حل مسئله بهینه‌سازی زمانبندی کامیون‌ها در انبار متقاطع چنددَربی با در نظر گرفتن اثر یادگیری و زوال‌پذیری کارها با استفاده از روش بهینه سازی مهندسی اجتماعی

مقاله علمی- پژوهشی

سیدایمان سیدی*، استادیار، گروه مهندسی صنایع، دانشگاه پیام نور، تهران، ایران
مریم حامدی، استادیار، گروه مهندسی صنایع، دانشگاه پیام نور، تهران، ایران
رضا توکلی‌مقدم، استاد، دانشکده مهندسی صنایع، پردیس دانشکده‌های فنی، دانشگاه تهران، تهران، ایران
*پست الکترونیکی نویسنده مسئول: iman_seyyedi@pnu.ac.ir

دریافت: ۱۴۰۰/۰۷/۲۳- پذیرش: ۱۴۰۱/۰۲/۰۵
صفحه ۲۰۶-۱۸۳

چکیده

به طور کلی هر زنجیره‌ی تامین شامل سه مرحله‌ی اصلی تهیه، تولید و توزیع است. استفاده از سیستم انبار متقاطع یک استراتژی جدید در مرحله توزیع برای بهبود زمان پاسخگویی به مشتریان با انتقال محصولات به طور مستقیم از کامیون‌های دریافت به کامیون‌های ارسالی است. به طور کلی برای پردازش یک فعالیت، هر دو منبع ماشین و منابع انسانی مورد نیاز است. بسیاری از محققان تاکنون روش‌های برنامه‌ریزی متعددی برای سیستم‌های انبار متقاطع توسعه داده اند، اما اکثراً محدودیت‌های مهم منابع انسانی را نادیده گرفته اند. در این مقاله برای اولین بار به بررسی مسئله زمانبندی کامیون‌ها در انبار متقاطع چند دربی با در نظر گرفتن اثرات عوامل انسانی و زوال پذیری کارها برای پر کردن شکاف بین مدل‌های برنامه‌ریزی نظری و آنچه در دنیای واقعی انجام می‌گیرد پرداخته‌ایم و برای این منظور یک مدل برنامه‌ریزی عدد صحیح مختلط برای مسئله یاد شده ارایه شده است. با توجه به ادبیات تحقیق زمان حل مدل ارایه شده توسط روش‌های دقیق با افزایش اندازه مساله به سرعت افزایش می یابد تا حدی که روش‌های دقیق به سختی می‌تواند به جواب بهینه دست پیدا کنند. برای حل مسائل در مقیاس بزرگ از چهار الگوریتم فراابتکاری شامل الگوریتم‌های ژنتیک (GA)، رقابت استعماری (ICA)، کشتل (KA) و بهینه سازی مهندسی اجتماعی (SEO) استفاده شده است. در نهایت نتایج عددی به دست آمده از تمامی الگوریتم‌های فراابتکاری مورد بررسی و تحلیل حساسیت قرار گرفته‌اند. الگوریتم‌های فراابتکاری را بر اساس معیارهای بهترین، میانگین جواب‌ها، Rpd و زمان مورد مقایسه قرار داده‌ایم. در نتیجه الگوریتم‌های SEO و الگوریتم کشتل از نظر کیفیت جواب بهتر از سایر الگوریتم‌ها عمل نمودند.

واژه‌های کلیدی: انبار متقاطع، زمانبندی، اثر یادگیری، زوال‌پذیری، الگوریتم فراابتکاری

۱- مقدمه

برآورده ساختن نیاز مشتری در مکان و زمان مناسب و با کمترین هزینه را می‌توان از مهمترین اهداف مدیریت زنجیره تامین برشمرد. به طور کلی هر زنجیره‌ی تامین شامل سه مرحله‌ی اصلی تهیه، تولید و توزیع است که هرکدام تسهیلات بسیاری را شامل می‌شود. مراکز توزیع نقش مهمی در مرحله توزیع دارند که این مراکز نه تنها نقطه تلاقی جمع آوری و تحویل محصولات هستند، بلکه وسیله اصلی برآوردن تقاضای مشتریان هستند (ژائو و چنگ ،۲۰۰۹). در محیط توزیع امروز، فشار برای ایجاد کارآیی بیشتر

است. شرکت‌ها با کاهش موجودی در هر مرحله از عملیات، از جمله توزیع، سعی در کاهش هزینه‌ها دارند. عملیات مورد نیاز در یک مرکز توزیع دارای پنج عملکرد اصلی دریافت، مرتب‌سازی، ذخیره‌سازی، انتخاب و ارسال است. اگر همکاری این پنج بخش بهبود یابد، هزینه‌ها کاهش و کیفیت و بهره‌وری افزایش می‌یابد (بارتولدی،۲۰۰۰). بهترین راه برای کاهش هزینه و بهبود کارائی، واقعاً بهبود یک عملکرد نیست بلکه حذف آن عملکرد در صورت امکان است. یک استراتژی نوآورانه انبارداری که پتانسیل بسیاری برای کنترل هزینه‌های لجستیک و توزیع و در عین حال افزایش سطح خدمات به مشتری دارد، انبار متقاطع می‌باشد. یکی از مهمترین تصمیم‌های عملیاتی در بحث انبار متقاطع که به طور مداوم از دهه‌های گذشته مورد توجه محققان بوده است بحث برنامه‌ریزی کامیون‌ها می‌باشد (ویسیتیپانیچ و هنگمچای، ۲۰۱۷). به طور کلی مسائل زمان‌بندی به صورت تخصیص منابع در طول زمانی مشخص برای انجام مجموعه‌ای از وظایف مربوط به یک فرآیند تعریف می‌شود. به صورت خاص زمان‌بندی وسایل حمل و نقل در سیستم انبار متقاطع را می‌توان به فرآیندهای تخلیه وسایل حمل ورودی و بارگیری وسایل حمل خروجی تقسیم کرد که معمولاً بین آنها یک فاصله زمانی برای انجام فعالیت‌های درون انبار از قبیل شناسایی و تشخیص محموله، دسته‌بندی و انتقال از جایگاه دریافت به سمت جایگاه ارسال وجود دارد. هدف از مسائل زمانبندی تصمیم‌گیری در مورد توالی کامیون‌های ورودی و خروجی در سیستم انبار متقاطع می‌باشد. اگر برنامه زمانی کامیون‌ها به خوبی برنامه‌ریزی شده باشد می‌تواند به طور قابل توجهی زمان عملیات در سیستم انبار متقاطع را کاهش دهد. به طور کلی برای پردازش یک فعالیت، هر دو منبع ماشین و منابع انسانی مورد نیاز است. بسیاری از محققان تاکنون روش‌های برنامه‌ریزی متعددی برای سیستم های انبار متقاطع توسعه داده‌اند، اما اکثراً محدودیت‌های مهم منابع انسانی را نادیده گرفته‌اند. برنامه‌ریزی منابع باید ویژگی‌های منحصر به فرد ذاتی ماشین‌ها و منابع انسانی را توام در نظر بگیرند تا برنامه‌ای بهینه ارایه دهد. این سیستم‌ها به عنوان محدودیت منابع دوگانه شناخته می‌شوند و زمانی که محدودیت‌های ظرفیت از دستگاه و اپراتورهای انسان بوجود می‌آیند، رخ می‌دهند (ژو و همکاران، ۲۰۱۱). با این حال، در مسائل زمانبندی که تاکنون برای این مساله ارایه شده است، محدودیت‌های ناشی از منابع انسانی نادیده گرفته شده است و تنها ماشین‌ها را به عنوان یک منبع محدود در نظر گرفته‌اند. در حالیکه، توانایی یک سیستم انبار متقاطع و موفقیت آن بیش از آنکه به میزان در دسترس بودن منابع بستگی داشته باشد به قابلیت منابع بستگی دارد. بنابراین، این خصوصیات انسانی منحصر به فرد باید مورد بررسی قرار گیرد. اپراتورهای انسانی، بر خلاف ماشین آلات، قادر به یادگیری و به دست آوردن مهارت‌های جدید هستند. در ادبیات تحقیق مساله سیستم انبار متقاطع معمولاً زمان پردازش به عنوان یک پارامتر ثابت در نظر گرفته می‌شود، بدون توجه به این واقعیت که معمولاً زمان پردازش هر کار به موقعیت آن کار در توالی، زمان شروع آن و یا هر دو آنها بستگی دارد. در زمان بندی سیستم‌های تولیدی، پردازش یک کار در موقعیت‌های بعدی (تکرارهای بعدی) توسط نیروی انسانی می‌تواند منجر به کاهش زمان انجام کارها گردد. این پدیده به‌عنوان «اثر یادگیری» در ادبیات شناخته می‌شود. به علت اثرات یادگیری زمان پردازش مشاغل کمتر از زمان پردازش طبیعی آنها است. از آنجایی که در اکثر سیستم‌های انبار متقاطع عملیات تخلیه و بار گذاری توسط نیروی انسانی انجام می‌گیرد، با در نظر گرفتن ماهیت انسان و با توجه به مطالب فوق الذکر ما می‌توانیم اثر یادگیری را در اینگونه از مسائل نیز مورد بررسی قرار دهیم. این بدان معنی است که زمان لازم برای تخلیه و بارگیری، بر اساس تعداد کارهایی که قبلاً انجام می شود، کاهش می‌یابد. در زمان‌بندی سیستم‌های تولیدی، هر گونه تأخیر در شروع پردازش کارها می‌تواند منجر به افزایش زمان انجام کارها گردد. این پدیده به عنوان «زوال‌پذیری کارها» در ادبیات شناخته می‌شود. گوپتا و گوپتا (**۱۹۸۸**) برای اولین بار مفهوم زوال پذیری کارها را به طور مستقل در مسائل زمان بندی معرفی کردند. از آن زمان به بعد، مدل‌های مربوط به زمان بندی با زوال پذیری کارها به طور گسترده مورد مطالعه محققان قرار گرفت. با بررسی‌های صورت گرفته در ادبیات انبارهای متقاطع تاکنون در هیچ از تحقیقات صورت گرفته اثرات زوال‌پذیری در نظر گرفته نشده است.

بویسن و همکاران (۲۰۱۰) ثابت کردند که مسأله‌ی برنامه‌ریزی کامیون‌ها در یک سیستم انبار متقاطع که دارای یک درب ورودی و یک درب خروجی است قویاً ***NP-hard*** میباشد. بنابراین، می‌توان نتیجه گرفت که برنامه‌ریزی کامیون در

یک سیستم انبار متقاطع چند دربی نیز ***NP-hard*** می‌باشد. از ان جایی که این مسائل از نوع مسائل پیچیده می باشند و زمان حل بهینه آن‌ها در رده زمان نمایی است در ادبیات تحقیق برای حل مسائل از الگوریتم‌های فراابتکاری استفاده شده است. در این مقاله از الگوریتم‌های فراابتکاری قدیمی مورد استفاده در ادبیات تحقیق مانند **GA** [۱] و **ICA** [۲] برای مساله مورد نظر و همچنین از الگوریتم‌های جدید برای بررسی کارایی آنها در این مساله مانند الگوریتم‌های **KA**[۳] و **SEO**[۴] استفاده شده است و در نهایت نتایج عددی به دست آمده از تمامی الگوریتم‌های فراابتکاری مورد بررسی و تحلیل حساسیت قرار گرفته است.

تاکنون مطالعه مسئله زمان بندی کامیون‌ها در سیستم انبار متقاطع با در نظر گرفتن اثرات همزمان زوال پذیری و یادگیری علیرغم اینکه پدیده زوال پذیری کار و اثر یادگیری ممکن است در بسیاری از شرایط زمان‌بندی واقعی به صورت همزمان وجود داشته باشند، مورد توجه محققان قرار نگرفته است. بنابراین، در این مقاله برای اولین بار به بررسی مسئله بهینه‌سازی برای زمانبندی کامیون‌ها در انبار متقاطع چند دربی با در نظر گرفتن عوامل انسانی و زوال پذیری کارها برای پر کردن شکاف بین مدل‌های برنامه ریزی نظری و آنچه در دنیای واقعی انجام می گیرد پرداخته‌ایم و برای این منظور یک مدل برنامه‌ریزی عدد صحیح مختلط برای مسئله یاد شده ارایه شده است. در ادامه این مقاله مروری بر ادبیات موضوع خواهیم داشت و مطالعات پیشین در رابطه با موضوع بحث، مورد بررسی قرار خواهد گرفت. سپس مسأله مورد تحقیق و فرضیات مسأله، به طور جامع معرفی و مدل ریاضی ارائه شده کاملا تشریح خواهد شد. سپس به بررسی روش‌های حل پشنهادی پرداخته شده است. در بخش چهارم به بررسی و تجزیه و تحلیل نتایج و عملکرد روش‌های پیشنهادی پرداخته شده است. در انتها نیز به جمع بندی و نتیجه‌گیری تحقیق می‌پردازیم. همچنین پیشنهاداتی جهت تحقیقات آتی بیان خواهد شد.

۲- پیشینه تحقیق

یکی از مهمترین مطالعات صورت گرفته در این زمینه می توان به کار ارایه شده توسط یو (۲۰۰۲) اشاره نمود. او مسائل متفاوتی از زمانبندی کامیون‌ها را مطرح نمود که هر مسئله متشکل از پنج عنصر کلیدی بود. تعداد درب‌های ورودی، الگوی قرارگیری کامیون‌ها در مقابل درب ورودی، وجود یا عدم وجود انبار موقت، تعداد درب‌های خروجی و الگوی قرارگیری کامیون‌ها در مقابل درب خروجی. بطور کلی بیشتر مطالعاتی که در زمینه زمانبندی و تعیین توالی کامیون‌ها در انبار متقاطع انجام شده است بر روی کمینه کردن زمان تکمیل آخرین کامیون خروجی و تعیین سطح موجودی به عنوان تابع هدف تمرکز دارند. در این موارد می توان به کارهای ارایه شده توسط سانگ و چن (۲۰۰۷) یو و اگبلو (۲۰۰۸)، عربانی و همکاران (۲۰۱۱)، مکنون و باپتیست (۲۰۱۰)، وحدانی و زندیه (۲۰۱۰)، آلپان و همکاران (۲۰۱۱)، بلنگر و همکاران (۲۰۱۳)، سیدی و همکاران (۲۰۲۱) و غیره اشاره نمود.

یکی از مهمترین مطالعات در این زمینه توسط یو و اگبلو (۲۰۰۸) انجام پذیرفت. آنها یک مدل ریاضی برای یک سیستم انبار متقاطع، با یک درب ورود و یک درب خروج و همچنین یک انبار موقت در جلوی درب خروجی، ارایه داده‌اند. هدف این مطالعه یافتن بهترین توالی برای کامیون‌های ورودی و خروجی بود به نحوی که زمان خروج آخرین کامیون از بارانداز ارسال به حداقل برسد. انبار موقت به این جهت کارایی دارد تا محصولاتی که از کامیون ورودی آورده شده‌اند ولی کامیون خروجی (که مشغول بارگیری است) نیازی به آن‌ها ندارد، در این انبار ذخیره شوند تا به کامیون مناسب خود اختصاص یابند. آنها برای حل مدل پیشنهادی نه روش ابتکاری ارایه دادند و کارآیی روش‌های ابتکاری پیشنهاد شده را به وسیله‌ی مقایسه‌ی آن‌ها با جواب‌های دقیق به دست آمده از روش شمارش کامل نشان داده‌اند. شایان ذکر است که این مقاله در سال‌های بعد مورد توجه محققان قرار گرفت، به طوری که تعدادی از کارهای بعدی با الگو قرار دادن این کار به مطالعات در این زمینه ادامه دادند. بویسن و فیلندر (۲۰۱۰) به مرور مساله زمان‌بندی انبار متقاطع پرداخته‌اند. آن‌ها مسائل زمان‌بندی را از سه جهت مورد بررسی قرار داده‌اند: درب‌ها، مشخصات عملیاتی و توابع هدف. از طرفی تمام مسائل زمان‌بندی کامیون‌ها با علامت‌های $[\alpha|\beta|\gamma]$ نشان داده می‌شوند علامت α نشان دهنده‌ی حالت درب‌ها است که به ٤ صورت E، M، EM ، G صورت تقسیم می‌شوند. E، نوعی درب است یا مختص کامیون ورودی یا خروجی، M، دربی است که مختص هر دو کامیون ورودی و خروجی است، EM شامل هر دو درب

E و M می‌شود و در حالت G، تخصیص دربها به کامیون‌ها از قبل مشخص است. ممکن است در مقابل هر یک از علائم فوق یک عدد نیز ثبت می‌شود که نشان دهنده‌ی تعداد دربها در مساله است. از طرفی β که نشان دهنده‌ی مشخصات عملیاتی است، به ۹ بخش کلی تقسیم گردیده است که می‌توان آن‌ها را به ترتیب با β_1 تا β_9 نشان داد. وحدانی و زندیه (۲۰۱۰) با در نظر گرفتن مفروضات مدل یو و اگبلو (۲۰۰۸) پنج روش فراابتکاری را جهت حل مسائل بزرگ مورد بررسی قرار داده‌اند که این روش‌ها عبارتند از: **الگوریتم ژنتیک، جستجوی ممنوع، آنلینگ شبیه سازی شده،** الگوریتم الکترومغناطیس و روش جستجوی همسایگی.

لازم به ذکر است که نگارندگان از تکنیک روش‌شناسی سطح پاسخ برای طراحی استوار الگوریتم‌های فراابتکاری اشاره شده استفاده کرده‌اند. آلپان و همکاران (۲۰۱۱) یک مسأله ی زمانبندی را مورد مطالعه قرار داده که بنابر اشاره‌ی نویسندگان دارای دو تفاوت مهم با کارهای ارائه شده‌ی پیش از آن است: الف) تابع هدف مدنظر از جنس هزینه بوده و به زمان ارتباط ندارد، ب) توقف در عملیات بارگذاری مجاز است. از مفروضات مهم این مسأله می‌توان به این نکته اشاره نمود که در این تحقیق کامیون‌های ورودی براساس سیاست **FIFO** به دربها تخصیص داده می‌شوند و از طرف دیگر هزینه‌های مدنظر مستقل از زمان هستند. در این مسأله زمان‌بندی تنها برای کامیون‌های خروجی مورد مطالعه قرار می‌گیرد زیرا توالی کامیون‌های ورودی براساس برنامه‌های تولید از پیش مشخص است. کمینه‌کردن مجموع هزینه‌های موجودی انبار موقت و هزینه‌ی جابجایی کامیون‌های خروجی (در صورتی که یک کامیون نوبت خود را به کامیون دیگری برای بارگیری واگذار کند) هدف مسأله‌ی فوق الذکر است که جهت دستیابی به آن از یک برنامه ریزی پویا با بکارگیری یک حد برای جواب بهینه استفاده شده است.

در کاری از موسوی و همکاران (۲۰۱۳) یک مدل برنامه‌ریزی عدد صحیح مختلط دو مرحله‌ای برای مکان‌یابی مراکز انبار متقاطع و برنامه‌ریزی مسیریابی وسایل نقلیه در شبکه‌های توزیع ارایه نمودند. آنها الگوریتم شبیه سازی تبرید و جستجوی ممنوع را برای حل مدل استفاده کردند. آنها بطور تصادفی چندین مسأله را تولید کردند و نشان دادند که عملکرد تبرید شبیه‌سازی شده ترکیبی پیشنهادی موثر و به سرعت به راه حل‌های معقول همگرا می‌گردد. گلشاهی و همکاران (۲۰۱۷) با ارایه یک حد پایین جدید و همچنین دو روش هیوریستیک جدید به حل مدل ارائه شده در مقاله یو و اگبلو (۲۰۰۸) پرداختند. آنها همچنین چند روش فراابتکاری نیز برای حل این مساله به کار گرفتند که به نتایج قابل قبولی دست پیدا کردند. ویسیتیپانیچ و هنگمچای (۲۰۱۷) به بررسی مسئله زمانبندی انبار متقاطع با چند درب ورود و خروج پرداختند. آنها برای حل این مسئله روش فراابتکاری ازدحام ذرات تغییر یافته را معرفی نمودند. آنها جواب‌های به دست آمده از این مساله را با روش ازدحام ذرات اصلی مقایسه نمودند و به این نتیجه رسیدند که روش جدید توانایی بالایی در حل این مساله دارد. عدم قطعیت و ماهیت غیر جبری دنیای واقعی برنامه‌ریزی و زمانبندی در انبار متقاطع را یک کار بسیار پیچیده برای تصمیم گیرندگان می‌کند. این تغییرات مداوم که همیشه اتفاق می‌افتد اغلب منجر به افزایش هزینه‌ها و یا کاهش کارایی می شود. بیشتر عدم قطعیت‌ها در انبارهای متقاطع ناشی از زمان ورود ناشناخته کامیون‌ها به انبار متقاطع است. مولوی و همکاران (۲۰۱۸) مساله زمانبندی کامیون در یک انبار متقاطع با در نظر گرفتن موعد تحویل برای کامیون‌های خروجی به عنوان یک محدودیت در نظر گرفتند. هدف مدل آنها به حداقل رساندن کل هزینه ناشی از جریمه تاخیر در تحویل حمل و نقل می‌باشد. توالی کامیون‌های خروجی در این مساله مورد بررسی قرار گرفته است و فرض می‌شود که محموله‌ها بلافاصله پس از تخلیه به دربهای خروجی ارسال می‌شوند و در آن از سیاست اول در اولویت (**FIFO**) برای بارگیری محموله‌ها استفاده می‌شود. یک مدل برنامه‌ریزی عدد صحیح ترکیبی برای مدل پیشنهاد شده است. الگوریتم ترکیبی ژنتیک و جستجوی محلی متغیر برای حل مساله در مقیاس‌های متوسط و بزرگ استفاده شده است و در ادامه با سایر الگوریتم‌های متاهیوریستیک رایج در ادبیات مقایسه شده است. نتایج عددی نشان می‌دهد که موعد تحویل می‌تواند بین یک پنجره زمانی یا یک نقطه خاص بر اساس نیاز مشتریان یا محدودیت های انبار متقاطع محاسبه شود. علاوه بر این، در این مقاله نتیجه گرفته شده است که مرتب سازی محموله‌ها در داخل کامیون‌های ورودی می تواند عملکرد انبار متقاطع را بهبود بخشد.

ریجال و همکاران (۲۰۱۹) یک رویکرد یکپارچه برای حل دو مساله به طور همزمان در انبار متقاطع درحالیکه که درب‌های بارانداز می تواند به ترکیبی هم به عنوان ورودی و هم خروجی عمل کنند، ارائه می کند. آنها یک الگوریتم جستجوی همسایگی بزرگ سازگار برای حل مساله یکپارچه و پیدا کردن راه حل‌های خوب در یک زمان معقول پیشنهاد نمودند. نتایج آزمایش‌های محاسباتی وسیع نشان می‌دهد که هزینه‌های عملیاتی در انبار متقاطع، به طور متوسط به میزان ۱۲٪ در مقایسه با بهترین راه حل‌ها کاهش می‌یابد (و حتی ۲۰-۳۰٪ در مقایسه با نتایج الگوریتم‌های فراابتکاری از ادبیات موضوع). علاوه بر این، با رویکرد یکپارچه، آنها توانستند که میزان درب‌های ورودی برای کار در حالت مختلط و جایی که این درب‌های انعطاف پذیر باید در یک ترمینال انبار متقاطع قرار گیرند را تشخیص دهند. نتایج عددی نشان می دهد که میانگین صرفه‌جویی در هزینه‌ها ۹,۷٪ می‌باشد زمانی که ۶۰٪ از درب‌ها در حالت مختلط استفاده شوند و این صرفه جویی به ۱۲,۳٪ افزایش می‌یابد زمانی که این درب‌ها در مرکز انبار متقاطع قرار گیرند. دالبنتز (۲۰۱۹) مطالعه‌ای با هدف بهبود زمان‌بندی کامیون‌های ورودی و خروجی در یک انبار متقاطع صورت داده است. وی یک مدل برنامه‌ریزی خطی یکپارچه برای مساله زمانبندی کامیون‌ها با هدف مینیمم کردن هزینه کامیون‌ها طراحی ارایه نمود. در این مقاله از یک الگوریتم تکاملی شروع موازی برای حل مساله استفاده شده است. نتایج محاسباتی برتری الگوریتم پیشنهادی را از نظر معیارهای کلیدی الگوریتمی در مقابل پنج الگوریتم دیگر که عموماً در ادبیات انبار متقاطع مورد استفاده قرار می‌گیرند نشان می‌دهد. همچنین وی در این مقاله نشان داد که چگونه می‌توان این الگوریتم توسعه‌یافته را به طور موثر برای تجزیه و تحلیل مسائل مهم مدیریتی در حوزه حمل و نقل استفاده نمود. فونسکا و همکاران (۲۰۱۹) تحقیقی در مورد زمانبندی کامیون‌ها در انبار متقاطع موازی انجام دادند. آنها ابتداً این مساله را به‌عنوان یک مسئله زمانبندی فلوشاپ دو ماشینه با محدودیتهای اولویت و با هدف کمینه کردن زمان تکمیل کارها مدلسازی نمودند، سپس آن را به مساله انبار متقاطع تعمیم دادند. در ادامه برای حل این مساله آنها یک روش ترکیبی براساس تکنیک آرام سازی لاگرانژی و الگوریتم حجمی پیشنهاد نمودند. با استفاده از ضرایب لاگرانژین، هیوریستیک‌های سودمندی با مراحل جستجوی محلی، راه‌حل‌های قابل قبول را ایجاد نمودند. این الگوریتم راه‌حل‌های بسیار خوبی برای مسائل با اندازه‌های کوچک و بزرگ پیدا می‌کند. سیدی و همکاران (۲۰۱۹) مساله زمانبندی کامیون‌ها و اختصاص آنها به درب‌ها به طور همزمان را مورد بررسی قرار دادند. آنها در این تحقیق روش‌های ابتکاری بسیار قوی ارایه نموده‌اند که با کمترین زمان و با بهترین عملکرد نسبت به دیگر رویکردهای ابتکاری ارائه شده تاکنون در مدل‌های مشابه به اهداف مورد نظر دست می‌یابند. در رویکردهای ارایه شده با شناسایی و درنظر گرفتن زمان تاخیری که در طی اجرای فرآیندهای انبارمتقاطع ایجاد می‌شود، سعی برآن بوده که این تاخیرها به حداقل برسند یا حتی در صورت امکان حذف شوند. همچنین، به دلیل اینکه در محیط واقعی معمولاً فضای کوچکی را برای انبارسازی موقت کالاها در نظر می‌گیرند، سیاست روش‌های ابتکاری ارایه شده بر این بوده که انبارش موقت محصولات را نیز به حداقل برساند. در ادبیات موضوع کارهایی بوده‌اند که به مرور ادبیات مرتبط با انبار متقاطع پرداخته‌اند. در جامع ترین کارهای ارایه شده تاکنون میتوان به بویسن و فیلندر (۲۰۱۰)، ون بل و همکاران (۲۰۱۲) و لیدر و آلپان (۲۰۱۶) اشاره نمود.

علاوه بر تحقیقات فوق، اخیرا مطالعات زمان‌بندی با در نظر گرفتن اثرات همزمان زوال پذیری و یادگیری به سبب کاربرد وسیعی که در زندگی روزمره انسان‌ها دارند، مورد توجه محققان قرار گرفته است. در ادامه به مرور ادبیات تعدادی از این تحقیقات که در سال‌های اخیر انجام گرفته‌اند می‌پردازیم.

هوانگ و همکاران (۲۰۱٤) به بررسی مسئله زمان‌بندی ماشین‌های موازی یکسان با فرض وجود زوال‌پذیری کارها و اثرات یادگیری پرداختند و نشان دادند که مسئله مورد مطالعه با اهداف کمینه سازی توابع هزینه شامل مجموع زمان تکمیل و مجموع اختلاف مطلق در زمان‌های تکمیل، و توابع هزینه شامل مجموع زمان انتظار و مجموع اختلاف مطلق در زمان‌های انتظار به صورت چند جمله‌ای قابل حل باقی می‌ماند. رستمی و همکاران (۲۰۱۵) به بررسی مسئله زمان‌بندی ماشین‌های موازی با فرض وجود اثرات یادگیری و زوال‌پذیری کارها تحت محیط فازی و با اهداف کمینه سازی مجموع زمان‌های دیرکرد و زودکرد کارها، و حداکثر زمان تکمیل کارها پرداختند. صالحی‌میر و رضائیان (۲۰۱٦) یک مدل زمان‌بندی جدید را با در نظر گرفتن زمان آماده‌سازی وابسته به توالی گذشته، زمان دسترسی به کارها و اثرات همزمان یادگیری و زوال‌پذیری کارها پیشنهاد نمودند که در آن زمان پردازش واقعی یک کار بر روی هر ماشین تابعی از زمان شروع آن کار، زمان دسترسی به آن کار و موقعیت زمان‌بندی شده آن کار بر روی ماشین مربوطه است. آن‌ها همچنین در این مطالعه از یک الگوریتم فراابتکاری ترکیبی مبتنی بر الگوریتم بهینه‌سازی ازدحام ذرات و ژنتیک برای کمینه‌سازی مجموع بارهای کاری بر روی ماشین‌ها موازی نامرتبط بهره بردند.

طبق بررسی‌های به عمل آمده تا کنون تنها یک مقاله توسط امینی و همکاران (۲۰۱٤) در حالت یک درب ورودی و یک درب خروجی ارایه شده است که به بررسی اثر یادگیری در فرایند تخلیه و بارگیری توسط نیروی انسانی می‌پردازد. آنها یک مدل ریاضی با توجه به مدل‌های موجود ارایه کرده‌اند. آنها سپس چهار الگوریتم اکتشافی به همراه الگوریتم شبیه‌سازی تبرید به منظور غلبه بر پیچیدگی مسائل بزرگ، بکار گرفتند و در نهایت عملکرد الگوریتم‌های پیشنهادی با راه حل‌های بهینه‌ای که با استفاده از روش شمارش کامل به دست آمده، مقایسه نمودند.

۳- مدل‌سازی

در این تحقیق با الهام از مدل‌های موجود و به منظور رفع کاستی‌ها و تطبیق مدل با شرایط واقعی، و با توجه به نیاز صنعت و مسائل دنیای واقعی توسعه مدل ریاضی ارایه شده است. در مدل مورد بررسی در این مقاله برای کامیون‌های ورودی و خروجی بیش از یک درب در نظر گرفته شده است. همچنین در این مدل زمان ورود کامیون‌های مختلف متفاوت و زمان تخلیه و بارگیری کالاهای مختلف از درب‌های ورودی و خروجی نیز متفاوت هستند. در مسایل زمان‌بندی سنتی انبار متقاطع که تاکنون ارایه شده است، زمان پردازش کارها مقادیری ثابت و مستقل از توالی کارها فرض شده‌اند. این فرض در اکثر مواقع مناسب نیست، زیرا با تکرار انجام کار، توانایی و مهارت کارگر افزایش یافته و در نتیجه زمان پردازش کارها کاهش می‌یابد. همانطور که پیش‌تر نیز بیان شد این پدیده با نام اثر یادگیری شناخته می‌شود. همچنین در زمان بندی سیستم‌های تولیدی، هر گونه تأخیر در شروع پردازش کارها می‌تواند منجر به افزایش زمان انجام کارها گردد. این پدیده به عنوان «زوال‌پذیری کارها» در ادبیات شناخته می‌شود. در دنیای واقعی معمولاً کارها از نوع زوال‌پذیر بوده که با تعویق زمان پردازش، زمان واقعی نیز بیشتر می‌شود. این رویه میتواند به دلیل فرسودگی ابزار یا ماشین و یا دلایل دیگری باشد. در ادبیات مورد بررسی بیشتر تحقیقات یک تابع زمان پردازش خطی برای هر کار در نظر گرفته‌اند. مدل پایه‌ای مسأله زمانبندی انبار متقاطع توسط یو و اگبلو (۲۰۰۸) ارایه گردید. در این مدل فرض بر آن است که زمان تخلیه بار از کامیون ورودی و زمان بارگیری در کامیون خروجی یکسان، و برابر با یک واحد از زمان به ازای هر واحد از کالا می‌باشد. همچنین فرض شده است که یک انبار موقت در نزدیکی جایگاه ارسال قرار دارد. اگر یک کالایی به جایگاه ارسال برسد و این کالا مورد نیاز جهت بارگیری در کامیون خروجی موجود در جایگاه نباشد، به انبار موقت منتقل می‌شود، تا زمانی که کامیون خروجی مناسب در جایگاه ارسال قرار گیرد. وقتی یک کامیون به جایگاه دریافت می‌رسد تنها زمانی مجاز به ترک جایگاه می‌باشد که تمامی کالاهای موجود در آن تخلیه شود. به طور، مشابه کامیون‌های خروجی نیز تنها زمانی مجاز به ترک جایگاه ارسال می‌باشند که تمامی کالاهای مورد نیازشان را بارگیری کنند.

۳-۱- مدل پیشنهادی

در این مدل برای نزدیک‌تر شدن به دنیای واقعی اثرات همزمان زوال‌پذیری و یادگیری روی زمان پردازش در نظر گرفته شده است. برای این منظور یک مدل کلی با زوال‌پذیری کارها و اثرات یادگیری پیشنهاد می‌شود که در آن زمان پردازش واقعی یک کار، یک تابع کلی از زمان شروع پردازش کار مورد نظر (اثر زوال پذیری) و موقعیت زمان‌بندی آن در توالی (اثر یادگیری) است. مدل کلی استفاده شده به شرح زیر می‌باشد:

$$P_{j[r]} = P_j r^b + at \qquad (۱)$$

جایی که $\mathbf{a \geq 0}$ و $\mathbf{b \leq 0}$ به ترتیب نرخ زوال‌پذیری و نرخ یادگیری می‌باشند. $\mathbf{t}$ زمان شروع کار $\mathbf{j}$ام و $\mathbf{r}$ هم موقعیت زمان‌بندی کار $\mathbf{j}$ام در توالی و P_j هم زمان پردازش اولیه می‌باشد. مفروضات زیر برای مدل در نظر گرفته شده است.

۳-۲- مفروضات و خصوصیات مدل

- زمان ورود کامیون های ورودی متغیر می‌باشد.
- تمامی کالاهای ورودی باید از سیستم خارج شوند و ذخیره‌سازی بلند مدت مجار نمی‌باشد.
- تعداد کل کالاهای وارد شده از یک نوع باید برابر با تعداد کل کالاهای خراج شده از همان نوع باشد.
- زمان تعویض کامیون‌ها برای تمامی کامیون‌های ورودی و خروجی برابر می‌باشد.
- ظرفیت انبار موقت نامحدود می‌باشد.
- چندین درب ورودی و خروجی وجود دارند.
- تأثیر یادگیری بر زمان پردازش در نظر گرفته شده است.
- تأثیر زوال‌پذیری کارها بر زمان پردازش نیز درنظرگرفته‌شده است.

- برای کامیون‌های ورودی و خروجی زمان تحویل در نظر گرفته شده است.
- برای کامیون‌ها ظرفیت محدود در نظر گرفته شده است.

۳-۳- متغیرها و پارامترهای مدل

R: تعداد کامیون‌های ورودی $i=1 \dots R$

S: تعداد درب های خروجی $j=1 \dots S$

N: تعداد کالاها $k=1 \dots N$

T: زمان تکمیل

r_{ik}: تعداد کالای نوع k که در کامیون ورودی i وجود دارد

s_{jk}: تعداد کالای نوع k که در کامیون خروجی j وجود دارد

D: مدت زمان تعویض کامیون ها

M: یک عدد بسیار بزرگ

c_i: زمان ورود کامیون ورودی iام به درب دریافت

F_i: زمان خروج کامیون ورودی iام از درب دریافت

d_j: زمان ورود کامیون خروجی jام به درب ارسال

L_j: زمان خروج کامیون خروجی jام از درب ارسال

x_{ijk}: تعداد کالای نوع k که از کامیون ورودی i به کامیون خروجی j انتقال می‌یابد

v_{ij}: اگر کالای از کامیون ورودی iام به کامیون خروجی j انتقال یابد=۱ در غیر اینصورت=۰

p_{ij}: اگر کامیون ورودی iام مقدم بر کامیون ورودی j ام باشد=۱ در غیر اینصورت=۰

q_{ij}: اگر کامیون خروجی iام مقدم بر کامیون خروجی j ام باشد=۱ در غیر اینصورت=۰

t_k^I: مدت زمان تخلیه کالای نوع k از کامیون‌های ورودی

t_k^O: مدت زمان تخلیه کالای نوع k از کامیون‌های خروجی

b: نرخ یادگیری

a: نرخ زوال‌پذیری

A^i: زمان ورود کامیون ورودی iام

p_{ij}: اگر کامیون ورودی iام مقدم بر کامیون ورودی j ام باشد و هر دو به یک درب تخصیص داده شده باشند=۱ در غیر اینصورت=۰

q_{ij}: اگر کامیون خروجی iام مقدم بر کامیون خروجی j ام باشد و هر دو به یک درب تخصیص داده شده باشند =۱ در غیر اینصورت=۰

z_{im}: اگر کامیون iام به درب ورودی mام اختصاص یابد=۱ در غیر اینصورت=۰

y_{jn}: اگر کامیون jام به درب خروجی nام اختصاص یابد=۱ در غیر اینصورت=۰

b_{ijmn} اگر کامیون iام به درب ورودی mام اختصاص یابد و کامیون jام به درب ورودی nام اختصاص یابد و بین این دو نقل و انتقال صورت گیرد=۱ در غیر اینصورت=۰

T_{mn}: مدت زمان ارسال کالا از درب ورودی m ام به درب خروجی nام

۴-۳- مدل ریاضی

$$\text{Min Z} = T \qquad for\ all\ j \qquad (۲)$$

$$s.t.$$

$$T \geq L_j \qquad for\ all\ j \qquad (۳)$$

$$\sum_{j=1}^{S} X_{ijk} = r_{ik} \qquad for\ all\ \ i,k \qquad (۴)$$

$$\sum_{i=1}^{R} X_{ijk} = S_{jk} \qquad for\ all\ \ j,k \qquad (۵)$$

$$X_{ijk} \leq M v_{ij} \qquad for\ all\ \ i,j,k \qquad (۶)$$

$$\sum_{m=1}^{I} z_{im} = 1 \qquad for\ all\ i \qquad (۷)$$

$$\sum_{n=1}^{O} y_{jn} = 1 \qquad for\ all\ j \qquad (۸)$$

$$\sum_{m=1}^{I}\sum_{n=1}^{O} b_{ijmn} = v_{ij} \qquad for\ all\quad i,j \qquad (۹)$$

$$b_{ijmn} \le z_{im} \qquad for\ all\quad i,j,m,n \qquad (۱۰)$$

$$b_{ijmn} \le y_{jn} \qquad for\ all\quad i,j,m,n \qquad (۱۱)$$

$$z_{im} + z_{jm} - 1 \le P_{ij} + P_{ji} \qquad for\ all\quad i,j,m\ and\ i \neq j \qquad (۱۲)$$

$$P_{ij} + P_{ji} \le 1 \qquad for\ all\quad i,j\ and\ i \neq j \qquad (۱۳)$$

$$y_{in} + y_{jn} - 1 \le q_{ij} + q_{ji} \qquad for\ all\quad i,j,n\ and\ i \neq j \qquad (۱۴)$$

$$q_{ij} + q_{ji} \le 1 \qquad for\ all\quad i,j\ and\ i \neq j \qquad (۱۵)$$

$$F_i \ge c_i + \sum_k t_k^I r_{ik}\left(1 + \sum_j \{1 - P_{ij}\}\right)^b + ac_i \qquad for\ all\ i \qquad (۱۶)$$

$$C_j \ge A_i - M(1 - P_{ij}) \qquad for\ all\ i,j \quad and\ i \neq j \qquad (۱۷)$$

$$C_i \ge A_i \qquad for\ all\ i \qquad (۱۸)$$

$$C_j \ge F_i + D - M(1 - P_{ij}) \qquad for\ all\ i,j \quad and\ i \neq j \qquad (۱۹)$$

$$C_i \ge F_j + D - MP_{ij} \qquad for\ all\ i,j \quad and\ i \neq j \qquad (۲۰)$$

$$P_{ij} = 0 \qquad for\ all\ i \qquad (۲۱)$$

$$L_j \ge d_j + \sum_k t_k^O s_{jk}\left(1 + \sum_i \{1 - q_{ij}\}\right)^b + ad_j \qquad for\ all\ j \qquad (۲۲)$$

$$d_j \ge L_i + D - M(1 - q_{ij}) \qquad for\ all\ i,j \quad and\ i \neq j \qquad (۲۳)$$

$$d_i \ge L_j + D - Mq_{ij} \qquad for\ all\ i,j \quad and\ i \neq j \qquad (۲۴)$$

$$q_{jj} = 0 \qquad for\ all\ j \qquad (۲۵)$$

$$L_j \geq c_i + \sum_{m=1}^{I}\sum_{n=1}^{O} T_{mn} b_{ijmn} + \sum_k X_{ijk}\left(t_k^O \left(1 + \sum_i \{1 - q_{ij}\}\right)^b + ad_i + t_k^I \left(1 + \sum_j \{1 - P_{ij}\}\right)^b + ac_j \right) - M(1 - v_{ij}) \qquad for\ all\ i,j \qquad (۲۶)$$

$$F_i \leq DDate_i; \qquad for\ all\ i \qquad (۲۷)$$

$$L_j \leq RDate_j \qquad for\ all\ j \qquad (۲۸)$$

$$\sum_k r_{ik} \leq cap_i \qquad for\ all\ i \qquad (۲۹)$$

$$\sum_k s_{jk} \leq cap_j \qquad for\ all\ j \qquad (۳۰)$$

$$All\ variables \geq 0 \qquad (۳۱)$$

در این مدل حالتی بررسی می‌شود که چند درب برای کامیون‌های ورودی و خروجی با در نظر گرفتن اثر یادگیری و اثر زوال‌پذیری در نظر گرفته شده است. زمان ورود کامیون‌های مختلف و همچنین زمان تخلیه و بارگیری کالاهای مختلف از درب‌های ورودی و خروجی نیز متفاوت در نظر گرفته شده‌اند. تابع هدف حداقل‌سازی زمان انجام کل عملیات در سیستم می‌باشد. به عبارت دیگر آن از زمانی که اولین کامیون ورودی، اولین کالای خود را تخلیه می‌کند شروع می‌شود و تا زمانی که اخرین کالا در آخرین کامیون خروجی بارگیری شود پایان می‌پذیرد. محدودیت اول که در رابطه ۳ نمایش داده شده، نشان دهنده این نکته است که زمان تکمیل عملیات بزرگتر مساوی زمانی است که آخرین ماشین ارسال سکوی ارسال را ترک می‌کند. محدودیت‌های ٤ و ٥ نشان می‌دهد که تعداد کل اقلام دریافت شده توسط بارکش‌های دریافت با مقدار کل اقلام ارسال شده توسط بارکش‌های ارسال برابر است. محدودیت ٦ ارتباط بین متغیرهای X_{ijk} (تعداد کالای نوع k که از کامیون ورودی i به کامیون خروجی j انتقال می‌یابد)و v_{ij} (آیا کالای از کامیون ورودی iام به کامیون خروجی j انتقال می‌یابد) را نشان می‌دهد. محدودیت‌های ۷ و ۸ بیانگر این مسئله هستند که هر کامیون ورودی به یک درب ورودی و هر کامیون خروجی به یک درب خروجی تخصیص می‌یابد. محدودیت‌های ۹ و ۱۰ و ۱۱ رابطه‌ی درستی بین متغیرهای b_{ijmn} ، v_{ij} ، z_{im} و y_{in} به وجود می‌آورد. محدودیت‌های ۱۲ و ۱۳ بین متغیرهای z_{im} و P_{ij} ارتباط برقرار می‌کند. محدودیت‌های ١٤ و ١٥ بین متغیرهای y_{in} و q_{ij} ارتباط برقرار می‌کند. محدودیت‌های ١٦، ١٧، ١٨، ١٩ و ٢٠ زمان ورود و خروج ماشین‌های دریافت را بر اساس ترتیب شان در توالی ماشین‌ها، زمان ورود آنها و نرخ یادگیری و زوال‌پذیری مشخص می‌کند. محدودیت ٢١ تضمین می‌کند که هیچ کامیون ورودی از خودش در ترتیب توالی‌ها پیشی نگیرد. محدودیت‌های ٢٢، ٢٣ و ٢٤ زمان ورود و خروج ماشین‌های ارسال را بر اساس ترتیب شان در توالی ماشین‌ها و نرخ یادگیری و زوال‌پذیری مشخص می‌کند. محدودیت ٢٥ تضمین می‌کند که هیچ کامیون خروجی از خودش در ترتیب توالی‌ها پیشی نگیرد. محدودیت ٢٦ بین زمان خروج برای هر کامیون خروجی و زمان

ورود برای هر ماشین ورودی برای هر نوع کالا، ارتباط برقرار می‌کند. همچنین در محدودیت ۲۶ با توجه به اینکه در این مدل زمان ارسال کالا از هر درب ورودی m ام به هر درب خروجی nام متفاوت است، این حالت را نیز در نظر می‌گیرد. محدودیت‌های ۲۷ و ۲۸ موعد تحویل کامیون ورودی iام و خروجی jام را مشخص می‌کند. همچنین محدودیت‌های ۲۹ و ۳۰ ظرفیت کامیون‌های ورودی iام و خروجی jام را مشخص می‌کند.

٤- رویکردهای حل

پیچیدگی استفاده از بهینه‌سازی ریاضی در بسیاری از مسائل بهینه‌سازی در ابعاد بزرگ و عدم جوابگویی در حل مسائل سخت با تعداد متغیر زیاد و توابع هدف غیرخطی، به توسعه راه‌حل‌های جایگزین منجر شده است. از این رو گاهی در عمل، معمولاً جواب‌هایی با کیفیت خوب که به‌وسیله‌ی تکنیک‌های بهینه‌سازی تقریبی به دست می‌آیند برای ما رضایت بخش خواهد بود. از اینرو در این مقاله از چهار الگوریتم فراابتکاری شامل الگوریتم‌های ژنتیک (**GA**)، رقابت استعماری (**ICA**)، کشتل (**KA**) و بهینه سازی مهندسی اجتماعی (**SEO**) استفاده شده است. دلیل انتخاب این چهار الگوریتم استفاده از الگوریتم‌های قدیمی استفاده شده در مسائل انبار متقاطع و همچنین استفاده از الگوریتم‌های جدید برای آزمایش کارایی آنها در مسئله مورد نظر می‌باشد.

٤-١- نمایش جواب

روش کدگذاری مسئله مورد نظر در شکل (۱) نشان داده شده است. ابعاد کروموزوم با مجموع تعداد کامیون‌های دریافت، ارسال و درب‌های تخصیصی کامیون‌ها به آنها می‌باشد. برای مثال یک انبار متقاطع با چهار کامیون ورودی، چهار کامیون خروجی، دو درب ورودی و دو درب خروجی را در نظر بگیرید. برای این مثال، ابعاد جواب برابر با ۱٦ می‌باشد زیرا هشت کامیون ورودی و خروجی و دو درب وجود دارد. مقدار هر بعد در ابتدا با یک عدد تصادفی یکنواخت بین [۰، ۱] تولید می‌شود. همانطور که در شکل (۱) نشان داده شده است ساختار یک جواب متشکل از چهار قسمت درب ورودی، کامیون ورودی، درب خروجی و کامیون خروجی است. سمت چپ قسمت پایین جواب تعیین کننده‌ی توالی کامیون‌های دریافت، و سمت راست آن تعیین کننده‌ی توالی کامیون‌های ارسال می‌باشد. قسمت پایین جواب نشان دهنده درب‌های تخصیصی به کامیون‌ها است. این طرح کدگذاری در تمام الگوریتم‌های فراابتکاری استفاده شده در این مطالعه بکار رفته است. برای رمزگشایی این اعداد تصادفی در ابعاد یک جواب، در این مقاله از یک قاعده فهرست‌بندی مرتب‌سازی برای تولید دنباله‌ای از کامیون‌ها و اختصاص کامیون‌ها به درب‌ها استفاده می‌شود. همانطور که در شکل (۱) نشان داده شده است، دنباله‌ای از کامیون‌ها بر اساس ترتیب مقادیر صعودی در هر ژن تعیین می‌شود، تخصیص درب‌ها با استفاده از یک لیست مرتب‌سازی با تکرار تعداد درب‌های بارانداز مشخص می‌شود. در این مثال، کامیون‌های ورودی با ترتیب کامیون شماره ٤، کامیون شماره ۱، کامیون شماره ۲ و کامیون شماره ۳ در توالی وارد می‌شوند. علاوه بر این، کامیون شماره ٤، کامیون شماره ۱، کامیون شماره ۲ و شماره ۳ کامیون به ترتیب به درب‌های شماره ۲، درب شماره ۱، درب شماره ۲ و درب شماره ۱ اختصاص می‌یابند. مزیت این روش این است که همیشه یک راه حل قابل قبول را ایجاد می‌کند.

۰,۸۵	۰,٤۷	۰,٥٤	۰,۱٦	۰,۷٦	۰,۹۸	۰,٤	۰,٤۳
۰,۹٦	۰,۱۸	۰,۷	۰,۳۳	۰,٤٤	۰,۲٤	۰,۰٦	۰,۹٤

درب‌های خروجی				درب‌های ورودی			
کامیون‌های خروجی				کامیون‌های ورودی			
تخصیص درب‌ها				تخصیص درب‌ها			
۲	۲	۱	۱	۲	۱	۲	۲
٤	۱	۳	۲	۳	۱	۲	٤
توالی کامیون‌های خروجی				توالی کامیون‌های ورودی			

شکل ۱. مثالی از طرح کدگذاری برای حالت چنددربی

٤-١-١- الگوریتم ژنتیک

نخستین بار هلند (۱۹۹۲) ایده‌ی استفاده از الگوریتم ژنتیک را برای مسائل بهینه‌سازی مطرح نمود. سپس این الگوریتم به طور گسترده در طیف وسیعی از مسائل بهینه سازی مورد استفاده قرار

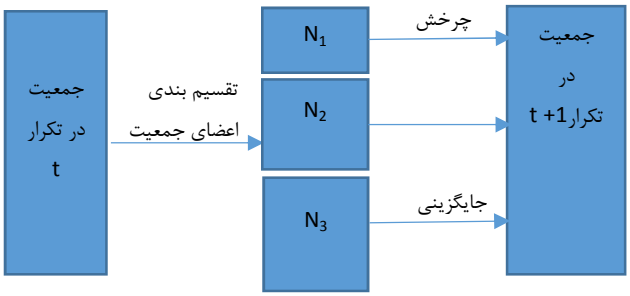
گرفت. **GA** یک الگوریتم جمعیت محور است. یعنی کار خود را با یک جمعیت اولیه از راه‌حل‌ها آغاز می‌کند. سپس یک نسل جدید ایجاد می‌کند که جایگزین جمعیت فعلی می‌شود. به هر یک از اعضای جمعیت کروموزوم گفته می‌شود. اغلب جمعیت اولیه بصورت تصادفی تولید می‌شود. سپس میزان برازندگی هر یک از اعضای جمعیت (کروموزوم) تعیین می‌شود. در مسائل بهینه‌سازی، معمولاً میزان برازندگی مطابق با تابع هدف تعیین می‌شود. سپس جمعیت فعلی توسط عملگرهای الگوریتم ژنتیک تکامل می‌یابد. الگوریتم ژنتیک دارای سه عملگر به نام‌های انتخاب، تقاطع و جهش است.

```
.1  Initialise generation 0:
2.  k := 0;
3.  Pk := a population of n randomly-generated
    individuals
    Evaluate Pk:
    Compute fitness(i) for each i ∈ Pk
    Do
5.      Create generation k + 1:
          5.1. Copy: Select (1 − χ) × n members of Pk
          and insert into Pk+1;
          5.2. Crossover: Select χ × n members of
          Pk; pair them up; produce offspring; insert
          the offspring into Pk+1;
          5.3. Mutate: Select μ × n members of
        Pk+1; invert a randomly-selected bit in
6.      each;
7.      Evaluate Pk+1: Compute fitness(i) for each i
        ∈ Pk;
        Increment: k := k + 1; }
8.  while fitness of fittest individual in Pk is not high
9.  enough;
    return the fittest individual from Pk
```

٤-١-٢- الگوریتم کشتل

الگوریتم کشتل توسط حاجی آقائی-کشتلی و امین نیری (۲۰۱۳) جهت بهینه‌سازی مسائل پیوسته ارایه گردید. ایده‌ی اصلی این الگوریتم از رفتار غذایی یک نوع پرنده‌ی خاص به نام کشتل الهام گرفته شده است. **KA** یک الگوریتم جمعیت‌محور است.

در این الگوریتم به هریک از اعضای جمعیت کشتل گفته می‌شود. کشتل‌های خوش شانس غذای بهتری را در دریاچه می‌یابند. سپس کشتل‌هایی که در همسایگی آن‌ها هستند به سمتشان جذب می‌شوند و همه با هم بدور منبع غذایی چرخش می‌کنند. در حین چرخش اگر کشتلی منبع غذایی بهتری پیدا کند به عنوان کشتل خوش‌شانس شناخته می‌شود. همچنین تعدادی از کشتل‌ها به منظور یافتن غذا به سمت نقاط بکر دریاچه حرکت می‌کنند. آنها در حین این حرکت، موقعیت دو کشتل دیگر را در نظر می‌گیرند. تعدادی کشتل نیز در دریاچه وجود دارند که هیچ غذایی به دست نمی‌آورند، آنها دریاچه را ترک می‌کنند و کشتل‌های جدیدی جایگزینشان می‌گردند. شکل (۳) نمای شماتیک از گام‌های اصلی الگوریتم کشتل را بطور خلاصه نشان می‌دهد. همچنین شبه کد الگوریتم الگوریتم کشتل به طور خلاصه در شکل (٤) نشان داده شده است.



شکل ۳. فرآیند الگوریتم KA

1. Land the Keshtels
2. Find the Lucky Keshtels (LK).
3. For each LK:
   3.1. Swirl the Nearest Keshtel (NK) around the LK.
   3.2. If NK finds better food than LK, replace NK with LK, find new NK, go to step 3.1.
   3.3. If the food still exists, attract the NK, go to step, 3.1.if not, go to step 4.
4. Let the LKs remain in the lake
5. Startle the Keshtels which have found less food and land new ones.
6. Hustle the remained Keshtels in the lake.

شکل ٤. شبه کد الگوریتم کشتل

٤-١-٣- الگوریتم رقابت استعماری

الگوریتم رقابت استعماری برای بهینه‌سازی نه از یک پدیده طبیعی، بلکه از یک پدیده اجتماعی انسانی الهام گرفته است. الگوریتم ICA پدیده تاریخی استعمار را در راستای یک تکامل اجتماعی–سیاسی جوامع انسانی تحلیل نموده و با مدلسازی ریاضی این فرایند، یک الگوریتم قدرتمند برای بهینه سازی ارائه نموده است. همانند همه الگوریتم های قرار گرفته در دسته الگوریتم‌های بهینه سازی تکاملی، الگوریتم رقابت استعماری نیز مجموعه اولیه ای از جوابهای احتمالی را تشکیل می‌دهد. این جوابهای اولیه در الگوریتم رقابت استعماری با عنوان "کشور" شناخته می‌شوند. در این الگوریتم، همه کشورها در چندین امپراتوری گروه‌بندی می‌شوند. الگوریتم رقابت استعماری با روند خاصی که در ادامه می‌آید، این جوابهای اولیه (کشورها) را به تدریج بهبود داده و در نهایت جواب مناسب مسئله بهینه سازی (کشور مطلوب) را در اختیار می‌گذارد. مراحل الگوریتم ICA را می‌توان به صورت شبه کد شکل ۵ خلاصه کرد.

1. Initialization of the algorithm. Generate some random solution in the search space and create initial empires.
2. Assimilation: Colonies move towards imperialist states in different in directions.
3. Revolution: Random changes occur in the characteristics of some countries
4. Position exchange between a colony and Imperialist. A colony with a better position than the imperialist, has the chance to take the control of empire by replacing the existing imperialist.
5. Imperialistic competition: All imperialists compete to take possession of colonies of each other.
6. Eliminate the powerless empires. Weak empires lose their power gradually and they will finally be eliminated.
7. If the stop condition is satisfied, stop, if not go to 2.
8. **End**

شکل ۵. شبه کد الگوریتم ICA

٤-١-٤- بهینه‌سازی مهندسی اجتماعی

الگوریتم مهندسی اجتماعی توسط فتح‌الهی‌فرد و همکاران (۲۰۱۸) معرفی شده که الهام گرفته از قواعد مهندسی اجتماعی به عنوان یک پدیده نوظهور در دنیای واقعی امروز شبیه‌سازی شده است). این الگوریتم تنها با دو جواب ابتدایی تصادفی به نام مهاجم و مدافع شروع به کار می‌کند و فازهای جست و جو با استفاده از قواعد مهندسی اجتماعی می‌باشد که در آن مهاجم با

استفاده از تکنیک‌های خاصی به دنبال رسیدن به اهداف مطلوب خود است. شکل (٦) فلوچارت این الگوریتم را نشان می‌دهد. در این روش هر جواب بیان کننده هر فرد و خصوصیات آن فرد شامل توانایی‌های او در زمینه ریاضیات، ورزشی، تجارت و غیره بیان کننده متغیرهای مساله می‌باشند. همان طور که پیش‌تر گفته شد، این الگوریتم با دو جواب تصادفی شروع به کار می‌کند که جواب بهتر را مهاجم و دیگری را مدافع نام‌گذاری می‌کنیم. برای شبیه‌سازی یادگیری و بازآموزی از طرف مهاجم به مدافع، یک سری آزمایشات تصادفی برای هر مشخصه تعریف شده که در ان مهاجم یک مشخصه خود را در مدافع آزمایش می‌کند و مقدار یادگیری محاسبه شده و مدافع جدید به این صورت که بیشترین نرخ بازآموزی را داشت در صورت وجود جایگزین مدافع فعلی می‌شود. در ادامه انجام حملات از مدافع با توجه به تکنیک‌های که در اختیار اوست انجام میگیرد در طی این کنش‌ها مدافع به نقاطی که مدنظر مهاجم است منتقل شده تا پاسخ به حملات انجام گیرد و مدافع ارزیابی شده و این روند تا پایان حملات تکرار شده و اگر مدافع ارزش بیشتری نسبت به مهاجم داشته باشد جای این دو از یکدیگر عوض شده و در پایان مدافع جدید برای راه اندازی مجدد الگوریتم استفاده شده است.

```
T1=clock;
Initialize attacker and defender
It=1;
while solving_time < Max_time
Do training and retraining;
Num_attack=1;
while Num_attack < Max_attack
Spot an attack;
Check the boundary;
Respond to attack;
if the OF defender is lower than attacker
Exchange the defender and attacker position;
End if
Num_attack= Num_attack+1;
End while
Create a new solution as defender;
It=It+1;
T2=clock;
Solving_time=T2- T1;
End while
Return attacker.
```

شکل ٦. شبه‌کد الگوریتم پیشنهادی SEO

۵- نتایج محاسباتی

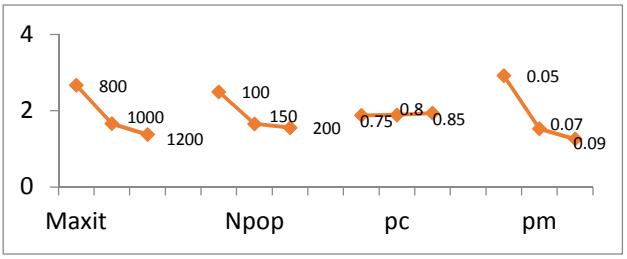
در این قسمت ابتدا به طراحی مسأله‌هایی جهت بررسی عملکرد الگوریتم‌های ارایه شده می‌پردازیم. سپس پارامترهای هر الگوریتم تعیین و تنظیم می‌گردد. تمامی مسائل توسط الگوریتم‌ها اجرا، و نتایج بدست آمده توسط آنها با استفاده از حل دقیق مورد بررسی و تجزیه و تحلیل قرار می‌گیرد. سپس معیارهای خاصی تعریف، و کارایی الگوریتم‌ها نسبت به یکدیگر مطابق با آن معیارها مورد قیاس قرار می‌گیرد. برای مقایسه عملکرد الگوریتم‌های فراابتکاری در حالت چنددربی از ۳۵ مسأله در ابعاد کوچک، متوسط و بزرگ که توسط ون بل و دیگران (۲۰۱۳) طراحی گردید، استفاده شده است. مشخصات مربوط به آن‌ها در مقاله ون بل و دیگران (۲۰۱۳) قابل دسترسی می باشد.

۵-۱- تنظیم پارامتر

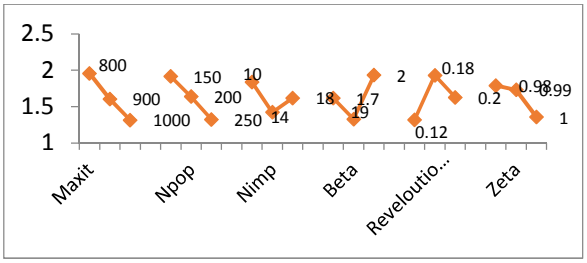
هر کدام از روش‌های فراابتکاری به پارامترهایی برای انجام فرایند مخصوص خود نیاز دارند تا نتایج به دست آمده از روش‌های فراابتکاری مورد نظر را مورد بررسی قرار دهد. تمامی الگوریتم‌های فراابتکاری معرفی شده در قسمت قبل دارای پارامترهایی می‌باشند که برای عملکرد بهتر الگوریتم و ایجاد جواب‌هایی با کیفیت بالاتر نیاز به تنظیم شدن دارند. در نتیجه ابتدا سطوح مختلف برای هریک از پارامترها در نظر گرفته می‌شود. و سپس با استفاده از روش‌هایی سطح مناسب برگزیده می‌شود. طراحی آزمایش روشی برای ایجاد بالاترین بازدهی با کمترین زمان و هزینه می‌باشد. تنظیم پارامترهای الگوریتم با هدف کاهش احتمال تولید راه‌حل‌های بد و کمک به تولید راه‌حل‌های مناسب انجام می‌شود. پس از تعیین تعداد پارامترها و سطوح آنها، تعداد آزمایش‌های مورد نیاز را از طریق جدول پیشنهادی تاگوچی به نام آرایه‌های ارتوگونال مشخص می‌شود.از آنجا که مقیاس مسائل آزمایشی متفاوت است، ما نمی‌توانیم نتیجه به دست آمده را به طور مستقیم اعمال کنیم. برای از بین بردن این مشکل، با استفاده از **RPD** بر روی مقادیری که از طریق آزمایشات برای هر مسأله به دست می‌آید، بی‌مقیاس‌سازی صورت می‌گیرد. مقدار **RPD** با استفاده از رابطه‌ی ٤٩.

$$RPD = \frac{Sol_{ij}^{k} - Bestsol^{k}}{Bestsol^{k}} \qquad (٤٩)$$

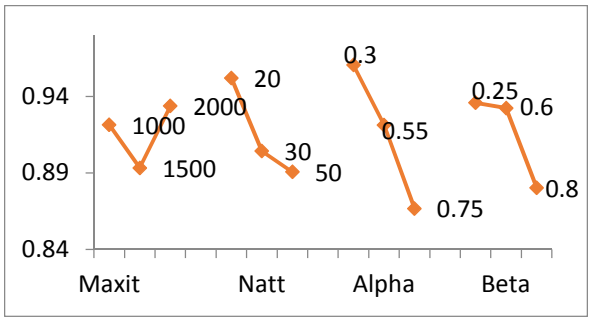
در رابطه‌ی بالا Sol_{ij}^{k}، مقدار به دست آمده برای مسأله‌ی ***K***ام در تکرار ***j***ام از آزمایش ***i*** است. ***Bestsol***^{k} بهترین مقدار بدست آمده برای مسأله ی ***k***ام است. شکل‌های ۷، ۸، ۹، ۱۰ و ۱۱ مقادیر **RPD** به‌دست‌آمده برای پارامترهای مختلف را نشان می‌دهند.



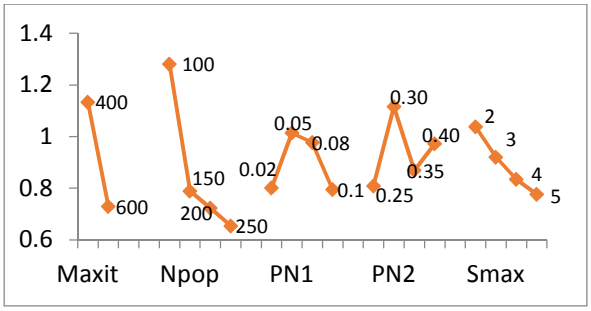
شکل ۷. نمودار RPD پارامترهای الگوریتم GA



شکل ۸. نمودار RPD برای پارامترهای الگوریتم ICA



شکل ۹. نمودار RPD برای پارامترهای الگوریتم SEO



شکل ۱۰. نمودار RPD برای پارامترهای الگوریتم KA

میانگین **RPD** یک پاسخ متوسط برای ترکیب‌های مختلف از سطوح فاکتورهای کنترل در یک طرح تاگوچی می‌باشد. با توجه به هدف این تحقیق که مینیمم سازی است، در صدد تعیین سطح عاملی هستیم که میانگین را به حداقل برساند. به عنوان مثال در شکل (۷) که مربوط به میانگین **RPD** الگوریتم **GA** است، چهار پارامتر ۳ سطحی وجود دارد. می‌خواهیم بدانیم که این چهار پارامتر بر تابع هدف چه تاثیری دارند. نقاط نشان‌داده شده در این نمودار تخمینی از تابع هدف در هر سطح از پارامتر را ارایه می‌دهد. از آنجا که در این مقاله به دنبال به حداقل رساندن زمان تکمیل کار می‌باشیم، بنابراین، در هر پارامتر، سطحی را تعیین می‌کنیم که کمترین میانگین را دارا باشد. نتایج نهایی از تنظیم پارامترهای الگوریتم‌های فراابتکاری در جدول (۱) نمایش داده شده است.

جدول ۱. بهترین سطح برای پارمترهای الگوریتم‌های فراابتکاری

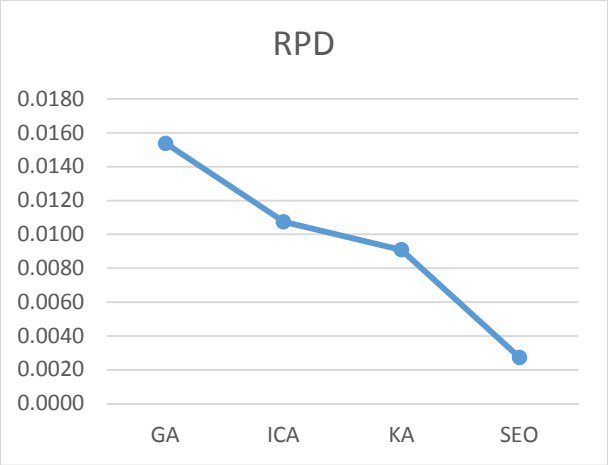
الگوریتم	پارامتر	بهترین سطح
	Max it	۱۲۰۰
GA	*N pop*	۲۰۰
	Pc	۰,۷۵
	Pm	۰,۰۹
	Max it	۷۰۰
	N pop	۲۵۰
KA	*PN1*	۰,۱

	PN2	۰,۲۵
	Smax	٦
	Max it	۱۰۰۰
	N pop	۲۵۰
ICA	*N imp*	١٤
	Beta	۱,۹
	Revolution rate(R)	۰,۱۲
	Zeta	۱
	Max it	۲۰۰۰
SEO	*تعداد حملات (Natt)*	۵۰
	نرخ یادگیری (α)	۰,۷۵
	نرخ پاسخ به حمله (β)	۰,۸

۵-۲- نتایج الگوریتمهای فراابتکاری در مدل پیشنهادی

به منظور بررسی عملکرد الگوریتم‌های پیشنهادی، از مسائل نمونه‌ی ون بل و دیگران (۲۰۱۳) استفاده گردید. بعد از محاسبه‌ی مقدار پارامترها هر الگوریتم ۳۰ بار اجرا گردید. بهترین نتیجه و میانگین نتایج با توجه به مقادیر مختلف نرخ یادگیری و نرخ زوال‌پذیری به دست آمد. در این نتایج به اعتبارسنجی مدل ارائه شده و تحلیل حساسیت مدل نسبت به تغییر در مقادیر پارامترهای ورودی می‌پردازیم. با تغییر مقادیر نرخ یادگیری و زوال‌پذیری به تحلیل حساسیت مسئله نسبت به فعال بودن یا نبودن اثرات زوال‌پذیری و یادگیری می‌پردازیم. به همین منظور مقدار نرخ یادگیری را از $\boldsymbol{b = -0.1}$ تا $\boldsymbol{b = -0.9}$ تغییر می‌دهیم. همچنین نرخ زوال‌پذیری را نیز از $\boldsymbol{\alpha = 0.1}$ تا $\boldsymbol{\alpha = 0.9}$ تغییر می‌دهیم. با افزایش نرخ یادگیری اثر کاهشی آن روی زمان پردازش کمتر بیشتر خواهد شد در نتیجه انتظار داریم که مقدار تابع هدف قطعاً بدتر نخواهد شد و به احتمال زیادی بهبود خواهد داشت. مسئله مورد نظر را مجدد با اعمال این تغییرات حل می‌کنیم. نتایج به دست آمده حاکی از صادق بودن ادعای ما است. در ارتباط با تحلیل حساسیت مسئله نسبت به فعال بودن یا نبودن اثر زوال پذیری نیز اگر نرخ زوال‌پذیری کاهش یافته باشد لذا، اثر افزایشی زوال‌پذیری روی زمان پردازش کاهش خواهد یافت در نتیجه انتظار داریم که تابع هدف قطعاً بدتر نشود و با احتمال زیادی مقداری کاهش خواهد داشت. مسئله مورد نظر را با اعمال این تغییرات حل می‌کنیم. نتایج به دست آمده در جداول پیوست حاکی از صادق بودن ادعای ما است. در جدول (۲) بهترین نتیجه و میانگین نتایج با توجه به مقادیر مختلف نرخ یادگیری و نرخ زوال‌پذیری و جواب دست آمده توسط هر راه‌حل فراابتکاری نشان داده شده است. معیارهای مختلفی برای محاسبه پراکندگی در آمار وجود دارد. اگرچه دامنه و انحراف استاندارد تاکنون بیشتر مورد استفاده قرار گرفته است اما روش‌های دیگری نیز برای تعیین کمیت پراکندگی وجود دارد. به منظور مقایسه و بررسی کارایی الگوریتم‌های فراابتکاری، از معیار (**RPD**) با استفاده از فرمول (٤٩) و زمان پردازش استفاده شده است. میانگین مقدار **RPD** برای جواب‌های به دست آمده توسط الگوریتم‌ها در تمامی مسائل کوچک، متوسط، بزرگ و بسیار بزرگ در سی بار اجرای الگوریتم در شکل‌های (۱۱)، (۱۲) و (۱۳) نشان داده شده است. همانطور که در جداول (۲)، (۳) و (٤) نشان داده شده است، برای نمونه‌های کوچک، الگوریتم‌های پیشنهادی ما با توجه به راه‌حل‌های بهینه به دست آمده توسط نرم افزار **Lingo** در یک زمان محاسباتی بسیار کم به راه‌حل بهینه دست پیدا می‌کنند. همانطور که در شکل‌های (۱۱)، (۱۲) و (۱۳) نیز دیده می‌شود میزان شکاف به دست آمده از جواب بهینه برای تمامی روش‌های فراابتکاری صفر می‌باشد. برای نمونه‌های متوسط (مسائل ۹ الی ١٦) راه‌حل دقیق تنها قادر به پاسخ‌گویی تا

مساله ۱۳ است و پس از آن به دلیل بزرگ شدن ابعاد مسائل در زمان منطقی قادر به حل آنها نمی‌باشد. از مسئله ۹ تا ۱۳ میزان شکاف به‌دست آمده از جواب بهینه برای اغلب روش‌های فراابتکاری بسیار اندک و نزدیک به صفر می‌باشد. از مساله ۱۳ به بعد الگوریتم‌های فراابتکاری با یکدیگر و با استفاده از معیاره **RPD** مقایسه می‌شوند تا عملکرد آنها مشخص شود. همانطور که در جداول (۲)، (۳) و (٤) مشاهده می‌شود،۲۰ مساله متوسط و بزرگ حل و مقایسه می‌شوند. همانطور که در این جداول دیده می‌شود از میان پنج روش فراابتکاری مطابق معیارهای مورد بررسی، الگوریتم **SEO** بهترین راه‌حل‌ها را برای اکثر مسائل از بین سایر روشها به دست می‌آورد. همچنین روش **KA** نیز دارای کارایی بسیار بالایی می‌باشد. همچنین از این شکل‌ها می‌توان دریافت که روش‌های **GA** و **ICA** مقادیر **RPD** بسیار بزرگتری را در مقایسه با سایر روش‌ها به دست می‌آورند. همانطور که در شکل‌های (۱۱)، (۱۲) و (۱۳) مشاهده می‌شود، واضح است که روش‌های فراابتکاری **SEO**، **KA** در مقایسه با سایر روش‌ها رفتار متفاوتی دارند. بعضی از الگوریتم‌ها مانند **GA** دارای میزان شکاف نزدیک به ۰,۰۱٦ ، در حالی که روش فراابتکاری **SEO** و **KA** مقادیر **RPD** کمتری دارند. به خصوص روش **SEO** دارای کمترین مقادیر میانگین شکاف به اندازه‌ی کمتر از ۰,۰۰۳ می‌باشد که عملکرد برجسته این روش را در بین کلیه روش‌ها نشان می‌دهد.



شکل ۱۱. میانگین RPD الگوریتم‌های فراابتکاری نسبت به بهترین جواب $\beta = -0.1, \alpha = 0.1$

جدول۲. نتایج به دست آمده توسط الگوریتم‌های فراابتکاری در حالت چند دربی $\beta = -0.1\ \alpha = 0.1$

set	Lingo	GA		ICA		KA		SEO	
		Best	Average	Best	Average	Best	Average	Best	Average
1	7137.87	7137.87	8110.75	7182.05	9827.37	7137.87	10538.38	7137.87	7165.31
2	3461.14	3461.14	3912.01	3461.14	4183.71	3461.14	4124.88	3461.14	3975.78
3	8737.82	8737.82	13379.08	8737.82	10819.52	8737.82	13312.48	8737.82	9307.97
4	2642.56	2642.56	3886.06	2642.56	3299.72	2642.56	3545.48	2642.56	2643.59
5	7102.79	7102.79	11003.73	7102.79	9981.19	7102.79	9640.19	7102.79	9686.77
6	3428.07	3428.07	5246.49	3428.07	3990.78	3428.07	3940.18	3428.07	4180.68
7	6069.31	6069.31	9315.83	6069.31	7893.90	6069.31	8533.98	6069.31	7033.36
8	15282.85	15282.8	15989.26	15282.8	16347.45	15282.85	23579.72	15282.85	16317.9
9	3926.20	4029.62	5757.54	3998.52	4902.18	4010.18	4679.80	3926.20	5330.97
10	7056.30	7242.55	10914.38	7172.05	9587.23	7146.96	8310.78	7163.25	8176.33
11	9751.14	9848.59	11000.31	9848.59	11113.36	9891.45	11172.72	9926.63	10494.6
12	5186.40	5242.95	6778.57	5236.65	5980.92	5230.80	7030.10	5226.68	6815.42
13	10675.54	10725.5	16756.42	10686.7	12279.09	10696.64	11697.66	10702.80	14842.7
14	NA	16111.3	19417.41	16159.2	18400.78	16119.77	18550.93	15943.17	19945.4
15	NA	14094.9	17570.99	14113.1	17260.29	13994.32	16036.80	13497.49	14863.6

16	NA	24887.3	26584.51	24856.7	29720.56	24667.78	27982.42	23888.43	30404.4
17	NA	6554.97	7018.08	6537.70	9164.14	6522.19	6824.60	6327.42	6403.30
18	NA	12654.3	18597.70	12377.2	14688.97	12321.78	16197.13	12292.61	15158
19	NA	33315.3	43632.01	33063.8	52268.64	33012.71	38820.84	33561.94	36530.6
20	NA	23566.9	28098.43	23371.1	27165.85	23456.22	31233.73	23526.78	27995.73
21	NA	10814.2	13910.25	10670.3	11337.21	10676.07	16499.55	10748.43	13656.20
22	NA	17984.7	25467.77	17896.1	19665.61	17937.12	23339.92	17839.52	19886.89
23	NA	91945.9	96960.99	90858.8	135852.47	90906.87	127595.4	91839.80	115090.8
24	NA	18577	23727.78	18433.6	25127.57	18487.39	26675.49	18100.20	19393.75
25	NA	51769.2	58369.33	51265.2	78371.27	51085.88	66088.52	51202.30	58722.37
26	NA	13411.7	15462.49	13411.8	14939.10	13470.12	16431.44	13518.03	18330.90
27	NA	9377.41	11327.26	9366.14	10921.99	9355.68	9883.59	9348.32	10713.71
28	NA	5049.21	6226.66	5030.96	5803.88	5035.62	5303.25	5026.06	5391.70
29	NA	12144.4	13556.89	12180.5	16435.21	12047.00	13114.75	12017.68	12603.61
30	NA	8360.63	9286.27	8371.37	10254.61	8300.94	11414.77	8006.24	10000.17
31	NA	15055	21243.16	15036.5	17943.52	14922.24	16526.23	14450.79	17919.30
32	NA	22941.9	29328.67	22881.4	29135.22	22827.14	33123.54	22145.46	30702.90
33	NA	18094.3	27421.26	17698	25515.31	17618.80	21782.66	17577.09	21731.10
34	NA	6203.37	6301.49	6156.55	6291.14	6147.03	6385.34	6149.30	6407.07
35	NA	7641.22	7685.02	7577.71	7666.10	7605.32	7885.94	7528.32	7606.18

جدول۳. نتایج به‌دست آمده توسط الگوریتم‌های فراابتکاری در حالت چند دربی $\boldsymbol{\alpha = 0.5\ \beta = -0.5}$

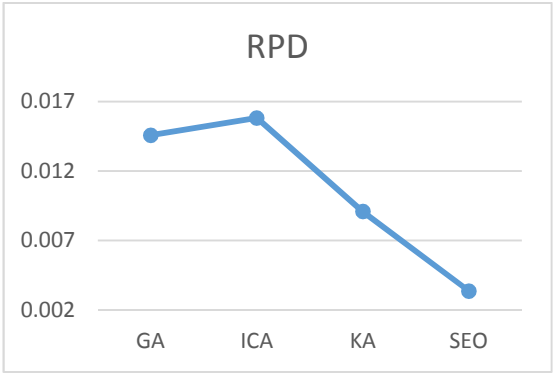
set	Lingo	GA		ICA		KA		SEO	
		Best	Average	Best	Average	Best	Average	Best	Average
1	6619.86	6619.86	8110.75	6619.86	9114.18	6619.86	9773.59	6619.86	6645.31
2	3094.40	3094.40	3912.01	3094.40	3740.40	3094.40	3687.80	3094.40	3554.50
3	8497.40	8497.40	13379.08	8497.40	10521.82	8497.40	12946.19	8497.40	9051.86
4	2405.41	2405.41	3886.06	2405.41	3003.59	2405.41	3227.30	2405.41	2680.82
5	6100.73	6100.73	11003.73	6100.73	8573.04	6100.73	8280.15	6100.73	8320.16
6	3231.33	3251.98	5246.49	3231.33	3761.75	3231.33	3714.04	3231.33	3940.74
7	5882.16	6055.04	9315.83	5882.16	7650.49	5882.16	8270.83	5882.16	6816.49
8	15231.24	15328.6	15989.26	15374.3	16292.25	15231.24	23500.09	15231.24	16262.86
9	3687.62	3755.54	5757.54	4029.62	4604.29	3766.50	4395.43	3687.62	5007.03
10	6931.97	6956.30	10914.38	7242.55	9298.82	6931.97	8060.77	6947.76	7930.36
11	4740.72	4788.09	4861.31	4788.09	4856.99	4808.93	4861.42	4826.03	4921.62
12	10271.53	10371	10383.52	10383.5	10471.23	10359.46	10474.60	10351.31	10432.95
13	13224.02	13237.9	14781.29	13285.9	14831.41	13250.16	13343.42	13257.79	14934.42
14	NA	15230.7	15185.65	15185.6	15347.58	15193.63	15185.65	15027.18	15206.48
15	NA	11977.4	12078.18	11962.1	12081.94	11876.64	12194.32	11455.00	11532.70
16	NA	18849.1	19212.45	18872.3	19090.21	18705.84	19303.67	18114.85	18398.44
17	NA	10106.3	10229.19	10079.6	10328.57	10055.74	10513.34	9755.44	9969.15

18	NA	8150.60	8153.92	7972.11	8036.49	7936.41	8323.80	7917.62	8028.79
19	NA	17853	17853.03	17718.3	17786.78	17690.89	17967.23	17985.21	18093.42
20	NA	23867.1	24705.37	23668.7	24846.22	23754.94	25790.23	23826.41	24991.92
21	NA	16278.3	16450.31	16061.8	16352.63	16070.38	16722.21	16179.31	16553.63
22	NA	15229.8	15523.27	15154.7	15535.19	15189.44	15524.53	15106.79	15466.32
23	NA	74195	74366.52	73317.8	73590.28	73356.59	74595.78	74109.40	74465.73
24	NA	11967.9	12003.48	11875.5	11978.14	11910.18	12132.48	11660.75	11798.13
25	NA	28364.2	28510.38	28088.1	28292.61	27989.87	28443.95	28053.66	28244.59
26	NA	14175.2	14242.31	14275.2	14317.80	14236.89	14242.31	14290.36	14482.09
27	NA	8508.79	8584.05	8619.02	8633.41	8499.28	8751.06	8523.48	8603.62
28	NA	9358.93	9767.63	9392.87	9842.12	9367.59	9771.02	9387.50	9942.52
29	NA	12544.3	13193.98	12507.1	13921.56	12406.79	13901.35	12376.60	13786.07
30	NA	7641.72	7670.45	7631.91	7751.12	7577.42	7828.61	7308.41	7397.96
31	NA	4419.63	4460.47	4425.07	4487.37	4386.04	4533.00	4247.47	4333.31
32	NA	9222.90	9265.66	9247.26	9282.86	9201.02	9348.98	8926.25	9020.10
33	NA	17698	18227.86	18094.3	17904.09	17618.80	18312.09	17577.09	17839.97
34	NA	6156.55	6301.49	6203.37	6291.14	6147.03	6385.34	6249.30	6407.07
35	NA	7577.71	7685.02	7641.22	7666.10	7605.32	7885.94	7528.32	7606.18

جدول ٤. نتایج به‌دست آمده توسط الگوریتم‌های فراابتکاری در حالت چند دربی $\alpha = 0.9$ $\beta = -0.9$

set	Lingo	GA		ICA		KA		SEO	
		Best	Average	Best	Average	Best	Average	Best	Average
1	6307.23	6307.23	7727.71	6307.23	8683.76	6307.23	9312.03	6307.23	6331.48
2	2889.22	2889.22	3652.62	2889.22	3492.39	2889.22	3443.28	2889.22	3318.81
3	8329.91	8329.91	13115.37	8329.91	10314.43	8329.91	12691.01	8329.91	8873.44
4	2273.06	2273.06	3672.24	2273.06	2838.32	2273.06	3049.72	2273.06	2533.31
5	5518.48	5518.48	9953.54	5518.48	7754.83	5518.48	7489.90	5518.48	7526.09
6	3108.88	3128.75	5047.68	3108.88	3619.20	3108.88	3573.30	3108.88	3791.41
7	5780.54	5950.43	9154.90	5834.82	7518.33	5780.54	8127.95	5780.54	6698.73
8	15197.23	15594.3	15953.56	15439.9	16255.87	15197.23	23447.61	15197.23	16226.54
9	3526.31	3591.27	5505.69	3853.36	4402.89	3601.74	4203.16	3526.31	4788.01
10	6644.55	7667.87	10461.84	6942.26	8913.27	6644.55	7726.55	6659.69	7601.55
11	4517.92	4580.07	4632.85	4585.07	4628.74	4582.93	4632.96	4599.23	4690.33
12	10162.14	10260.6	10272.94	10272.9	10359.72	10249.13	10363.05	10241.07	10321.84
13	11752.68	11765	13136.68	12807.7	13181.22	11775.90	11858.79	11982.69	13272.77
14	NA	14785	14741.24	14741.2	14898.43	14748.99	14841.24	14587.41	14761.47
15	NA	11881	11946.66	11831.8	11950.38	12047.32	12161.54	11330.27	11407.12
16	NA	18894.8	19055.13	18717.8	18933.89	18552.66	19145.59	17966.51	18247.78
17	NA	10139.1	10261.19	10412.6	10959.91	9988.89	10443.45	9690.59	9902.88
18	NA	6310.38	6312.96	6172.19	6222.04	6144.56	6444.48	6130.01	6216.08
19	NA	16503.8	16503.81	16379.2	16442.57	16353.92	16609.38	16726.00	16826.04

20	NA	22541.9	22816.06	21858.7	22946.14	21938.32	23817.96	22004.31	23080.69
21	NA	15118.7	15277.49	14919	15087.36	14826.96	15428.35	14927.45	15272.82
22	NA	14037.7	14308.21	14168.5	14319.20	14000.51	14309.37	13924.33	14255.72
23	NA	68320.7	68478.66	67512.9	67763.87	67548.69	68689.77	68241.90	68570.01
24	NA	11209.6	11742.32	11124.6	11519.01	10956.49	11160.99	10727.03	10853.41
25	NA	26803.5	26941.64	26842.6	26935.86	26449.78	26878.87	26810.05	26990.48
26	NA	13638	13702.60	13734.2	13775.23	13697.39	13702.60	13848.83	13933.29
27	NA	8361.80	8435.76	8470.13	8484.26	8352.46	8599.88	8376.24	8454.99
28	NA	9338.03	9745.82	9371.90	9820.14	9346.68	9749.21	9466.53	9920.31
29	NA	12496.4	13143.62	12459.4	13868.42	12359.44	13848.29	12329.36	13733.45
30	NA	7307.45	7334.93	7298.07	7412.07	7245.96	7486.17	6988.72	7074.35
31	NA	4336.38	4275.53	4241.59	4301.31	4204.18	4345.05	4071.35	4153.64
32	NA	8409.68	8448.68	8431.90	8464.36	8389.74	8524.65	8139.19	8224.77
33	NA	43026.4	44314.44	43989.7	43527.31	42833.73	44519.22	42732.35	43371.43
34	NA	6101.42	6245.07	6147.83	6234.82	6092.00	6328.17	6215.35	6349.71
35	NA	5866.84	5949.93	5916.01	5935.27	5888.22	6105.48	5828.60	5888.88

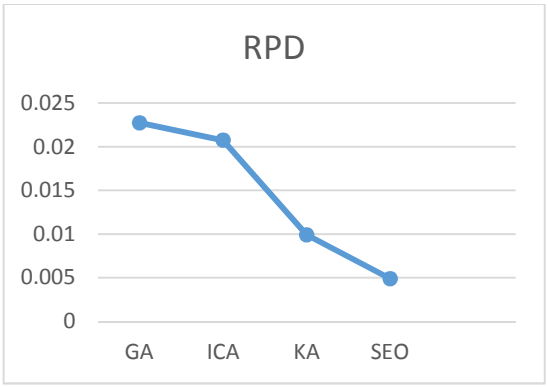


شکل ۱۲. میانگین RPD الگوریتم‌های فراابتکاری

$\beta = -0.5 \ \alpha = 0.5$

جدول۵. میانگین زمان پردازش

set	GA	ICA	KA	SEO
21	21.01	28.81	12.28	6.43
22	22.91	27.77	13.03	4.72
23	17.30	39.78	11.66	11.52
24	36.40	84.81	34.32	10.36
25	28.06	94.78	32.15	17.61
26	39.53	81.39	58.80	18.68
27	70.61	113.21	82.52	25.85
28	66.10	137.69	109.68	24.33
29	75.45	130.44	86.96	23.21
30	132.31	152.26	120.08	35.68
31	134.27	199.95	132.14	42.91
32	153.61	205.33	137.84	41.87
33	115.10	206.55	172.99	46.29
34	150.39	344.80	226.00	65.47
35	145.98	409.23	223.84	82.31



شکل ۱۳. میانگین RPD الگوریتم‌های فراابتکاری

$\boldsymbol{\beta = -0.9 \ \alpha = 0.9}$

همچنین جهت بررسی زمان پردازش میانگین زمان صرف شده جهت یافتن بهترین جواب‌ها در سی بار اجرای الگوریتم محاسبه شده است. این نتایج در جدول (۵) نشان داده شده است. همانطور که مشاهده می‌شود، الگوریتم *SEO* دارای بالاترین سرعت همگرایی می‌باشد.

٦- نتیجه‌گیری

این مطالعه به بررسی مسأله زمانبندی کامیون‌ها در سیستم انبار متقاطع با یک انبار موقت در نزدیکی درب خروجی می‌پردازد. در واقع هدف مسأله حداقل‌سازی زمان اجرای کل عملیات و تعیین بهترین توالی برای کامیون‌های ورودی و خروجی می‌باشد. در این تحقیق مدل اولیه با در نظر گرفتن اثر یادگیری و زوال‌پذیری وارد فاز جدیدی از مسائل زمانبندی کامیون‌های انبار متقاطع شده است. تاکنون یک فرض رایج در مسائل برنامه‌ریزی و زمان‌بندی پیشرفته این بود که زمان انجام یک فرایند، ثابت و مستقل از جایگاهش در توالی تولید است. در حالی که بسیاری از محققین با بررسی برنامه زمان‌بندی تجربی و تئوری به این نتیجه رسیده‌اند که زمان‌های پردازش کارها تراکم پذیر هستند که یکی از رویکردها برای کاهش زمان پردازش کارها و در نظر گرفتن تراکم پذیری و استفاده از پدیده یادگیری است. به همین دلیل در این مقاله برای اولین بار عامل‌های انسانی یادگیری و زوال‌پذیری را در امر مدل سازی دخیل و با در نظر گرفتن آن، مدل ریاضی ارایه شده از نظر ساختاری به واقعیت نزدیک‌تر شده است.

نوآوری این مقاله در نظر گرفتن اثرات یادگیری و زوال‌پذیری در مدل مساله زمانبندی انبار متقاطع برای اولین بار است. همچنین در این مقاله روش‌های فراابتکاری بسیار قوی ارایه شده‌اند که با کمترین زمان و با بهترین عملکرد نسبت به دیگر رویکردهای فراابتکاری ارایه شده در مدل‌های مشابه به اهداف دست می‌یابند. همچنین با بررسی مرور ادبیات کارهای گذشته، برای حل مسائل در مقیاس بزرگ از چهار الگوریتم فراابتکاری شامل الگوریتم‌های ژنتیک (GA)، رقابت استعماری (ICA)، کشتل (KA) و بهینه سازی مهندسی اجتماعی (SEO) استفاده شده است. سپس با استفاده از روش تاگوچی به تنظیم پارامتر الگوریتم‌های فوق پرداخته‌ایم. در نهایت نتایج عددی به دست آمده از تمامی الگوریتم‌های ابتکاری و فراابتکاری مورد بررسی و تحلیل حساسیت قرار گرفته‌اند. الگوریتم‌های فراابتکاری را بر اساس معیارهای بهترین، میانگین جواب‌ها، Rpd و زمان مورد مقایسه قرار داده‌ایم. در نتیجه الگوریتم‌های SEO و الگوریتم KA از نظر کیفیت جواب بهتر از سایر الگوریتم‌ها است.

مسائل مطرح شده برای سیستم‌های فراباراندار و مدل‌های آن‌ها بسیار گسترده هستند و برای آن‌ها می‌توان روش‌های حل مختلفی را در نظر گرفت. همچنین می‌توان با در نظر گرفتن فرضیات جدید، مدل‌های سابق را توسعه داد و آن‌ها را به شرایط واقعی و کاربردی نزدیک‌تر کرد. لذا، موارد زیر را می‌توان به‌عنوان پیشنهادات آتی برای این تحقیق درنظر گرفت:

– ارایه‌ی مدل زمانبندی چند هدفه و همچنین روش‌های ابتکاری و فراابتکاری تلفیقی و جدید

–در نظر گرفتن عدم قطعیت در پارامترهای ورودی مسأله به همراه معرفی منابع جدید تغییرپذیری در سیستم با بررسی جزئیات بیشتر ترمینال‌ها موجب افزایش سطح عدم قطعیت و افزایش استواری مدل در عمل می‌شود.

–حالت سرویس دهی درب‌ها به صورت ترکیبی از درب‌های اختصاصی و مختلط در نظر گرفته شود.

۷- پی‌نوشت‌ها

1. Genetic Algorithm
2. Imperialist Competitive Algorithm
3. Keshtel Algorithm
4. Social Engineering Optimizer

۸- مراجع

-Alpan, G., Ladier, A.-L., Larbi, R., & Penz, B., (2011), "Heuristic solutions for transshipment problems in a multiple door cross docking warehouse", Computers & Industrial Engineering, 61(2), pp.402-408 .

-Amini, A., & Tavakkoli-Moghaddam, R., (2016), "A bi-objective truck scheduling problem in a cross-docking center with probability of breakdown for trucks", Computers & Industrial Engineering, 96, pp.180-191.
-Amini, A., Tavakkoli-Moghaddam, R., & Omidvar, A., (2014), "Cross-docking truck scheduling with the arrival times for inbound trucks and the learning effect for unloading/loading processes", Production & Manufacturing Research, 2(1), pp.784-804.
-Arabani, A. B., Ghomi, S. F., & Zandieh, M., (2011), "Meta-heuristics implementation for scheduling of trucks in a cross-docking system with temporary storage", Expert systems with Applications, 38, pp. 1964-1979.
-Bartholdi III, J. J., & Gue, K. R., (2000), "Reducing labor costs in an LTL cross docking terminal", Operations Research, 48(6), pp.823-832.
-Bellanger, A., Hanafi, S., & Wilbaut, C., (2013),"Three-stage hybrid-flow shop model for cross-docking", Computers & Operations Research, 40(4), pp.1109-1121.
-Boysen, N., & Fliedner, M., (2010), "Cross dock scheduling: Classification, literature review and research agenda", Omega, 38(6), pp.413-422 .
-Dulebenets, M. A., (2019), "A Delayed Start Parallel Evolutionary Algorithm for just-in-time truck scheduling at a cross-docking facility", International Journal of Production Economics, 212, pp.236-258.
-Fonseca, G. B., Nogueira, T. H., & Ravetti, M. G., (2019), "A hybrid Lagrangian metaheuristic for the cross-docking flow shop scheduling problem", European Journal of Operational Research, 275(1), pp.139-154.
-Golshahi-Roudbaneh, A., Hajiaghaei-Keshteli, M., & Paydar, M. M., (2017), Developing a lower bound and strong heuristics for a truck scheduling problem in a cross-docking center, Knowledge-Based Systems, 129, pp.17-38.
-Gupta J.N.D., Gupta S.K., (1988), "Single facility scheduling with nonlinear processing times, Computers and Industrial Engineering 14, pp.387–393.
-Holland, J. H., (1992), "Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence", MIT press.
-Huang, X., Wang, M. Z., & Ji, P., (2014), "Parallel machines scheduling with deteriorating and learning effects", Optimization Letters, pp.1-8.
-Ladier, A. L., & Alpan, G., (2016), "Cross-docking operations: Current research versus industry practice, Omega, 62, pp.145-162.
-Maknoon, M. Y., & Baptiste, P., (2010), "Moving freight inside cross docking terminals", Paper presented at the 2010 8th International Conference on Supply Chain Management and Information.
-Mir, M. S. S., & Rezaeian, J., (2016), "A robust hybrid approach based on particle swarm optimization and genetic algorithm to minimize the total machine load on unrelated parallel machines", Applied Soft Computing, 41, pp.488-504.
-Mousavi, S. M., Tavakkoli-Moghaddam, R., & Jolai, F., (2013), "A possibilistic programming approach for the location problem of multiple cross-docks and vehicle routing scheduling under uncertainty", Engineering Optimization, 45(10), pp.1223-1249.
-Molavi, D., Shahmardan, A., & Sajadieh, M. S., (2018), "Truck scheduling in a cross docking systems with fixed due dates and shipment sorting", Computers & Industrial Engineering, 117, pp.29-40.
-Rijal, A., Bijvank, M., & de Koster, R., (2019), "Integrated scheduling and assignment of trucks at unit-load cross-dock terminals with mixed service mode dock doors", European Journal of Operational Research, 278(3), pp.752-771.
-Rostami, M., Pilerood, A. E., & Mazdeh, M. M., (2015), "Multi-objective parallel machine scheduling problem with job deterioration and learning effect under fuzzy environment", Computers & Industrial Engineering, 85, pp.206-215.
-Rohrer, M., (1995), "Simulation and cross docking", Paper presented at the Simulation Conference Proceedings, Winter.
-Song, K., & Chen, F., (2007), "Scheduling cross docking logistics optimization problem with multiple inbound vehicles and one outbound vehicle", Paper presented at the 2007 IEEE International Conference on Automation and Logistics.

scheduling with blocking. Interdisciplinary Journal of Contemporary Research in usiness, 4(8), pp.394-402.
-Vahdani, B., & Zandieh, M., (2010), "Scheduling trucks in cross-docking systems: Robust meta-heuristics", Computers & Industrial Engineering, 58(1), pp.12-24.
-Wisittipanich, W., & Hengmeechai, P., (2017), "Truck scheduling in multi-door cross docking terminal by modified particle swarm optimization", Computers & Industrial Engineering.
-Xu, J., Xu, X., & Xie, S. Q., (2011), "Recent developments in Dual Resource Constrained (DRC) system research", European Journal of Operational Research, 215(2), pp.309-318.
-Yu, W., (2002), "Operational strategies for cross docking systems".
-Yu, W., & Egbelu, P. J., (2008), "Scheduling of inbound and outbound trucks in cross docking systems with temporary storage", European Journal of Operational Research, 184(1), pp.377-396.
-Zhao, Q. H., & Cheng, T. E., (2009), "An analytical study of the modification ability of distribution centers", European Journal of Operational Research, 194(3), pp.901-910.
-Seyedi, I., Hamedi, M., Tavakkoli-Moghaddam, R., (2019), "Truck Scheduling in a Cross-Docking Terminal by Using Novel Robust Heuristics, International Journal of Engineering, 32(2), pp.296-305.
-Seyedi, I., Hamedi, M., & Tavakkoli-Moghadaam, R. (2021). Developing a mathematical model for a multi-door cross-dock scheduling problem with human factors: A modified imperialist competitive algorithm. Journal of Industrial Engineering and Management Studies, 8(1), pp.180-201.
-Seyedi, I. & Maleki-Daronkolaei, A. (2013). Solving a two-stage assembly flowshop scheduling problem to minimize the mean tardiness and earliness penalties by three meta-heuristics. Caspian Journal of Applied Sciences Research, 2(4), pp. 67-78.
-Seyedi, I., Mirzazadeh, S., Maleki-Daronkolaei, A., Mukhtar, M., & Sahran, S. (2016). An inventory model with reworking and setup time to consider effect of inflation and time value of money. Journal of engineering science and Technology, 11(3), pp. 416-430.
Seyedi, I., Maleki-Daronkolaei, A., & Kalashi, F. (2012). Tabu search and simulated annealing for new three-stage assembly flow shop

Solving Truck Scheduling Optimization Problem in Multi-Door Cross Dock with Learning Effect and Deteriorating Jobs Using Social Engineering Optimizer

Iman Seyedi, Assistant Professor, Department of Industrial Engineering, Payame Noor University, Tehran, Iran.

Maryam Hamedi, Assistant Professor, Department of Industrial Engineering, Payame Noor University, Tehran, Iran.

Reza Tavakkoli-Moghaddam, Professor, School of Industrial Engineering, College of Engineering, University of Tehran, Tehran, Iran.

E-mail: iman_seyyedi@pnu.ac.ir

Received: October 2021- Accepted: May 2022

ABSTRACT

Each supply chain consists of procurement, production, and distribution stages. The cross-docking system is a new strategy at the distribution stage to improve customer response time by moving products directly from pickup trucks to delivery trucks. Generally, both machine and human resources are needed for an activity to be done. Many researchers have developed numerous planning methods for cross-docking systems, but human resource constraints have been mainly ignored. In this paper, for the first time, we examine the problem of truck scheduling in multi-door cross-dock, considering the learning effects and the deterioration of tasks to fill the gap between theoretical planning models and what is happening in the real world. We have proposed a mixed integer programming model for this problem. According to the research literature, with increasing the size of the problem, the complexity of the integer programming model is expanding rapidly. The exact methods can hardly achieve the optimal solution. To solve large-scale problems, five meta-heuristic algorithms are used, including Genetic Algorithms (GA), Imperial Competitive Algorithm (ICA), Keshtel Algorithm (KA), and Social Engineering Optimization (SEO). Finally, the numerical results obtained from all meta-heuristic algorithms are analyzed. We compare the meta-heuristic algorithms based on the best, average, Rpd, and time criteria. As a result, the SEO and KA algorithm performed better than the other algorithms in solution quality.

Keywords: Cross Dock, Scheduling, Learning Effect, Deterioration, Meta-Heuristic