

## زمان بندی دوهدفه خطوط جریان منعطف بدون وقفه با پنجره زمانی تحویل و امکان رد کار

بابک اسلانی<sup>۱</sup>، مصطفی زندیه<sup>۲</sup>، مجید عادل<sup>۳</sup>

**چکیده:** یکی از چالش های اساسی تولیدکنندگان در سیستم های تولید براساس سفارش، برقراری توازن بین سفارش های ورودی و منابع محدود است. از این رو آنها باید برخی سفارش ها را انتخاب و بعضی را رد کنند. علاوه بر این، در پذیرش سفارش ها باید زمان بندی کارهای جاری در نظر گرفته شود. در تحقیق پیش رو در مورد مسئله زمان بندی و پذیرش سفارش در سیستم تولیدی خط جریان منعطف بدون وقفه بحث شده است. سفارش هایی که امکان تحویل در موعد مقرر را نداشته باشند، رد می شوند. بیشتر تحقیقات صورت گرفته مبتنی بر بهینه سازی تک هدفه اند، در صورتی که در شرایط واقعی باید اهداف مختلف را در تصمیم گیری مدنظر قرار داد. به همین دلیل مدل دوهدفه ای با اهداف حداکثرسازی سود و حداقل سازی میزان انحراف ها از موعد تحویل، ارائه شده است. به دلیل پیچیدگی زیاد مدل، روش های دقیق جواب بهینه ای را تولید نمی کنند، بنابراین دو الگوریتم فراابتکاری NSGA II و MOSA برای این مسئله توسعه داده شده و عملکرد آنها در سه دسته مختلف مسائل از طریق شاخص های کمی مقایسه شد.

**واژه های کلیدی:** بهینه سازی چندهدفه، پنجره زمانی تحویل، جریان منعطف بدون وقفه، خط، زمان بندی و پذیرش سفارش.

۱. کارشناس ارشد مدیریت صنعتی، دانشکده مدیریت و حسابداری، دانشگاه شهید بهشتی، تهران، ایران

۲. دانشیار گروه مدیریت دانشکده مدیریت و حسابداری، دانشگاه شهید بهشتی، تهران، ایران

۳. دانشجوی دکتری مدیریت تولید و عملیات، دانشکده مدیریت و حسابداری، دانشگاه شهید بهشتی، تهران، ایران

تاریخ دریافت مقاله: ۱۳۹۴/۰۱/۱۴

تاریخ پذیرش نهایی مقاله: ۱۳۹۴/۰۶/۲۳

نویسنده مسئول مقاله: مجید عادل

E-mail: majidadeli@gmail.com

## مقدمه

در دهه گذشته در صنعت و همچنین تحقیقات دانشگاهی به طور گسترده‌ای بر اهمیت راهبردی پذیرش تأکید شده است. شرکت‌ها به دلایلی از جمله مسیر راهبردی (تمرکز روی بخش‌های خاص بازار) و تخصیص ظرفیت (نگهداری ظرفیت محدود برای سودآورترین و مطلوب‌ترین مشتریان)، بین کارهایی که برای پردازش می‌پذیرند، تمایز قائل می‌شوند. تصمیمات مدیریتی به منظور ایجاد توازن بین درآمد حاصل از پردازش یک سفارش خاص و منابع پردازشی این سفارش عمل می‌کنند. پردازش یک سفارش ممکن است موجب تأخیر در دیگر سفارش‌ها و تحمیل هزینه جرمیه به تولیدکننده شود (روم و اسلوتنیک، ۲۰۰۹). بسیاری از شرکت‌ها مایل به پذیرش تمام سفارش‌های با ارزش مثبت هستند، بدون اینکه اثر این کار بر برنامه‌ریزی سفارش‌های پذیرفته شده فعلی را در نظر بگیرند. ریشه این ناهماهنگی، اغلب در تفکیک تصمیم‌گیری پذیرش سفارش توسط واحد فروش از زمان‌بندی است که به طور معمول در دست واحد تولید است. واحد فروش به منظور افزایش فروش سعی می‌کند تا حد ممکن سفارش‌ها را بپذیرد، درحالی‌که تلاش واحد تولید عمل به تاریخ‌های تحویل وعده داده شده است. این تضاد منافع به تأخیر شایان توجه سفارش‌ها، نقض موعد تحویل مقرر یا استفاده بیش از حد عادی از منابع منجر می‌شود؛ بنابراین ضروری است که فرایندهای انتخاب سفارش و زمان‌بندی به صورت یکپارچه در نظر گرفته شوند (هربوتس، هرولن، لئوس و دمپولمیستر، ۲۰۰۷).

تولیدکنندگانی که به صورت سفارشی تولید می‌کنند، از مزایایی مانند کاهش هزینه‌های موجودی و کاهش خطر آسیب‌پذیری منتفع می‌شوند. از سوی دیگر، محدودیت‌های ظرفیت تولید، این شرکت‌ها را به گزینش از بین سفارش‌های مشتریان مجبور می‌کند. سفارش مشتریان به طور معمول دارای موعدی مقرر برای تحویل است و ممکن است برای تحویل دیرتر از موعد تحویل جرمیه‌ای را نیز در نظر بگیرند (سزارت، آگوز و سیبل سلمان، ۲۰۱۲). تصمیمات در زمینه پذیرش یا رد سفارش‌ها، در شرکت‌های با سیستم تولید براساس سفارش نقش مهمی دارند. پرسش اصلی در مورد رد یا قبول سفارش‌هاست. این مسئله به توسعه موضوعی با عنوان پذیرش سفارش منجر شد. پذیرش سفارش نوعی تصمیم‌گیری است که سفارش‌ها را براساس در دسترس بودن ظرفیت کافی برای تکمیل آنها پیش از موعد مقرر تحویل، قبول یا رد می‌کند (سیمونیس، ۲۰۰۶). تمرکز پذیرش سفارش بر تصمیم‌گیری در مورد قبول یا رد سفارش توسط شرکتی است که با تقاضای بیش از حد ورود سفارش‌ها مواجه است (آگوز، سیبل سلمان و بیلگینتورک یالچین، ۲۰۱۰). سفارش‌های مشتریان، به طور نامنظم در طول زمان دریافت می‌شود و تصمیم در مورد پذیرش یا عدم پذیرش، می‌تواند به محض ورود هر سفارش یا پس از ورود

تعدادی از سفارش‌ها در دوره زمانی معینی اتخاذ شود. برای پذیرش یا رد سفارش مشتری، باید ملاحظاتی را در نظر گرفت که مهم‌ترین آنها عبارت‌اند از: ۱. در دسترس بودن منابع و مواد؛ ۲. هزینه و درآمد حاصل از پذیرش سفارش (ایوانسکیو، ۲۰۰۴). علاوه بر این، تحویل دیرتر برخی سفارش‌ها ممکن است به جریمه و کاهش درآمد بینجامد و در درازمدت سبب از دست دادن وجهه شرکت و سهم بازار شود. در بازار رقابتی، اهمیت تحویل به‌موقع ممکن است از دیدگاه هزینه - منفعت موجب عدم پذیرش برخی سفارش‌های مناسب شود (اسلوتنیک و مورتون، ۲۰۰۷).

مسئله پذیرش سفارش و زمان‌بندی هنگامی به‌وجود می‌آید که ظرفیت محدود تولید، تولیدکننده را به انتخاب از بین سفارش‌های ورودی ملزم کند. در این مسئله علاوه بر تصمیم‌گیری در مورد پذیرش یا رد، تعیین برنامه زمانی سفارش‌های پذیرفته‌شده نیز مطرح می‌شود. هدف یافتن مجموعه‌ای بهینه از سفارش‌های پذیرفته‌شده و برنامه زمانی سفارش‌های پذیرفته‌شده نیز مطرح می‌شود. هدف به‌منظور حداکثر کردن درآمد کل تولیدکننده است (اگوز، سیبل سلمان و بیلگینتورک یالچین، ۲۰۱۰). هدف از تعیین موعد تحویل، رسیدن به سطح بالای سرویس (حداکثر تعداد سفارش‌های تکمیل‌شده با توجه به موعد تحویل) است. موعد تحویل را می‌توان به‌عنوان ضرب‌الاجلی در نظر گرفت که هنگام انجام سفارش‌ها نمی‌توان از آن تجاوز کرد (ساویک، ۱۹۹۳).

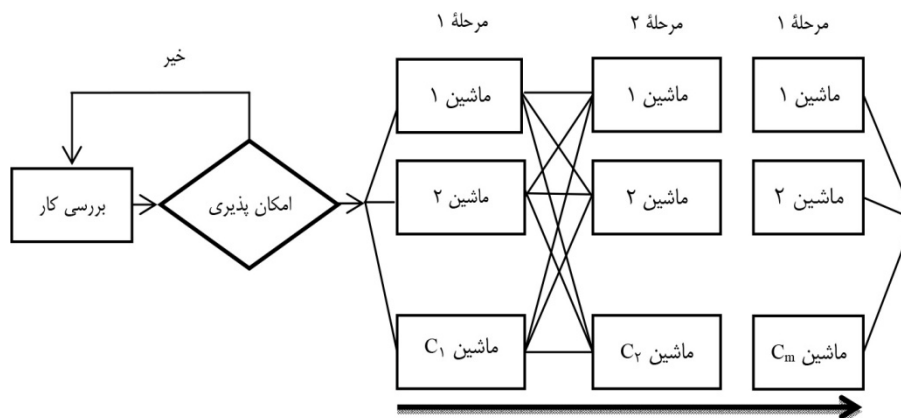
بسیاری از مقالات، با فرض وجود سفارش‌هایی در سیستم، تنها به بحث زمان‌بندی سفارش‌ها پرداخته‌اند و تعداد بسیار کمی یکپارچه‌سازی زمان‌بندی و پذیرش سفارش به‌صورت محدود را بررسی کرده‌اند. این موضوع نشان‌دهنده وجود شکافی است که با توجه به کاربردهای فراوان مسئله یکپارچه پذیرش سفارش و زمان‌بندی، نیاز جدی به افزایش تلاش‌ها در این زمینه تحقیقی احساس می‌شود. شایان ذکر است که استفاده‌کنندگان از این تحقیق کارخانه‌ها و صنایع تولیدی هستند که به‌طور خاص در محیط تولید بر مبنای سفارش مشتری کار می‌کنند.

### پیشینه پژوهش

خط جریان منعطف که معمولاً با نام‌های جریان تولیدی هیبریدی<sup>۱</sup>، جریان تولیدی با ماشین‌های موازی یا تولید جریانی با چند پردازشگر<sup>۲</sup> نیز شناخته می‌شود، خط جریانی با چند ماشین موازی در بعضی یا همه مراحل تولید است؛ بنابراین، تمام محصولات مسیر مشابهی را در سیستم دنبال می‌کنند. در هر یک از مراحل، یکی از دستگاه‌های موازی برای تولید انتخاب می‌شود (کوات و

1. Hybrid Flow Shop  
2. Multiprocessor Flow Shop

کوهن، ۲۰۰۷). در بسیاری از جریان‌های تولیدی ممکن است محدودیتی وجود داشته باشد که پس از پردازش یک کار، فرایندهای متعاقب آن کار باید بدون تأخیر انجام گیرند. در صورت لزوم، شروع پردازش کار را می‌توان در مرحله اول به تعویق انداخت، به طوری که برای پردازش در مرحله (مراحل) بعدی این کار، نیاز به وقفه نباشد. چنین جریان تولیدی را «جریان تولید محدود» یا «جریان تولید بدون وقفه» می‌نامند (جولایی، شیخ، ربانی و کریمی، ۲۰۰۹). شکل ۱ طرح کلی یک خط جریان منقطع بدون وقفه را نشان می‌دهد.



شکل ۱. خط جریان منقطع بدون وقفه

مطالعات در زمینه مسئله یکپارچه پذیرش سفارش و زمان‌بندی بسیار محدودند، ولی در سال‌های اخیر پژوهشگران به تحقیق در این حوزه علاقه‌مند شده‌اند. پوربابایی (۱۹۸۹) اولین بار مدلی را برای کمک به تولیدکنندگان در تعیین چگونگی پذیرش یا رد سفارش‌های ارائه‌شده با در نظر گرفتن به حداکثر رساندن سود خالص عملیاتی ارائه کرد. گوپتا، کیپاریسیس و ایپ (۱۹۹۲) مسئله انتخاب زیرمجموعه‌ای از پروژه‌های همزمان تعیین توالی بهینه با هدف بیشینه‌سازی بازده خالص کل پروژه‌های منتخب را در نظر گرفتند. ده، گوش و ولز (۱۹۹۳) مسئله توسعه‌یافته زمان‌بندی ماشین با زمان‌های پردازش تصادفی و موعد تحویل را در نظر گرفتند که انتخاب سفارش را نیز مورد توجه قرار می‌داد. استاج (۱۹۹۵) مدلی را برای حالت تک‌ماشین مسئله انتخاب سفارش و تعیین توالی، با احتمال قابل پیش‌بینی خرابی ماشین پیشنهاد کرد. اسلوتنیک و مورتون (۱۹۹۶) مدل پذیرش سفارش یک‌دوره‌ای را برای سیستم تک‌مرحله‌ای، با هدف به حداکثر رساندن مجموع سود با جریمه تأخیر وزنی پیشنهاد کردند. ایوانسکیو، فرانسو و برتران

(۲۰۰۲) مسئله پذیرش سفارش را در صنایع فرایند گروهی بررسی کردند. آنها سیاستی ترکیبی را با استفاده از تکنیک‌های تیرید شبیه‌سازی شده، به‌منظور برآورد اثر مستقیم پذیرش سفارش در استفاده از منابع، ارائه کردند.

لوئیس و اسلوتنیک (۲۰۰۲) کار اسلوتنیک و مورتون (۱۹۹۶) را به‌صورت چنددوره‌ای گسترش دادند، با این فرض که انتخاب سفارش‌ها، سفارش‌های آینده را تحت تأثیر قرار می‌دهد. برای مثال هر مشتری که سفارشش رد شود، هرگز باز نخواهد گشت. چارنسیریاساکسول، گریفین و کسکینوچاک (۲۰۰۴) پذیرش سفارش و زمان‌بندی یکپارچه را در مسئله‌ای تک‌ماشینه، با هدف به حداکثر رساندن سود بررسی کردند. آنها یک مدل مختلط عدد صحیح را برای این مسئله به‌دست آوردند و حل کردند.

روندری، چن، چن و کاکائیلدریم (۲۰۰۵) نیز در مطالعه‌ای عدد صحیحی را به‌همراه چند روش ابتکاری برای حل این مدل توسعه دادند. اسلوتنیک و مورتون (۲۰۰۷) پذیرش سفارش و زمان‌بندی در محیطی با یک ماشین را مورد مطالعه قرار دادند. مسئله مورد بررسی، توالی زمان وابسته به زمان‌های راه‌اندازی و موعد تحویل را در برمی‌گرفت. از آنجا که روش دقیق در حل مسائل بزرگ‌تر دچار مشکل می‌شد، آنها چند روش ابتکاری از جمله جست‌وجوی پرتو<sup>۱</sup> و روش‌های ابتکاری کوتاه‌بینانه مبتنی بر تخصیص<sup>۲</sup> را نیز توسعه دادند. رام و اسلوتنیک (۲۰۰۹) الگوریتم ژنتیکی برای همین مسئله ارائه کرده‌اند. علاوه‌بر این، نوییون، هرپوتس و لئوس (۲۰۰۹) دو الگوریتم شاخه و حد و چند روش ابتکاری برای پذیرش سفارش و زمان‌بندی را بدون ضرب‌الاجل‌ها و زمان راه‌اندازی وابسته به توالی در محیط تک‌ماشینه ارائه کرده‌اند. سسارت، اُگوز و سیبل سلمان (۲۰۱۲) الگوریتم جست‌وجوی ممنوعه را برای حل پذیرش سفارش و زمان‌بندی در محیط تک‌ماشینه با دسترسی به کار و زمان راه‌اندازی وابسته به توالی، توسعه دادند. مسئله زمان‌بندی با امکان رد در محیط‌های مختلف، مانند چندپردازنده (بارتال، لئوناردی، مارکتی - اسپاکامدا، اسگال و استوجیه، ۲۰۰۰) و ماشین واحد (ژانگ، لو و یوان، ۲۰۰۹) نیز مطالعه شده است. توابع هدف مورد استفاده به‌طور عمده زمان اتمام و مجموع زمان‌های تکمیل هستند و مدل‌ها با جریمه رد سفارش همراه است.

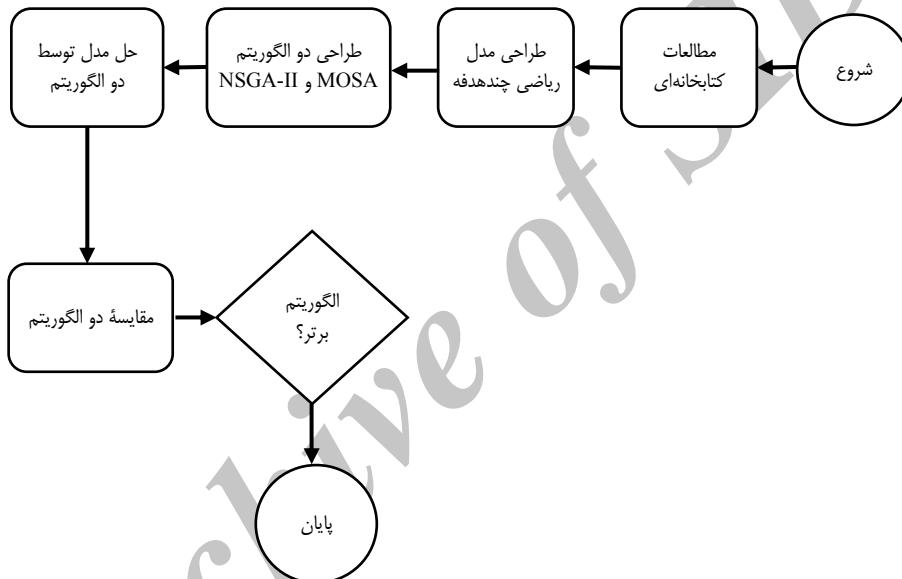
مسائل واقعی، پیش‌بینی ناپذیر و پویا هستند. بیشتر این مسائل غیرخطی و بسیار پیچیده‌اند. به همین دلیل رویکردهای فراابتکاری به‌سبب پوشش محدوده‌ای از مسائل واقعی، راهبرد امیدوارکننده‌تری نسبت به روش‌های دقیق به‌نظر می‌رسند.

1. Beam search
2. Assignment-based myopic heuristics

از طرفی تاکنون این مسئله با اهداف چندگانه متعارض در نظر گرفته نشده که این پژوهش در پی رفع این شکاف نیز است و با ارائه مدلی جدید به همراه روش‌های نوین برای حل این مدل، گامی در این راستا برخواهد داشت.

### روش‌شناسی پژوهش

این پژوهش از نوع توسعه‌ای کاربردی است. داده‌های پژوهش به صورت آزمایشی تولید شدند. مراحل تحقیق در شکل ۲ آمده است.



شکل ۲. مراحل تحقیق

در ادامه متغیرها و مدل‌سازی مسئله تعریف می‌شود.

### تعریف پارامترها

$tp_{i,k}$ : زمان پردازش کار  $i$  در مرحله  $k$ ؛

$s_i$ : زمان ایده‌آل شروع کار  $i$ ؛

$s'_i$ : زمان واقعی شروع کار  $i$ ؛

$W_i$ : سود حاصل از انجام کار  $i$  اگر در  $S_i$  شروع شود. بسته به نوع انحراف مثبت یا منفی از زمان شروع ایده‌آل جریمه دیرکرد  $W_i^+$  یا جریمه زودکرد  $W_i^-$  به تابع هدف تحمیل می‌شود؛  
 $W_i^+$ : جریمه دیرکرد کار  $i$ ؛  
 $W_i^-$ : جریمه زودکرد کار  $i$ ؛  
 $c_k$ : ظرفیت ماشین  $k$  ام؛

$P_{i,k}$ : مجموعه‌ای از فواصل زمانی که در آن کار  $i$  می‌تواند در مرحله  $K$  پردازش شود؛  
 $T$ : افق زمانی به  $T$  دوره زمانی گسسته برابر تقسیم شده و هر دوره زمانی با  $1, 2, \dots, T$  نمایش داده می‌شود.

### متغیرهای تصمیم‌گیری

$y_{i,j}$ : اگر کار  $i$  در دوره زمانی  $j$  شروع شود، برابر  $1$  و در غیر این صورت صفر است؛  
 $u_{i,j,k}$ : اگر کار  $i$  در مرحله  $k$  در دوره زمانی  $j$  شروع شود، برابر  $1$  در غیر این صورت صفر است؛  
 $x_i$ : اگر کار  $i$  پردازش شده باشد،  $1$  و در غیر این صورت صفر است؛  
 $m_i^{\pm}$ : انحراف‌ها از زمان شروع ایده‌آل کارهای انتخاب شده  $i$ ؛  
 $W_i$ : سود حاصل از انجام کار  $i$  است، اگر در  $S_i$  شروع شود. بسته به نوع انحراف مثبت یا منفی از زمان شروع ایده‌آل جریمه دیرکرد  $W_i^+$  یا جریمه زودکرد  $W_i^-$  به تابع هدف تحمیل می‌شود.

### مدل‌سازی مسئله

در این بخش نخست مفروضات مدل و مسئله بیان می‌شود. محیط این تحقیق تولید برای سفارش است. در هر یک از مراحل، کار را می‌توان روی هر ماشین پردازش کرد، اما بیش از یک کار را نمی‌توان همزمان روی یک ماشین پردازش کرد. کارها نمی‌توانند تقسیم شوند یا پیش از اتمام از سیستم خارج شوند. وقتی کاری انتخاب می‌شود، باید پردازش روی آن به‌طور پیوسته از ابتدا تا انتها بدون هیچ‌گونه زمان انتظاری بین مراحل انجام گیرد، به طوری که اختلاف زمان بین اتمام آخرین عملیات و زمان شروع عملیات اول آن، برابر با مجموع زمان کار در تمام ماشین‌ها باشد. بین تعداد روزهای دیرکرد یا زودکرد و جریمه در نظر گرفته شده برای آن، رابطه خطی وجود دارد. یکی از اهداف مدل، حداکثر کردن ارزش کل کارهای پردازش شده است (جولای و همکاران، ۲۰۰۹). مطابق روابط ۱ و ۲ علاوه بر حداکثرسازی سود، حداقل کردن میزان انحراف‌ها از موعد تحویل نیز هدف مدل است. براساس رابطه ۳، در هر مرحله، تعداد کل دوره‌های زمانی اختصاص داده شده به یک کار با زمان پردازش کار در آن مرحله برابر است. براساس رابطه ۴، اگر در هر مرحله در دوره‌های زمانی  $j$  و  $j+1$ ،  $u_{i,j,k} = 1$  و  $u_{i,j+1,k} = 0$  باشد، با مانع شدن از

برقراری رابطه  $u_{i,j,1} = 1$  در دوره زمانی  $l$  بعد از  $1 + j$ ، هر کار در هر مرحله به مجموعه‌ای مجاور از دوره‌های زمانی اختصاص داده می‌شود. رابطه ۵ تعریف دوره زمانی شروع برای هر کار است. اگر فاصله زمانی زمان شروع کار  $i$  باشد،  $y_{i,j} = 1$  خواهد شد و در غیر این صورت، برابر با صفر است. رابطه ۶ اطمینان می‌دهد که بیشترین کار ممکن در هر مرحله در هر دوره زمانی پردازش شوند و بیش از ظرفیت ماشین پردازش صورت نگیرد. مجموعه  $J_j^k$  اجازه می‌دهد تا کارهایی که به صورت بالقوه می‌توانند به فاصله زمانی تخصیص پیدا کنند، انتخاب شوند و به کاهش فضای جست‌وجو کمک می‌کند. روابط ۷ و ۸ بر فرضیه بدون وقفه بودن سیستم تأکید می‌کنند. رابطه ۹ دوره‌های زمانی را که هر کار می‌تواند پردازش شود، محدود کرده است. در رابطه ۱۰،  $n_i^-$  زودکرد و  $n_i^+$  تعداد زمان دیرکرد را برای کار را تعریف می‌کنند. عبارت  $n_i^+ + n_i^-$  انحراف از موعد تحویل را برای هر کار مشخص می‌کند. این انحراف در تابع هدف سود به صورت جریمه و با ماهیت هزینه است و در تابع هدف انحراف با ماهیت تعداد روز وارد می‌شود.

$$\text{Max} \sum_{i=1}^n w_i x_i - w_i^- n_i^- - w_i^+ n_i^+ \quad (\text{رابطه ۱})$$

$$\text{Min} \sum_{i=1}^n n_i^- - n_i^+ \quad (\text{رابطه ۲})$$

$$\text{St:} \quad (\text{رابطه ۳})$$

$$\sum_{i \in P_{i,k}} u_{i,j,k} = tp_{i,k} x_i \quad k = 1, 2, \dots, m, \quad i = 1, 2, \dots, n$$

$$tp_{i,k} u_{i,j,k} - p_{i,k} u_{i,j,k} + \sum_{l=j+2}^{b_{i,k}+tp_{i,k}-1} u_{i,j,k} \leq tp_{i,k} x_i \quad (\text{رابطه ۴})$$

$$k = 1, 2, \dots, m, \quad i = 1, 2, \dots, n, \quad j \in P_{i,k}$$



زمان بندی دوهدفه خطوط جریان منعطف بدون وقفه با.... ۴۵۳

$$(u_{i,j,1} + u_{i,j,2}) - u_{i,j-1,1} = y_{i,j} , \quad \text{رابطه (۵)}$$

$$i = 1, 2, \dots, n, \quad \forall j \in P_{i,k}, \quad j \neq a_{i,1}$$

$$\sum_{i \in J_j^k} u_{i,j,k} \leq c_k \quad k = 1, 2, \dots, m \quad j = 1, 2, \dots, T \quad \text{رابطه (۶)}$$

$$u_{i,j,k} + u_{i,j,k+1} \leq 1 \quad \text{رابطه (۷)}$$

$$k = 1, 2, \dots, m \quad j = 1, 2, \dots, n \quad \forall j \in I_{i,k}$$

$$u_{i,j,k} - (u_{i,j+1,1} + u_{i,j+1,k+1}) \leq 0 \quad \text{رابطه (۸)}$$

$$k = 1, 2, \dots, m \quad j = 1, 2, \dots, n \quad \forall j \in I_{i,k}$$

$$u_{i,j,k} \quad \forall j \in P_{i,k} \quad k = 1, 2, \dots, m \quad i = 1, 2, \dots, n \quad \text{رابطه (۹)}$$

$$\left[ \sum_{j \in P_{i,1}} j y_{i,j} - s_i x_i \right] - n_i^+ - n_i^- = 0 \quad i = 1, 2, \dots, n \quad \text{رابطه (۱۰)}$$

$$x_i, y_{i,j}, u_{i,j,k} \in \{0, 1\} \quad n_i^+ - n_i^- \geq 0 \quad \text{رابطه (۱۱)}$$

### حل مدل

این تحقیق از نوع توسعه‌ای است. مدل این مسئله از نوع غیرخطی است و با افزایش اندازه مسئله، پیچیدگی آن به حدی افزایش می‌یابد که حل آن با روش‌های کلاسیک و ابتکاری ممکن نیست، از این رو در این تحقیق از دو الگوریتم فراابتکاری چندهدفه NSGA II و MOSA استفاده می‌شود. عملکرد مناسب این دو الگوریتم در حل مسائل مختلف اثبات شده است. گردآوری اطلاعات در زمینه ادبیات موضوع و پیشینه پژوهش‌های صورت گرفته در زمینه تحقیق از روش کتابخانه‌ای استفاده شده است.

### الگوریتم ژنتیک چندهدفه

دب و همکاران (۲۰۰۲) الگوریتم بهبودیافته‌ای به نام NSGA II را معرفی کردند. در هر نسل الگوریتم، ابتدا با استفاده از جمعیت والد ( $P_t$ ) و عملگرهای تقاطع و جهش، جمعیت فرزندان ( $Q_t$ ) ایجاد می‌شود. پس از آن دو جمعیت با یکدیگر ترکیب می‌شوند و جمعیت جدید ( $R_t$ ) را تشکیل می‌دهند که این جمعیت جدید به دسته‌های غیرمغلوب مختلف طبقه‌بندی می‌شود. سپس جمعیت جدید به ترتیب با نقاط مختلف از جبهه‌های غیرمغلوب پر می‌شود. پر شدن با اولین جبهه غیرمغلوب شروع می‌شود و با نقاطی از جبهه غیرمغلوب دوم ادامه می‌یابد و این روند تا تکمیل ظرفیت، ادامه می‌یابد. جبهه‌هایی که نمی‌توانند در جمعیت جدید جایگزین شوند، حذف می‌شوند. اگر از یک جبهه تنها تعداد مشخصی نقطه، مجاز به ورود به جمعیت جدید باشند، به‌جای استفاده از نقاط تصادفی از رویه مشخصی به نام مرتب‌سازی بر مبنای فاصله ازدحامی<sup>۱</sup> استفاده می‌شود.

### نحوه نمایش جواب‌ها

با توجه به ماهیت مسئله مورد مطالعه که دربرگیرنده دو مفهوم زمان‌بندی و پذیرش سفارش است، هر جواب به صورت دو رشته مجزا، به مانند شکل ۳ نمایش داده می‌شود.

$S_1$	$S_2$	$S_3$	$S_4$	...	...	$S_{n-2}$	$S_{n-1}$	$S_n$	رشته زمان‌بندی
$X_1$	$X_2$	$X_3$	$X_4$	...	...	$X_{n-2}$	$X_{n-1}$	$X_n$	رشته پذیرش

شکل ۳. نمایش جواب‌ها

قسمت اول نشان‌دهنده زمان شروع هر کار در برنامه زمان‌بندی است. قسمت دوم جواب به صورت صفر و یک است و کارهای پذیرفته‌شده از بین تمامی کارهای ورودی را مشخص می‌کند. قسمت دوم هر جواب در واقع نتیجه به‌کارگیری الگوریتم ابتکاری معینی است که از قسمت اول جواب به‌عنوان ورودی استفاده می‌کند.

### ایجاد زمان‌بندی اولیه

در اولین گام اجرای هر الگوریتم جوابی تصادفی به صورت شکل ۴ تولید می‌شود:

1. Crowded distance sorting

$r_1$	$r_2$	$r_3$	...	...	....	....	$r_{n-2}$	$r_{n-1}$	$r_n$
-------	-------	-------	-----	-----	------	------	-----------	-----------	-------

شکل ۴. زمان بندی اولیه تصادفی

- که در آن  $r_i$  عددی تصادفی در بازه  $[a_{i,1}, b_{i,1}]$  است. سپس یک روش ابتکاری متشکل از مراحل زیر به کار گرفته می شود:
۱. کارها براساس سود به صورت نزولی مرتب شوند؛
  ۲. اولین کار وارد برنامه زمان بندی شده و ماتریس همپوشانی  $n \times n$  (تعداد کل کارهاست) تشکیل داده شود؛
  ۳. برای کارهای بعدی اگر بین کار فعلی و کار (کارهایی) که پذیرفته شده اند، همپوشانی وجود داشته باشد، مقدار عنصر  $i \times j$  ( $i$  کار فعلی و  $j$  کار پذیرفته شده) ماتریس برابر یک قرار داده شود. دو کار  $i$  و  $j$  در هر مرحله همپوشانی دارند، اگر و تنها اگر دو شرط زیر همزمان برقرار باشند:
    - زمان شروع کار  $i$  از مجموع زمان شروع کار  $j$  در مرحله به علاوه زمان پردازش آن در مرحله کمتر باشد.
    - زمان شروع کار  $j$  از مجموع زمان شروع کار  $i$  در مرحله به علاوه زمان پردازش آن در مرحله کمتر باشد.
  ۴. حداکثر مجموع ستون های ماتریس همپوشانی محاسبه و با ظرفیت مراحل مقایسه شود. اگر مقادیر حداکثری در تمامی مراحل از ظرفیت هر مرحله کوچک تر یا مساوی باشد، کار وارد برنامه زمان بندی می شود و مقدار متناظر با آن در قسمت دوم جواب برابر با ۱ در نظر گرفته می شود. اگر حداقل در یکی از مراحل مقدار حداکثر مجموع ستونی از ظرفیت مرحله بیشتر باشد، کار مورد نظر رد می شود و مقدار متناظر با آن در قسمت دوم جواب صفر در نظر گرفته می شود.

### مرتب سازی غیر مغلوب

پس از ایجاد جمعیت جدید که حاصل ترکیب جمعیت والد و فرزند است، با استفاده از روش مرتب سازی غیرمغلوب جوابها به رده های مختلف تقسیم می شوند.

### فاصله ازدحامی

به منظور به دست آوردن تخمینی از چگالی جواب‌ها از معیاری به نام فاصله ازدحامی استفاده می‌شود. برای محاسبه فاصله ازدحامی در هر لایه از جواب‌های غیرمغلوب، مراحل زیر انجام می‌گیرد:

۱. مقدار  $l$  برابر با تعداد اعضای سطح غیرمغلوب قرار داده می‌شود؛
۲. برای هر تابع هدف  $m$  مجموعه  $f_m$  به صورت صعودی مرتب شده و  $I^m$  خوانده می‌شود؛
۳. برای هر تابع هدف، فاصله نقاط ابتدا و انتها برابر  $d_{I_j}^m = \infty$  لحاظ می‌شود و برای نقاط میانی مقدار فاصله ازدحامی از رابطه ۱۲ محاسبه می‌شود:

$$d_{I_j}^m = \frac{f_m^{(I_{j+1}^m)} - f_m^{(I_{j-1}^m)}}{f_m^{max} - f_m^{min}} \quad (\text{رابطه ۱۲})$$

### عملگر انتخاب

برای ایجاد جمعیت فرزندان در هر نسل، والدین باید از جمعیت فعلی انتخاب شوند. الگوریتم NSGA II از سازوکار مشخصی به نام انتخاب رقابت باینری<sup>۱</sup> برای انتخاب والدین استفاده می‌کند. در ابتدا دو جواب تصادفی از جمعیت انتخاب می‌شوند. اگر یکی از این دو جواب در سطح غیرمغلوب بهتری باشد، به عنوان والد انتخاب می‌شود. ولی اگر هر دو جواب در یک سطح باشند، از نظر فاصله ازدحامی با یکدیگر مقایسه شده و جواب با فاصله ازدحامی بیشتر انتخاب می‌شود.

### عملگر تقاطع

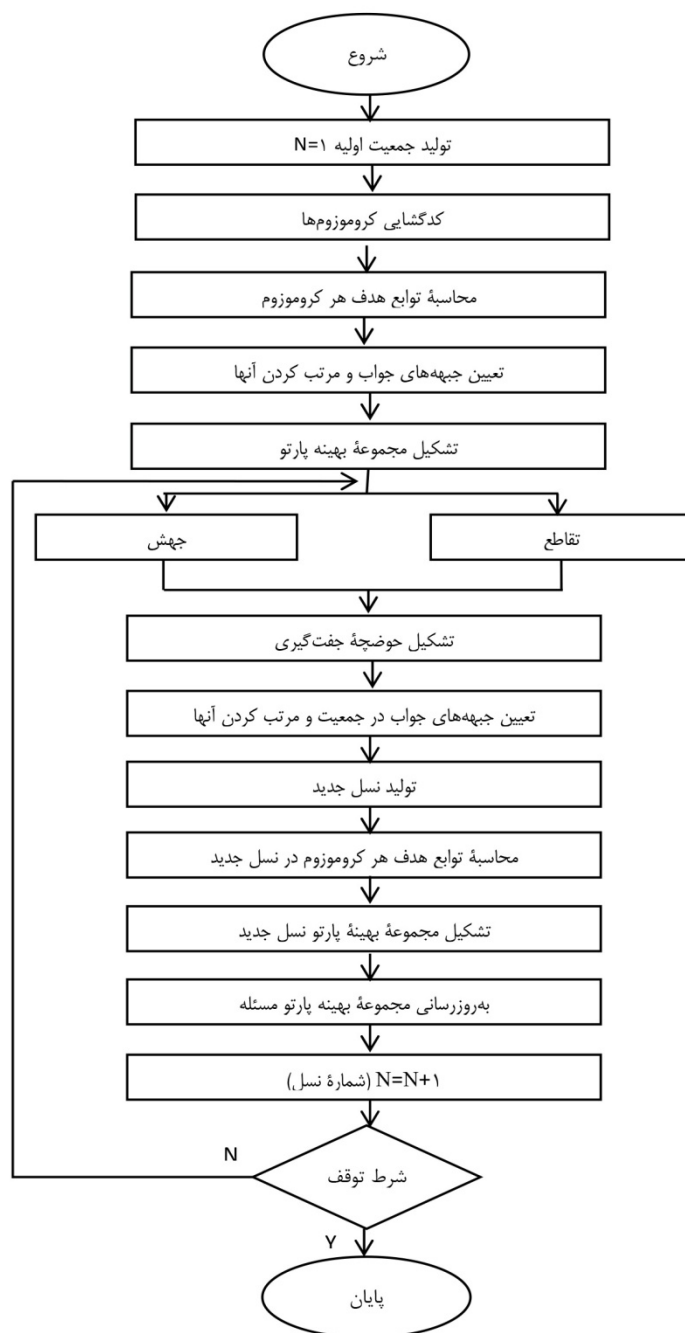
برای استفاده از این عملگر ابتدا با استفاده از عملگر انتخاب دو والد انتخاب می‌شوند. در این تحقیق از عملگر تقاطع تک‌نقطه‌ای<sup>۲</sup> استفاده شده است.

### عملگر جهش

برای به کار بردن عملگر جهش، یک جواب با استفاده از سازوکار انتخاب باینری، انتخاب می‌شود. سپس عدد تصادفی  $l$  بین بازه  $(1, n)$  انتخاب می‌شود و عنصر واقع در خانه  $l$  با استفاده از عملگر تعریف شده تغییر می‌یابد. الگوریتم NSGA-II توسعه داده شده در شکل ۵ نشان داده شده است.

---

1. Binary Tournament Selection  
2. Single point Crossover



شکل ۵. فلوجارت الگوریتم NSGA-II

### الگوریتم SA چندهدفه

منشأ الگوریتم تبرید شبیه‌سازی شده، کارهای کریکپاتریک و سرنی و همکارانشان در سال‌های ۱۹۸۳ و ۱۹۸۵ است. متالورژیست‌ها از تکنیک تبرید تدریجی، برای رسیدن به حالتی که در آن ماده جامد، به‌خوبی مرتب و انرژی آن کمینه شده باشد، استفاده می‌شود (سومان و کومار، ۲۰۰۶). برای حل مسئله بهینه‌سازی، الگوریتم ابتدا از یک جواب اولیه شروع کرده و سپس در یک حلقه تکرار به جواب‌های همسایه<sup>۱</sup> حرکت می‌کند. اگر جواب همسایه بهتر از جواب فعلی باشد، الگوریتم آن را به‌عنوان جواب فعلی قرار می‌دهد و به سمت آن حرکت می‌کند، در غیر این صورت، الگوریتم آن جواب را با احتمالی مشخص که از طریق رابطه متروپولیس<sup>۲</sup> به‌دست می‌آید، به‌عنوان جواب فعلی می‌پذیرد. رابطه ۱۳ این موضوع را به‌خوبی نشان می‌دهد.

$$P(x^{old} \rightarrow x^{new}) = \begin{cases} 1 & \text{if } \Delta > 0 \\ \exp(\Delta f/T) & \text{else} \end{cases} \quad (\text{رابطه ۱۳})$$

در هر دم، چندین تکرار اجرا می‌شود، سپس دم به آرامی کاهش داده می‌شود. در گام‌های اولیه دم خیلی بالا قرار داده می‌شود تا احتمال بیشتری برای پذیرش جواب‌های بدتر وجود داشته باشد. اولین نسخه چندهدفه SA توسط سرافینی در سال ۱۹۹۲ پیشنهاد شد که از روشی تقریباً مشابه الگوریتم تک‌هدفه استفاده می‌کرد. اولونگو و همکاران (۱۹۹۹) الگوریتم چندهدفه تبرید شبیه‌سازی شده (MOSA) را معرفی کردند که روی مسائل بهینه‌سازی ترکیبی چندهدفه آزمایش شده بود. الگوریتم آنها با تنها یک راه‌حل جاری کار می‌کرد، ولی جمعیت جواب‌های غیرمغلوب را در طول جست‌وجو ذخیره می‌کرد (سومان و کومار، ۲۰۰۶). در تحقیق پیش رو نیز از این نسخه چندهدفه استفاده شده است. تنها تفاوت این الگوریتم و الگوریتم تک‌هدفه، در معیار پذیرش جواب‌هاست که به‌صورت رابطه ۱۴ تغییر یافته است:

$$P = \min \left[ 1, \prod_{i=1}^N \exp \left\{ \frac{-\Delta s_i}{T_i} \right\} \right] \quad (\text{رابطه ۱۴})$$

در این رابطه  $\Delta s_i$  برای هر تابع هدف به‌صورت رابطه ۱۵ تعریف می‌شود:

$$\Delta s_i = f_i(S^{new}) - f_i(S^{old}) \quad (\text{رابطه ۱۵})$$

- 
1. Neighbors solution
  2. Metropolis Rule

### همسایگی الگوریتم MOSA

برای حرکت به نقاط همسایه از روش شکل ۶ استفاده می شود که به صورت تجربی به دست آمده است.

---

انتخاب عدد تصادفی  $l$  در بازه  $n$  و  $۱$

انتخاب عدد تصادفی  $K$  در بازه  $۱$  و  $۰$

اگر  $K < ۰/۵$  باشد

انتخاب عدد تصادفی  $b$  در بازه  $۲$  و  $۱$

اضافه کردن مقدار  $b$  به عنصر خانه  $l$

پایان

اگر  $K > ۰/۵$  و  $K < ۰/۸$  باشد

انتخاب عدد تصادفی  $c$  در بازه  $۲$  و  $۱$

کم کردن مقدار  $c$  به عنصر خانه  $l$

پایان

اگر  $K > ۰/۸$  باشد

انتخاب عدد تصادفی  $g$  در بازه  $n-۲$  و  $۱$

تخصیص مقدار تصادفی در بازه پنجره زمان برای عناصر خانه های  $g$  و  $g+۱$

کم کردن یک واحد از مقدار عنصر خانه  $g+۲$

پایان.

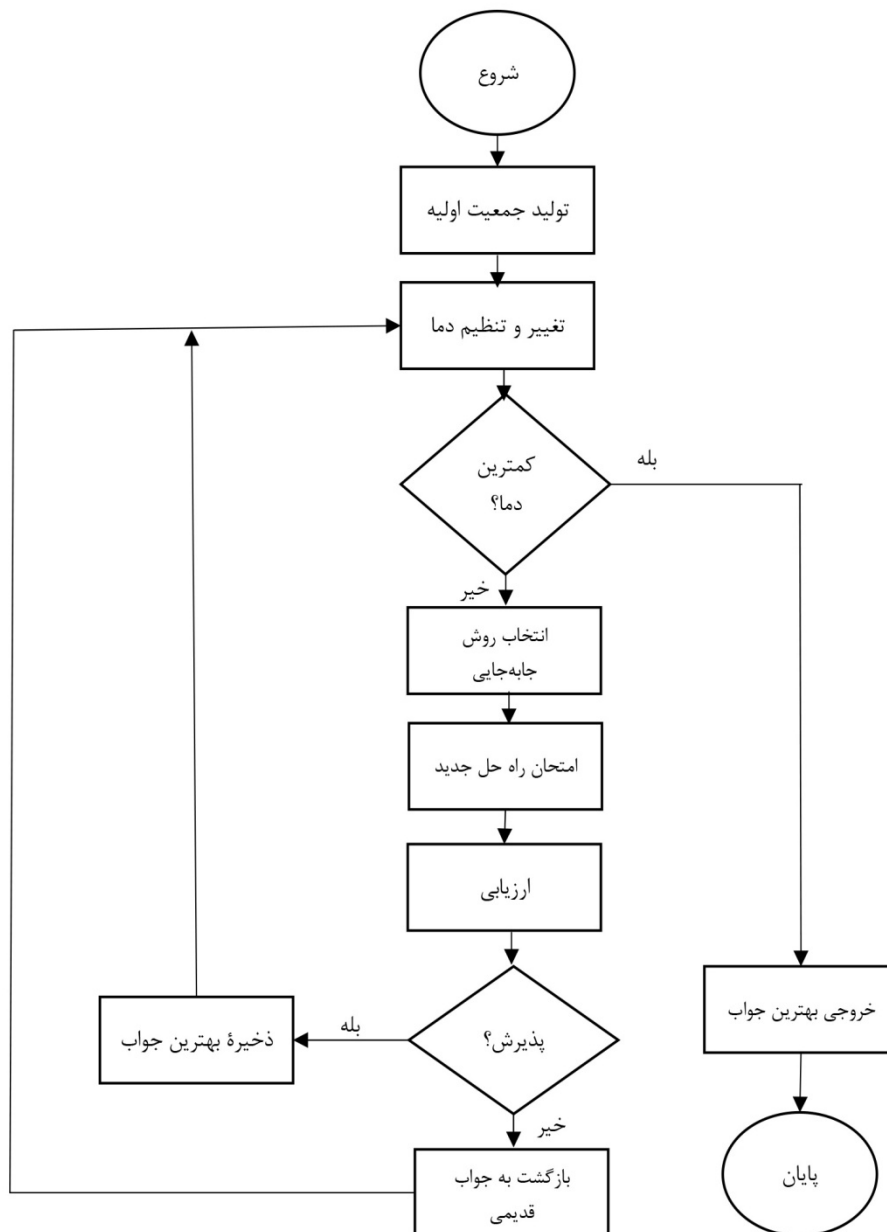
کنترل قرار گرفتن مقدار هر خانه در بازه پنجره زمانی مربوط به سفارش

پایان.

---

### شکل ۶. همسایگی MOSA

فلوچارت الگوریتم مذکور در شکل ۷ آمده است.



شکل ۷. فلوجارت الگوریتم MOSA



### یافته‌های پژوهش

#### نحوه تولید داده‌ها

به منظور بررسی عملکرد الگوریتم‌ها در مسائل با اندازه مختلف، سه دسته کلی از مسائل به صورت جدول ۱ تولید شدند. سپس داده‌های مسائل مطابق جدول ۲ تولید شدند.

جدول ۱. دسته بندی مسائل تولید شده بر اساس اندازه

کوچک						متوسط						بزرگ						اندازه مسئله
۴	۸	۴	۸	۴	۸	۴	۸	۴	۸	۴	۸	۴	۸	۴	۸	۴	۸	تعداد مراحل
۱۰	۱۰	۲۰	۲۰	۳۰	۳۰	۴۰	۴۰	۵۰	۵۰	۶۰	۶۰	۷۰	۷۰	۸۰	۸۰	۹۰	۹۰	تعداد کارها

جدول ۲. نحوه تولید داده‌های مسئله

متغیر	نحوه تولید
زمان شروع ایده آل	$S_i \sim \text{uniform} \left( 1, \frac{n \cdot b_{\text{job}}}{2} \right)$
پنجره زمانی $Time\ window = [a_i \ b_i]$	$a_i \ S_i - \text{uniform} (1, 3)$ $b_i \ S_i - \text{uniform} (1, 3)$
زمان‌های پردازش	$tp_{ij} \sim \text{uniform} (1, 5)$
سود	$W_i \sim \text{uniform} (40, 90)$
جریمه دیرکرد و زودکرد	$W_i^{\pm} \sim \text{uniform} (0, 2)$
تعداد ماشین‌آلات در هر مرحله	$\begin{cases} C_{\text{small-problem}} & \text{uniform} (2, 5) \\ C_{\text{medium-problem}} & \text{uniform} (3, 11) \\ C_{\text{large-problem}} & \text{uniform} (6, 12) \end{cases}$

#### اجرای الگوریتم‌ها

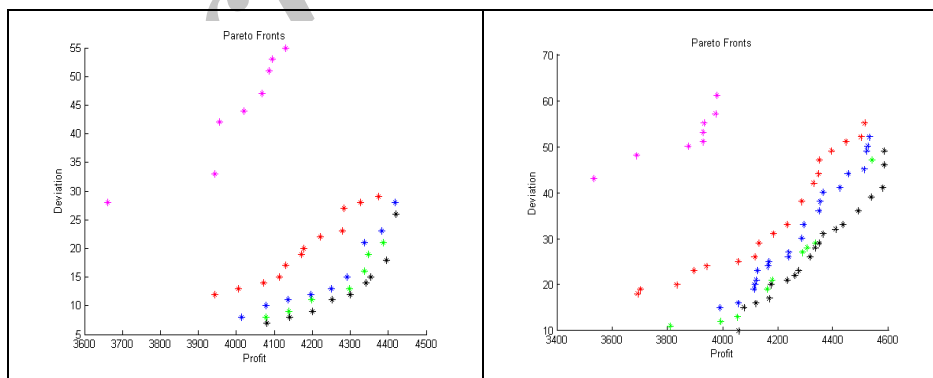
تمامی الگوریتم‌ها با نرم‌افزار Matlab® 2011b نوشته شده و روی سیستمی با پردازنده Dual-Core E5300@ 2.60 GHz و با حافظه ۲ GB اجرا شدند.

### تنظیم پارامترها

پارامترهای الگوریتم‌ها که به‌طور معمول از طریق تجربی تعیین می‌شوند، همراه با علامت اختصاری در جدول ۳ آمده است. با اجرای الگوریتم‌ها، همگرایی برای مسئله‌ای با ۹۰ کار و ۸ مرحله به‌صورت شکل‌های ۸ و ۹ به‌دست آمد.

جدول ۳. مقادیر پارامترهای الگوریتم‌ها

الگوریتم	پارامتر	علامت اختصاری	مسئله کوچک	مسئله متوسط	مسئله بزرگ
NSGA-II	تعداد اعضای جمعیت	PopulationSize	۲۰۰	۲۰۰	۲۰۰
	تعداد نسل‌ها	MaxIt	۱۰۰	۱۵۰	۲۰۰
	احتمال تقاطع	pCrossover	۰/۴	۰/۴	۰/۴
	احتمال جهش	pMutation	۰/۱	۰/۱	۰/۱
MOSA	تعداد اعضای جمعیت	npop	۱	۱	۱
	تعداد حرکت در هر دما	Nmove	۳۰۰	۲۵۰	۲۰۰
	تعداد نسل‌ها	maxit	۴۰۰	۳۵۰	۳۰۰
	بردار دمای اولیه	T0	[۱۰۰۰ ۱۰۰۰]	[۱۰۰۰ ۱۰۰۰]	[۱۰۰۰ ۱۰۰۰]
	فاکتور تبرید	alpha	۰/۹	۰/۹	۰/۹
	دمای آستانه	Tf	۱	۱	۱



شکل ۹. نمودار همگرایی جبهه پارتو الگوریتم NSGA II

شکل ۸. نمودار همگرایی جبهه پارتو الگوریتم MOSA

### مقایسه الگوریتم‌ها

در این تحقیق سعی شد که برای سه دسته مسائل کوچک، متوسط و بزرگ، دو الگوریتم در زمان تقریباً یکسانی اجرا شوند تا مقایسه صورت گرفته بین نتایج از اعتبار بیشتری برخوردار باشد. به منظور مقایسه عملکرد دو الگوریتم از تعدادی شاخص استفاده شده است که در ادامه توضیح داده می‌شوند.

### شاخص‌های مقایسه الگوریتم‌های فراابتکاری

اگر تنوع و نحوه توزیع جواب‌های پارتو مدنظر نباشد، بیشتر بودن تعداد جواب‌ها در جبهه پارتو را می‌توان به عنوان معیار برتری یک الگوریتم در نظر گرفت (زیتزلر، دب و تی‌یل، ۲۰۰۰). معیار فاصله‌گذاری<sup>۱</sup> که توسط اسکات<sup>۲</sup> معرفی شد، نحوه توزیع جواب‌های پارتو را نشان می‌دهد. این معیار با استفاده از رابطه ۱۶ به دست می‌آید. در صورت کوچک بودن این معیار، جواب‌ها به‌طور یکنواخت‌تری توزیع شده‌اند، پس هرچه الگوریتم معیار فاصله‌گذاری کمتری داشته باشد، بهتر است (اسکات، ۱۹۹۵).

$$S = \sqrt{\frac{1}{n} \sum_{i=1}^n (d_i - \bar{d})^2}, \quad d_i = \min \sum_{i=1}^n |f_m^i - f_m^k| \quad \text{رابطه ۱۶}$$

$$k \in n, \quad k \neq i, \quad \bar{d} = \sum_{i=1}^n \frac{d_i}{n}$$

بیشترین گسترش<sup>۳</sup> معیاری است که از مجموع فاصله اقلیدسی نقاط ابتدا و انتهای جبهه پارتو توابع هدف، مطابق رابطه ۱۷ به دست می‌آید. این معیار هرچه بیشتر باشد، بهتر است (زیتزلر، ۱۹۹۹).

$$D = \sqrt{\sum_{m=1}^M \left( \max_n f_m - \min_n f_m \right)^2} \quad \text{رابطه ۱۷}$$

معیار دیگر، متوسط فاصله اقلیدسی نقاط جبهه پارتو از نقطه ایده‌آل است. هرچه این معیار کمتر باشد، الگوریتم کارایی بهتری دارد (کوئلو کوئلو، فان فلدهایزن و لامونت، ۲۰۰۷). با استفاده

1. Spacing
2. Scott
3. Maximum Spread

از الگوریتم ژنتیک تک‌هدفه نقطه ایده‌آل برای هر تابع هدف محاسبه می‌شود و از ترکیب نقاط، مختصات نقطه ایده‌آل به صورت  $(f_1^*, f_2^*)$  به شرح رابطه ۱۸ محاسبه می‌گردد.

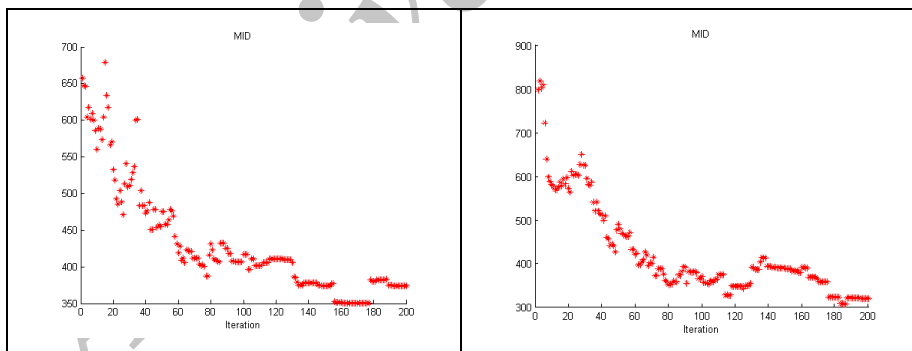
$$MID = \frac{1}{n} \sum_{i=1}^n d_i, d_i = \sqrt{(f_1^i - f_1^*)^2 + (f_2^i - f_2^*)^2} \quad \text{رابطه ۱۸}$$

معیار سطح پوشش<sup>۱</sup> نسبت جواب‌هایی از هر الگوریتم است که توسط جواب‌های الگوریتم دیگر به صورت ضعیف مغلوب می‌شوند. این معیار براساس رابطه ۱۹ محاسبه می‌شود:

$$C(A, B) = \frac{|\{b \in B \mid a \in A: a \leq b\}|}{|B|} \quad 0 \leq C(A, B) \leq 1 \quad \text{رابطه ۱۹}$$

اگر مقدار  $C(A, B)$  برابر ۱ باشد، تمام اعضای  $B$  به صورت ضعیف توسط اعضای  $A$  مغلوب می‌شوند. پس هنگامی که  $C(A, B)$  از  $C(B, A)$  بزرگ‌تر باشد، جواب‌های  $A$  نسبت به  $B$  بهترند (کوئلو کوئلو و همکاران، ۲۰۰۷).

برای بررسی عملکرد الگوریتم‌ها، باید همگرایی آنها را به سمت بهبود جواب در معیارهای مذکور بررسی کنیم. نمودار همگرایی برای معیار MID در شکل‌های ۱۰ و ۱۱ آمده است.



شکل ۱۱. نمودار همگرایی MID الگوریتم NSGA II

شکل ۱۰. نمودار همگرایی MID الگوریتم MOSA

هر مسئله از طریق هر الگوریتم شش بار اجرا شد. برای مقایسه شاخص‌های عملکردی در دو الگوریتم توسعه‌یافته، فرضیاتی مطرح شدند که این فرضیات با استفاده از نرم‌افزار Minitab 16

در سطح اطمینان ۹۰ درصد آزمون شدند. برای مثال عملکرد الگوریتم NSGA II در مقابل الگوریتم MOSA برای مسئله‌ای به اندازه B و در شاخص A به صورت رابطه ۲۰ مورد آزمون فرض قرار گرفت.

$$\begin{cases} H_0: \mu_{Criteria(A),Size(B)}(NSGA - II) = \mu_{Criteria(A),Size(B)}(MOSA) \\ H_1: \mu_{Criteria(A),Size(B)}(NSGA - II) \neq \mu_{Criteria(A),Size(B)}(MOSA) \end{cases} \quad \text{رابطه ۲۰}$$

نتایج آزمون‌ها در جدول ۴ آمده است.

جدول ۴. نتایج آزمون فرضیات عملکرد الگوریتم‌های فرا ابتکاری

شاخص	اندازه مسئله	نوع آزمون	p-value	نتیجه آزمون	عملکرد بهتر
فاصله‌گذاری	در مسائل بزرگ	u	۰/۰۲۰۲	فرضیه رد نمی‌شود	NSGA II
بیشترین گسترش	در مسائل بزرگ	u	۰/۰۳۰۶	فرضیه رد نمی‌شود	MOSA
متوسط فاصله از نقطه ایده‌آل	در کل مسائل	t	۰/۵۵۹	فرضیه رد می‌شود	-
تعداد جواب‌های پارتو	در کل مسائل	t	۰/۰۹۴	فرضیه رد نمی‌شود	MOSA
سطح پوشش	در مسائل بزرگ	u	۰/۰۰۵۱	فرضیه رد نمی‌شود	NSGA II

### نتیجه‌گیری و پیشنهادها

در این تحقیق مدل دوهدفه خطوط جریان منعطف بدون وقفه با پنجره زمانی تحویل و امکان رد کار بررسی شد. برای حل مدل، دو الگوریتم‌های فراابتکاری چندهدفه توسعه داده شدند و عملکرد آنها در دسته‌های متفاوتی از مسائل آزمایش شد. الگوریتم‌ها در سطح اطمینان ۹۰ درصد مقایسه شدند. به طور کلی الگوریتم MOSA تعداد جواب‌های بیشتری را تولید می‌کند، ولی با توجه به عملکرد بهتر الگوریتم NSGA II در مسائل بزرگ، به ویژه در شاخص‌های سطح پوشش و فاصله‌گذاری، می‌توان نتیجه گرفت که در این گونه مسائل کیفیت جواب‌های الگوریتم NSGA II بهتر است. در مسائل بزرگ، الگوریتم MOSA در شاخص بیشترین گسترش عملکرد بهتری دارد و سطح بیشتری از منطقه جواب را تحت پوشش خود قرار می‌دهد. عملکرد دو الگوریتم توسعه‌داده شده، در مسائل کوچک و متوسط، به جز در شاخص تعداد جواب‌های پارتو اختلاف معناداری ندارد.

از نتایج این تحقیق می‌توان در صنایع پتروشیمی و فولاد و به طور کلی در صنایع سفارش‌محور استفاده کرد، زیرا به دلیل ماهیت فرایندهای مورد استفاده، هیچ‌گونه وقفه‌ای در بین

مراحل تولید مجاز نیست، براساس نتایج این تحقیق توصیه می‌شود هدف جدیدی با عنوان میزان انحراف‌ها از موعد تحویل در نظر گرفته شود. در این حالت مقدار سود نسبت به حالت تک‌هدفه کمی کاهش می‌یابد، ولی این کاهش را می‌توان به‌عنوان سرمایه‌گذاری پنهانی در نظر گرفت که در درازمدت با جلب رضایت مشتریان و بهبود وجهه شرکت در بازار، به‌صورت سود بازگشت پیدا خواهد کرد. همچنین به‌جای تصمیم‌گیری جزیره‌ای و مجزای پذیرش سفارش و زمان‌بندی، یکپارچه‌سازی تصمیم‌گیری همزمان این دو بخش به کاهش هزینه‌ها و افزایش بازده سیستم کل منجر می‌شود. در واقع یکپارچه‌سازی این دو سیستم سبب هم‌افزایی می‌شود. این کار در تحقیقات محققان قبلی صورت نگرفته بود و مسائل با نگاه جزیره‌ای صرفاً یک هدف را در نظر گرفته بودند، از این رو نتایج این تحقیق به‌سبب لحاظ کردن اهداف متعدد و متناقض به دنیای واقعی نزدیک‌تر است و پاسخ آن در عمل دقیق‌تر است.

سیستم‌هایی در صنعت وجود دارند که در آنها یک تأخیر زمانی ثابت بین مراحل الزامی است. این تأخیر ممکن است به‌دلیل بازرسی و حمل‌ونقل باشد، از این رو در تحقیقات آتی می‌توان زمان ثابتی را بین اتمام پردازش در هر مرحله و آغاز پردازش در مرحله بعدی در نظر گرفت. بهبود الگوریتم‌های ارائه‌شده و استفاده از سایر الگوریتم‌های فراابتکاری نیز مانند جست‌وجوی ممنوعه و رقابت استعماری برای حل این مسئله در تحقیقات آینده قابل بررسی است.

## References

- Bartal, Y., Leonardi, S., Marhetti-Spaccamela, A., Sgall, J. & Stougie, L. (2000). Multi-processor scheduling with rejection. *SIAM Journal on Discrete Mathematics*, 13(1):64–78.
- Cesaret Bahriye, C., Oguz Ceyda, O. & Sibel Salman, F. (2012). A Tabu Search Algorithm for Order Acceptance and Scheduling. *Computers & Operations Research*, 39(6):1197–1205.
- Charnsirisaksul, K., Griffin, P. & Keskinocak, P. (2004). Order selection and scheduling with lead time flexibility. *IIE Transactions*, 36:697–707.
- Coello Coello, C.A. Van Veldhuizen, D.A. and Lamont, G.B. (2007). *Evolutionary Algorithm for solving Multi-objective problems*, Kluwer Academic Publishers, New York, second edition, May, ISBN, 0-7062-6364-7,4004.
- De, P., Ghosh, J.B. & Wells, C.E. (1993). Job selection and sequencing on a single machine in a random environment. *European Journal of Operational Research*, 70: 425–431.

- Deb, K., Pratap, A., Agarwal, S. & Meyarivan, T. (2002). A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Evolutionary Computation*, 6(2): 182-197.
- Gupta, S.K., Kyparisis, J. & Ip, C.M. (1992). Note—Project selection and sequencing to maximize net present value of the total return. *Management Science*, 38:751–752.
- Herbots, J., Herroelen, W. S., Leus, R. & Demeulemeester, E. L. (2007). Dynamic Algorithms for Order Acceptance and Capacity Planning within a Multi-Project Environment. *In proceedings of the 3rd Multidisciplinary International Conference on Scheduling: Theory and Applications (MISTA 2007)*, 28 -31 August 2007, Paris, France, 572-574.
- Ivanescu, V.C. (2004). *Order acceptance under uncertainty in batch process industries*. PhD thesis, Virginia Cristina, Technische Universities Eindhoven.
- Ivanescu, V.C., Fransoo, J.C. & Bertrand, J.W.M. (2002). Makespan estimation and order acceptance in batch process industries when processing times are uncertain. *OR Spectrum*, 24:467–495.
- Jolai Fariborz, F., Sheikh, ShayaSh., Rabbani, Massoud M. & Karimi, Behrooz. (2009). A genetic algorithm for solving no-wait flexible flow lines with due window and job rejection. *International Journal of Advanced Manufacturing Technology*, 42: 523–532.
- Lewis, H.F. & Slotnick, S.A. (2002). Multi-period job selection: planning workloads to maximize profit. *Computers & Operations Research*, 29:1081–1098.
- Nobibon, F. T, Herbots, J and & Leus, R. (2009). Order acceptance and scheduling in a single-machine environment: exact and heuristic algorithms. *Proceedings of the 4th Multidisciplinary International Scheduling Conference: Theory and Applications (MISTA 2009)*, 10-12 Aug 2009, Dublin, Ireland, 772-774, 2009.
- Oguz Ceyda, O., Sibel Salman, F. & Zehra Bilgintürk Yalçın, Z. (2010). Order acceptance and scheduling decisions in make-to-order systems, *International Journal of Production Economics*, 125(1):200-211.
- Pourbabai, B. (1989). A short term production planning and scheduling model. *Engineering Costs and Production Economics*, 18:159–167.
- Quadt, D. & Kuhn, H. K. (2007). Batch scheduling of jobs with identical process times on flexible flow lines. *International Journal of Production Economics*, 105(2):385–401.
- Rom, W. & Slotnick, S.A. (2009). Order acceptance using genetic algorithms. *Computers and Operations Research*, 36:1758–1767.

- Roundry, R., Chen, D., Chen, P. & Cakanyildirim, M. (2005). Capacity-driven acceptance of customer orders for a multi-stage batch manufacturing system: Models and algorithms. *IIE Transactions*, 37:1093–1106.
- Sawik, T. (1993). A scheduling algorithm for flexible flow lines with limited intermediate buffers. *Applied Stochastic Models and Data Analysis*, 9:127–138.
- Schott, J. (1995). *Fault tolerant design using single and multicriteria genetic algorithm optimization*. Master's thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology.
- Simonis, B.J. (2006). *Order Acceptance in Multipurpose Batch Process Industries Using the Regression Policy*, Master's thesis. Technische Universiteit Eindhoven Department of Mathematics and Computer Science.
- Slotnick, S.A. & Morton, T.E. (1996). Selecting jobs for a heavily loaded shop with lateness penalties, *Computers & Operations Research*, 23:131–140.
- Slotnick, S.A. & Morton, T.E. (2007). Order acceptance with weighted tardiness, *Computers & Operations Research*, 34: 3029–3042.
- Stadje, W. (1995). Selecting jobs for scheduling on a machine subject to failure. *Discrete Applied Mathematics*, 63:257–265.
- Suman, B. & Kumar, P. (2006). A survey of simulated annealing as a tool for single and multiobjective optimization, *Journal of the Operational Research Society*, 57:1143–1160.
- Zhang, L., Lu, L. & Yuan, J. (2009). Single machine scheduling with release dates and rejection. *European Journal of Operational Research*, 198:975–978.
- Zitzler, E. (1999). *Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications*. PhD thesis, Swiss Federal Institute of Technology (ETH) Zurich, Switzerland, TIK-Schriftenreihe Nr. 30, Diss ETH No. 13398, Shaker Verlag, Aachen, Germany.
- Zitzler, E., Deb, K., Thiele, L. (2000). Comparison of multiobjective evolutionary algorithms: Empirical results, *Evolutionary Computation journal* 8(2): 125–148.