰	۰/۴۳	۰/۰۳۵	۰/۹۹		میانگین نرمال شده	
۰/۷۲	۰/۸۰۳	۰/۷۳	۰/۶۶	۰/۵	۰/۷۳	۰/۰۲۳	۰/۲۳	۰/۱۸	۰/۱۹	۰/۱۸	۰/۱۶	۰/۱۹	۰/۲۰۳	۰/۰۵	۰/۰۱۷	۰/۰۸	۰/۱۶	۰/۳۳	۰/۰۸	۰/۷۷		میانگین نرمال شده کلی	



ب- جبهه پارتوی ایجادشده



الف- جواب های نامغلوب ایجادشده توسط روش های مختلف

شکل (۵): جواب های نامغلوب و جبهه پارتوی ایجادشده برای مسأله با ۱۲ کار و ۲ ماشین برای سه تابع هدف F3, F2, F1

توابع هدف اول و سوم کاهش می یابد. شکل (۷) به خوبی این امر را نشان می دهد. همزمان با کاهش توابع هدف اول و سوم، از آنجاکه به دلیل افزایش زمان تحویل کارها دیرتر تکمیل می گردند، بنابراین زمان تکمیل کارها و در نتیجه مقدار تابع هدف دوم افزایش می یابد که این امر نشان دهنده وجود تضاد بین اهداف مختلف است.

- تأثیر کاهش هزینه رد کردن کارها بر توابع هدف مختلف:

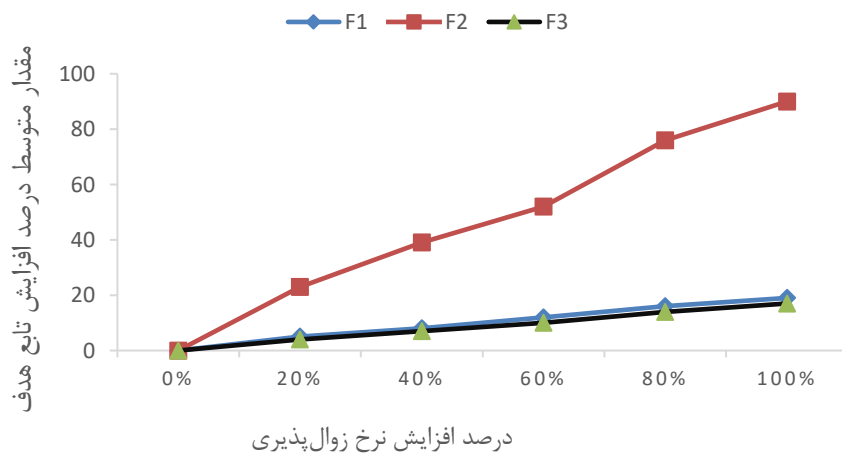
مطابق شکل (۸) با کاهش هزینه رد کردن کارها تا ۲۰٪، تابع هدف اول کاهش و دو تابع دیگر افزایش می یابند. دلیل کاهش تابع هدف اول این است که با کاهش هزینه رد کردن کارها، تعداد بیشتری از کارهای عامل اول رد شده و در نتیجه از هزینه تابع اول کاسته می شود. افزایش میزان تابع هدف دوم تا ۳۰٪ کاهش هزینه رد کردن کارها نیز ادامه یافته و پس از آن با ادامه کاهش هزینه مذکور، روند کاهشی در تابع هدف دوم مشاهده می گردد. هم چنین پس از ۲۰٪ تا ۵۰٪ کاهش هزینه رد کردن کارها، تابع هدف اول این بار افزایش یافته و تابع هدف سوم کاهش می یابد.

- تأثیر افزایش نرخ زوال پذیری بر توابع هدف مختلف:

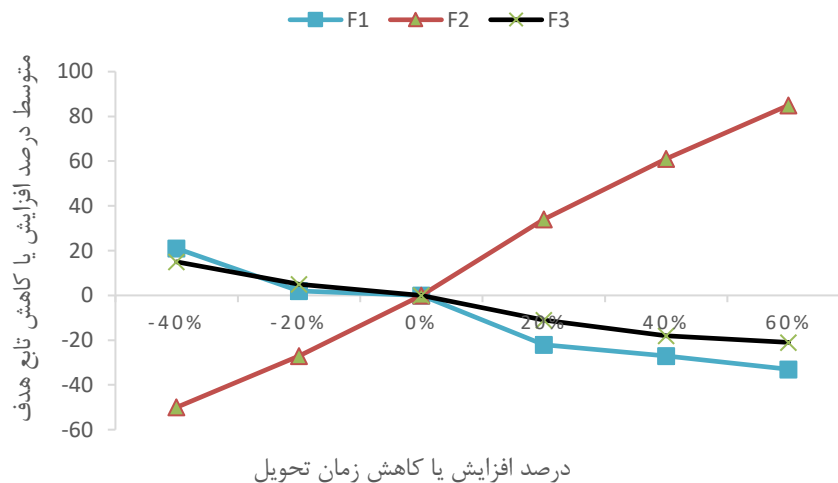
در این بخش تأثیر افزایش نرخ زوال پذیری بر توابع هدف مختلف بررسی خواهد شد. انتظار می رود که اگر نرخ زوال پذیری افزایش یابد، زمان پردازش کار نیز افزایش یابد. برای بررسی این موضوع، نرخ زوال پذیری در ۵ مقدار مختلف افزایش یافته و نتایج با حالت اولیه آن ها مقایسه شده و در شکل (۶) نشان داده شده است. با افزایش نرخ زوال پذیری، بدیهی است که زمان پردازش کارها افزایش یافته و در نتیجه زمان اتمام کار بیشتر گردیده و برخی کارها با تأخیر مواجه می شوند. همان طور که از شکل مشخص است، تأثیر افزایش نرخ زوال پذیری بر تابع هدف دوم خیلی بیشتر از توابع هدف اول و سوم می باشد.

- تأثیر افزایش و کاهش زمان تحویل بر توابع هدف مختلف:

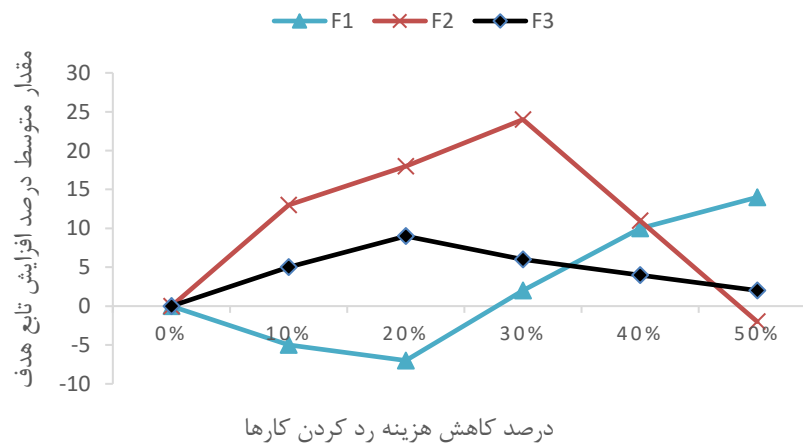
در این بخش تأثیر تغییر در مقدار زمان تحویل بر توابع هدف مختلف بررسی می گردد. بدیهی است که با افزایش مدت زمان تحویل، مدت زمان تأخیر کارها و تعداد کارهای رد شده کاهش و در نتیجه مقدار



شکل (۶): تحلیل حساسیت افزایش نرخ زوال پذیری



شکل (۷): تحلیل حساسیت افزایش یا کاهش زمان تحویل



شکل (۸): تحلیل حساسیت کاهش هزینه رد کردن کارها

روش محدودیت اپسیلون تقویت شده و الگوریتم NSGA-II می‌باشد. برای مطالعات آتی، می‌توان مسأله زمان‌بندی چندعاملی معرفی شده را با در نظر گرفتن مفروضات واقعی مانند اثر یادگیری، محدودیت در دسترس بودن ماشین آلات، محدودیت بافر بین مراحل یا ماشین‌های مختلف، زمان آماده‌سازی یا آزادسازی کارها به صورت احتمالی و یا فعالیت تعمیر و نگهداری احتمالی مدل‌سازی نمود. همچنین می‌توان از سایر معیارهای عملکردی مناسب برای مسائل چندهدفه جهت تنظیم پارامترها، مقایسه و اثبات کارایی الگوریتم پیشنهادی استفاده نمود. به‌عنوان پیشنهاد آخر می‌توان از سایر الگوریتم‌های ابتکاری و فراابتکاری جهت حل مسأله موردنظر در این مقاله استفاده نمود.

مراجع

- [1] M. Pinedo, C. Zacharias, and N. Zhu, "Scheduling in the service industries: An overview," *J. Syst. Sci. Syst. Eng.*, vol. 24, no. 1, pp. 1–48, 2015.
- [2] Y. Fu, H. Wang, G. Tian, Z. Li, and H. Hu, "Two-agent stochastic flow shop deteriorating scheduling

۶. نتیجه‌گیری و جمع‌بندی

در این مقاله یک مسأله زمان‌بندی چندعاملی در محیط جریان کارگاهی با فرض زوال‌پذیری کارها، زمان‌های آماده‌سازی وابسته به توالی و زمان آزادسازی کارها بررسی گردید. یک مدل برنامه‌ریزی عدد صحیح مختلط برای این مسأله توسعه داده شده و یک الگوریتم فراابتکاری ازدحام ذرات چندهدفه برای این مسأله ارائه گردید. از آنجاکه خروجی الگوریتم‌های فراابتکاری به شدت به مقادیر ورودی آن‌ها وابسته است، در این پژوهش برای تنظیم پارامترهای الگوریتم MOPSO از روش تاگوچی استفاده گردید. الگوریتم MOPSO در مثال‌های عددی مختلف با روش حل محدودیت اپسیلون تقویت شده و الگوریتم ژنتیک مبتنی بر مرتب‌سازی نامغلوب براساس چهار معیار زمان، تعداد جواب‌های موجود در جبهه پارتو، میانگین فاصله از جواب ایده‌آل و پراکندگی مقایسه گردید. سپس جهت انتخاب الگوریتم برتر از روش تصمیم‌گیری چندمعیاره ویکور استفاده شد. نتایج حاصل نشان‌دهنده همگرایی قابل قبول و پراکندگی خیلی خوب راه‌حل‌های الگوریتم MOPSO و همچنین عملکرد بهتر این الگوریتم نسبت به

- [16] S. Lu, X. Liu, J. Pei, and P. M. Pardalos, "Permutation flowshop manufacturing cell scheduling problems with deteriorating jobs and sequence dependent setup times under dominant machines," *Optim. Lett.*, vol. 15, no. 2, pp. 537–551, 2021.
- [17] H. Xuan, H. Zhang, and B. Li, "An improved discrete artificial bee colony algorithm for flexible flowshop scheduling with step deteriorating jobs and sequence-dependent setup times," *Math. Probl. Eng.*, vol. 2019, 2019.
- [18] S. Chand, R. Traub, and R. Uzsoy, "An iterative heuristic for the single machine dynamic total completion time scheduling problem," *Comput. Oper. Res.*, vol. 23, no. 7, pp. 641–651, 1996.
- [19] M. Sevaux and K. Sörensen, *A genetic algorithm for robust schedules in a just-in-time environment*. UA, Faculty of Applied Economics UFSIA-RUCA, 2003.
- [20] B. Chang, C.-W. Chang, and C.-H. Wu, "Fuzzy DEMATEL method for developing supplier selection criteria," *Expert Syst. Appl.*, vol. 38, no. 3, pp. 1850–1858, 2011.
- [21] M. D. Tksarı, "A branch and bound algorithm for minimizing makespan on a single machine with unequal release times under learning effect and deteriorating jobs," *Comput. Oper. Res.*, vol. 38, no. 9, pp. 1361–1365, 2011.
- [22] C.-C. Wu, P.-H. Hsu, and K. Lai, "Simulated-annealing heuristics for the single-machine scheduling problem with learning and unequal job release times," *J. Manuf. Syst.*, vol. 30, no. 1, pp. 54–62, 2011.
- [23] R. Xu, H. Chen, and X. Li, "Makespan minimization on single batch-processing machine via ant colony optimization," *Comput. Oper. Res.*, vol. 39, no. 3, pp. 582–593, 2012.
- [24] M. A. Rakrouki, T. Ladhari, and V. T'kindt, "Coupling Genetic Local Search and Recovering Beam Search algorithms for minimizing the total completion time in the single machine scheduling problem subject to release dates," *Comput. Oper. Res.*, vol. 39, no. 6, pp. 1257–1264, 2012.
- [25] W.-C. Lee, Y.-H. Chung, and M.-C. Hu, "Genetic algorithms for a two-agent single-machine problem with release time," *Appl. Soft Comput.*, vol. 12, no. 11, pp. 3580–3589, 2012.
- [26] M. Mahnam, G. Moslehi, and S. M. T. F. Ghomi, "Single machine scheduling with unequal release times and idle insert for minimizing the sum of maximum earliness and tardiness," *Math. Comput. Model.*, vol. 57, no. 9–10, pp. 2549–2563, 2013.
- [27] M. C. Vélez-Gallego, J. Maya, and J. R. Montoya-Torres, "A beam search heuristic for scheduling a single machine with release dates and sequence dependent setup times to minimize the makespan," *Comput. Oper. Res.*, vol. 73, pp. 132–140, 2016.
- [28] A. Abdelhadi, L. H. Mouss, and O. Kadri, "Hybrid multi-agent and immune algorithm approach to hybrid flow shops scheduling with sdst," *Acad. J. Manuf. Eng.*, vol. 18, no. 3, 2020.
- [29] J. Belabid, S. Aqil, and K. Allali, "Solving via a hybrid multi-objective evolutionary algorithm," *J. Intell. Manuf.*, vol. 30, no. 5, pp. 2257–2272, 2019.
- [3] A. Agnetis, J.-C. Billaut, S. Gawiejnowicz, D. Pacciarelli, and A. Soukhal, "Multiagent scheduling," *Berlin Heidelberg. Springer Berlin Heidelberg. doi*, vol. 10, no. 1007, pp. 973–978, 2014.
- [4] M. Cheng, P. R. Tadikamalla, J. Shang, and B. Zhang, "Two-machine flow shop scheduling with deteriorating jobs: minimizing the weighted sum of makespan and total completion time," *J. Oper. Res. Soc.*, vol. 66, no. 5, pp. 709–719, 2015.
- [5] A. Allahverdi, "The third comprehensive survey on scheduling problems with setup times/costs," *Eur. J. Oper. Res.*, vol. 246, no. 2, pp. 345–378, 2015.
- [6] J. N. D. Gupta and S. K. Gupta, "Single facility scheduling with nonlinear processing times," *Comput. Ind. Eng.*, vol. 14, no. 4, pp. 387–393, 1988.
- [7] H. Wang, Y. Fu, M. Huang, and J. Wang, "Multiobjective optimisation design for enterprise system operation in the case of scheduling problem with deteriorating jobs," *Enterp. Inf. Syst.*, vol. 10, no. 3, pp. 268–285, 2016.
- [8] A. S. Kunnathur and S. K. Gupta, "Minimizing the makespan with late start penalties added to processing times in a single facility scheduling problem," *Eur. J. Oper. Res.*, vol. 47, no. 1, pp. 56–64, 1990.
- [9] Tce. Cheng, W.-C. Lee, and C.-C. Wu, "Single-machine scheduling with deteriorating functions for job processing times," *Appl. Math. Model.*, vol. 34, no. 12, pp. 4171–4178, 2010.
- [10] M. Cheng, S. Sun, and L. He, "Flow shop scheduling problems with deteriorating jobs on no-idle dominant machines," *Eur. J. Oper. Res.*, vol. 183, no. 1, pp. 115–124, 2007.
- [11] W.-C. Lee, W.-C. Yeh, and Y.-H. Chung, "Total tardiness minimization in permutation flowshop with deterioration consideration," *Appl. Math. Model.*, vol. 38, no. 13, pp. 3081–3092, 2014.
- [12] J.-B. Wang and M.-Z. Wang, "Solution algorithms for the total weighted completion time minimization flow shop scheduling with decreasing linear deterioration," *Int. J. Adv. Manuf. Technol.*, vol. 67, no. 1–4, pp. 243–253, 2013.
- [13] M. Cheng, P. R. Tadikamalla, J. Shang, and S. Zhang, "Bicriteria hierarchical optimization of two-machine flow shop scheduling problem with time-dependent deteriorating jobs," *Eur. J. Oper. Res.*, vol. 234, no. 3, pp. 650–657, 2014.
- [14] A. Azadeh, A. H. Goodarzi, M. H. Kolae, and S. Jebreili, "An efficient simulation–neural network–genetic algorithm for flexible flow shops with sequence-dependent setup times, job deterioration and learning effects," *Neural Comput. Appl.*, vol. 31, no. 9, pp. 5327–5341, 2019.
- [15] N. He, Y. Qiao, N. Wu, and T. Qu, "Total completion time minimization for scheduling of two-machine flow shop with deterioration jobs and setup time," *Adv. Mech. Eng.*, vol. 9, no. 4, p. 1687814017698887, 2017.

- [40] W.-H. Wu, Y. Yin, W.-H. Wu, C.-C. Wu, and P.-H. Hsu, "A time-dependent scheduling problem to minimize the sum of the total weighted tardiness among two agents," *J. Ind. Manag. Optim.*, vol. 10, no. 2, p. 591, 2014.
- [41] Y. Yin, T. C. E. Cheng, L. Wan, C.-C. Wu, and J. Liu, "Two-agent single-machine scheduling with deteriorating jobs," *Comput. Ind. Eng.*, vol. 81, pp. 177–185, 2015.
- [۴۲] حراثیان، و حسین زاده، م، حل مسئله زمان‌بندی چندعاملی در محیط جریان کارگاهی با در نظر گرفتن اثر زمانی و رد کردن کارها با استفاده از یک الگوریتم فراابتکاری، نشریه پژوهش‌های مهندسی صنایع در سیستم‌های تولید، شماره ۱۰، ص ۱۷-۲۹، ۱۳۹۶.
- [43] T. V Adediran, A. Al-Bazi, and L. E. dos Santos, "Agent-based modelling and heuristic approach for solving complex OEM flow-shop productions under customer disruptions," *Comput. Ind. Eng.*, vol. 133, pp. 29–41, 2019.
- [44] D.-S. Chen, R. G. Batson, and Y. Dang, *Applied integer programming: modeling and solution*. John Wiley & Sons, 2011.
- [45] S. J. Sadjadi, M. Heidari, and A. A. Esboei, "Augmented ϵ -constraint method in multiobjective staff scheduling problem: a case study," *Int. J. Adv. Manuf. Technol.*, vol. 70, no. 5–8, pp. 1505–1514, 2014.
- [46] R. Eberhart and J. Kennedy, "Particle swarm optimization," in *Proceedings of the IEEE international conference on neural networks*, 1995, vol. 4, pp. 1942–1948.
- [47] C. A. C. Coello and M. S. Lechuga, "MOPSO: A proposal for multiple objective particle swarm optimization," in *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No. 02TH8600)*, vol. 2, pp. 1051–1056, 2002.
- [48] S. Saeedi, R. Khorsand, S. G. Bidgoli, and M. Ramezani, "Improved many-objective particle swarm optimization algorithm for scientific workflow scheduling in cloud computing," *Comput. Ind. Eng.*, vol. 147, p. 106649, 2020.
- permutation flow shop scheduling problem with sequence-independent setup time," *J. Appl. Math.*, 2020.
- [30] T. Ladhari and M. A. Rakrouki, "Heuristics and lower bounds for minimizing the total completion time in a two-machine flowshop," *Int. J. Prod. Econ.*, vol. 122, no. 2, pp. 678–691, 2009.
- [31] H. Amirian and R. Sahraeian, "Augmented ϵ -constraint method in multi-objective flowshop problem with past sequence set-up times and a modified learning effect," *Int. J. Prod. Res.*, vol. 53, no. 19, pp. 5962–5976, 2015.
- [32] W.-C. Lee, S.-K. Chen, C.-W. Chen, and C.-C. Wu, "A two-machine flowshop problem with two agents," *Comput. Oper. Res.*, vol. 38, no. 1, pp. 98–104, 2011.
- [33] W. Luo, L. Chen, and G. Zhang, "Approximation schemes for two-machine flow shop scheduling with two agents," *J. Comb. Optim.*, vol. 24, no. 3, pp. 229–239, 2012.
- [34] B. Mor and G. Mosheiov, "Polynomial time solutions for scheduling problems on a proportionate flowshop with two competing agents," *J. Oper. Res. Soc.*, vol. 65, no. 1, pp. 151–157, 2014.
- [35] B. Q. Fan and T. C. E. Cheng, "Two-agent scheduling in a flowshop," *Eur. J. Oper. Res.*, vol. 252, no. 2, pp. 376–384, 2016.
- [36] D. Lei, "Variable neighborhood search for two-agent flow shop scheduling problem," *Comput. Ind. Eng.*, vol. 80, pp. 125–131, 2015.
- [37] Y.-R. Shiau, W.-C. Lee, Y.-S. Kung, and J.-Y. Wang, "A lower bound for minimizing the total completion time of a three-agent scheduling problem," *Inf. Sci. (Ny)*, vol. 340, pp. 305–320, 2016.
- [38] V. Nasrollahi, G. Moslehi, and M. Reisi-Nafchi, "Minimizing the weighted sum of maximum earliness and maximum tardiness in a single-agent and two-agent form of a two-machine flow shop scheduling problem," *Oper. Res.*, pp. 1–40, 2020.
- [39] D. Bai *et al.*, "Competitive bi-agent flowshop scheduling to minimise the weighted combination of makespans," *Int. J. Prod. Res.*, pp. 1–22, 2021.



DOI: 10.22084/IER.2021.23501.2009

Multi Agent Flow Shop Scheduling Model with Deteriorating Jobs and Sequence-Dependent Setup Times Using Multi Objective Particle Swarm Optimization (MOPSO) Algorithm

S. Ghandi Bidgoli^{1*}, M. Amini Rarani²

¹. Assistant Professor Department of Industrial Engineering, Faculty of Engineering, University of Kashan, Kashan, Iran

². M.A Student Department of Industrial Engineering, Iran University of Science & Technology, Tehran, Iran

ARTICLE INFO

Article history:

Received 29 January 2021

Accepted 26 June 2021

Keywords:

Flow shop scheduling
Deteriorating jobs
Multi agent scheduling
Many Objective Particle Swarm
Optimization (MOPSO) algorithm
Sequence-Dependent Setup Time
Release date

ABSTRACT

Multi-agent and deteriorating scheduling has gained an increasing concern from academic and industrial communities in recent years. Multi-agent scheduling problem is a subset of multi-objective scheduling problems in which each agent has a set of jobs and its aim is to optimize its own objective function. This study addresses a three-agent flow shop deteriorating scheduling problem. In the investigated problem, the actual processing time of jobs is a linear function of their normal processing time and starting time. To make the proposed problem more realistic, two practical assumptions such as sequence-dependent setup times and release date of jobs are considered. A mixed integer programming model has also been developed for the problem, which is solved using the Augmented ϵ -constraint exact method. Also due to the complexity of the model and its inability to solve large-scale problems, Multi Objective Particle Swarm Optimization (MOPSO) algorithm is developed. Since the parameters of meta-heuristic algorithms affect their overall performance and output, the Taguchi method has been used to adjust the parameters of the MOPSO algorithm. Finally, in order to evaluate the performance of the proposed algorithm, numerical sample problems of different structures are solved using this algorithm as well as the Non-dominated Sorting Genetic Algorithm (NSGA-II) and the exact augmented ϵ -constraint method. The computational results confirm the acceptable convergence and very good dispersion of the solutions of the MOPSO algorithm as well as the better performance of this algorithm compared to the augmented ϵ -constraint method and the NSGA-II algorithm.

* Corresponding author. S. Ghandi Bidgoli
Tel.:031-55912416; E-mail address: s.ghandi@kashanu.ac.ir