مجله تحقیق در عملیات در کاربردهای آن، سال نوزدهم، شماره چهارم (پیاپی ۷۵)،زمستان ۰۱، ص ص ۶۱-۳۷
شاپا ۷۲۸۶-۲۲۵۱

افزایش قابلیت جستجو در الگوریتم بهینه‌سازی رقابت استعماری برای برنامه زمانبندی کامیون‌ها در سیستم انبار متقاطع

سید ایمان سیدی[1*]، مریم حامدی[2]، رضا توکلی مقدم[3]
۱- استادیار، گروه مهندسی صنایع، دانشگاه پیام نور، تهران، ایران
۲- استادیار، گروه مهندسی صنایع، دانشگاه پیام نور، تهران، ایران
۲- استاد، دانشکده مهندسی صنایع، دانشکدگان فنی، دانشگاه تهران، تهران، ایران

رسید مقاله: ۲۷ آذر ۱۴۰۰
پذیرش مقاله: ۱۸ خرداد ۱۴۰۱

چکیده

در محیط توزیع امروزه، انبار متقاطع به دلیل نقش اساسی آن در کاهش هزینه در زنجیره‌های تامین، مورد توجه بسیاری از محققان قرار گرفته است. سیستم انبار متقاطع با از بین بردن ذخیره‌سازی و مرتب‌سازی سفارشات، هزینه توزیع را کاهش می‌دهد. در این مقاله به بررسی مساله زمانبندی کامیون‌ها در انبار متقاطع پرداخته‌ایم. با توجه به ادبیات تحقیق مساله زمانبندی انبار متقاطع از جمله مسایل Np-hard می‌باشد؛ بنابراین در این مقاله، یک رویکرد ترکیبی فراابتکاری جدید برای حل مساله ارایه‌شده است. الگوریتم رقابت استعماری (ICA) یک الگوریتم اجتماعی-سیاسی است که از رقابت امپریالیستی الهام گرفته شده است. اما، هنگام جستجو در یک محیط پیچیده، کارایی آن به میزان قابل توجهی کاهش می‌یابد و این محدودیت، الگوریتم را برای رسیدن به یک راه حل خوب محدود می‌کند. در این مقاله مکانیسم جستجوی جدیدی برای حل این مشکل معرفی شده است. این الگوریتم بر مبنای روش تجزیه و تحلیل مولفه اصلی (PCA) ارایه‌شده است که در آن PCA در واقع یک مجموعه با بعد پایین از ویژگی‌ها را از یک مجموعه دارای بعد بالا استخراج می‌کند تا به ثبت اطلاعات بیشتر با تعداد کمتری از متغیرها کمک کند و به این دلیل این روش PCICA نام‌گذاری شده است. نتایج به‌دست آمده و مقایسه آن با روش‌های ICA، SA و GA نشان می‌دهد که PCICA به‌طور قابل ملاحظه‌ای نسبت به سایرین بهتر عمل کرده و می‌تواند راه حل‌های خوبی پیدا کند.

کلمات کلیدی: انبار متقاطع، زمانبندی، الگوریتم رقابت استعماری، تجزیه و تحلیل مؤلفه اصلی.

۱ مقدمه

تاکنون استراتژی‌های لجستیک بسیاری برای مدیریت زنجیره تامین (SCM) ارایه شده است. تهیه، تولید و توزیع سه مرحله اصلی است که در هر زنجیره تامین مشترک است [۱]. انبار متقاطع یک استراتژی لجستیکی است که

* عهده‌دار مکاتبات
آدرس الکترونیکی: iman_seyyedi@pnu.ac.ir

[Downloaded from jorlu.ir on 2026-11-30]
[DOI: 10.52547/jamlu.19.4.37]

در آن محصولات از یک تامین‌کننده یا کارخانه تولید مستقیماً با حداقل بارگذاری یا مدت ذخیره در اختیار مشتریان قرار می‌گیرد. در عملیات انبارداری حداکثر هزینه مربوط به ذخیره‌سازی و بازیابی است که انبار متقاطع امکان از بین بردن آنها را دارد [۲]. علاوه بر این، کاهش موجودی کالا و همچنین کاهش هزینه‌های حمل و نقل از دیگر مزایای انبار متقاطع است که می‌تواند برای شرکت‌هایی با گنجایش انبار کم مفید باشد. از این رو انبار متقاطع می‌تواند با کمک به آنها در عرضه سریع‌تر و کارآمدتر محصولات به بازار، زنجیره‌های عرضه را ساده کند. مشهورترین مورد استفاده برای انبار متقاطع شرکت وال مارت است. این استراتژی به وال مارت کمک کرده است تا سهم بازار در تجارت و در نهایت سودآوری آن را افزایش دهد.

برای مرور جامع مفهوم انبار متقاطع و دستورالعمل‌های اجرای موفق آن می‌توان به آثار ارایه‌شده توسط بوسن و فلیدر [۳]، استفان و بوسن [۴]، ون بل و همکاران[۵] و لادیر و آلپان [۶] اشاره کرد که مساله انبار متقاطع را بررسی و طبقه‌بندی کردند.

در جامع‌ترین کارهای ارایه‌شده می‌توان به کار آگوستینا و همکاران [۷] که به مرور مسایل انبار متقاطع با عنوان مدل‌های ریاضی برای برنامه‌ریزی انبار متقاطع اشاره نمود. آن‌ها مسایل انبارهای متقاطع به سه سطح تصمیماتی عمده تقسیم کرده اند: سطح استراتژیک[1]، سطح تاکتیکی[2] و سطح عملیاتی.

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 [۸]. برنامه‌ریزی و زمانبندی در هر جنبه‌ای، یک انقلاب در صرفه‌جویی در زمان به حساب می‌آید که کاربردهای گسترده‌ای در محیط‌های تولیدی و خدماتی دارد[۹]. به طور کلی مسایل زمان‌بندی به صورت تخصیص منابع در طول زمانی مشخص برای انجام مجموعه‌ای از وظایف مربوط به یک فرآیند تعریف می‌شود[۱۰]. 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 مسایل 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. مقاله حاضر در سطح عملیاتی و به ویژه به مساله برنامه‌ریزی کامیون‌ها می‌پردازد.

در ادامه این مقاله مروری بر ادبیات موضوع خواهیم داشت و مطالعات پیشین در رابطه با موضوع بحث، مورد بررسی قرار خواهد گرفت. سپس مساله مورد تحقیق و فرضیات مساله، به طور جامع معرفی و مدل ریاضی ارایه‌شده کاملا تشریح خواهد شد. سپس به بررسی روش‌های حل پیشنهادی پرداخته شده است. در ادامه به بررسی و تجزیه و تحلیل نتایج و عملکرد روش‌های پیشنهادی پرداخته شده است. در انتها نیز به جمع‌بندی و نتیجه‌گیری تحقیق می‌پردازیم. همچنین پیشنهاداتی جهت تحقیقات آتی بیان خواهد شد.

1 Strategic level
2 Tactical level

[Downloaded from jorlu.ir on 2026-11-30]

[DOI: 10.52547/jamlu.19.4.37]

۲ پیشینه تحقیق

مساله زمان‌بندی کامیون‌ها در انبار متقاطع شامل تخلیه بار از کامیون‌های ورودی در بارانداز دریافت، انجام فرایندهای درون انبار مانند شناسایی، طبقه‌بندی و حمل ونقل و در انتها انتقال محصولات به کامیون‌های خروجی در بارانداز می‌شود. مدل‌های مختلفی برای انبار متقاطع وجود دارد که وابسته به تعداد جایگاه‌های دریافت و حمل، الگوی حرکت کامیون‌ها در جایگاه‌های دریافت و حمل و همچنین مجاز بودن یا نبودن انبار موقت می‌باشد. به عنوان یکی از مهم‌ترین مطالعات صورت گرفته در این زمینه، می‌توان به کار ارایه‌شده توسط یو[1] [۱۱] اشاره نمود. او سه عامل برای مساله زمان‌بندی شامل تعداد درب‌ها، الگوی قرار گرفتن کامیون‌های ورودی و خروجی و وجود انبار موقت (مجاز بودن انبار موقت یا نبودن) در نظر گرفت. او هدف از تحقیق خود را پیدا کردن بهترین سکو در بارانداز تقاطعی یا برنامه‌ریزی دنباله‌ای برای کامیون‌های ورودی و خروجی برای به حداقل رساندن زمان عملیات در نظر گرفت. لی و همکاران [۱۲] یک مدل زمان‌بندی کامیون‌های ورودی و خروجی برای کمینه کردن فاصله‌ها پیشنهاد داده‌اند. برای حل مدل موردنظر از الگوریتم ژنتیک استفاده کرده‌اند.

یو و اگبلو[2] [۲] را می‌توان از مهم‌ترین پیشگامان عرصه‌ی زمان‌بندی کامیون‌ها در انبار متقاطع نامید. آن‌ها یک مدل زمان‌بندی برای یک بارکش در بارانداز ورودی و خروجی و ترکیب آن‌ها با تخصیص محصولات ارایه کرده‌اند. مدل آن‌ها می‌تواند زمان زمان‌بندی بارکش و تخصیص محصول را مشخص نماید. هدف مدل کمینه‌سازی زمان عملیات کل است، در حالتی که یک انبار موقت میانی برای نگهداری کالاها به صورت موقتی در باراندازهای ارسال وجود دارد. به‌طور 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 می‌توان به کارهای ارایه‌شده توسط سانگ و چن[3] [۱۳] ، عربانی و همکاران[4] [۱۴]، مکنون و باپتیست[5] [۱۵]، وحدانی و زندیه[6] [۱۶]، آلپان و همکاران[7] [۱۷]، بلنگر و همکاران[8] [۱۸] و غیره اشاره نمود.

وحدانی و زندیه [۱۶] با در نظر گرفتن مفروضات مدل یو و اگبلو [۲] پنج روش فراابتکاری را جهت حل مسایل بزرگ مورد بررسی قرار داده‌اند که این روش‌ها عبارتند از: GA، TS، SA، الگوریتم الکترومغناطیس[9] (EMA) و روش جستجوی همسایگی (VNS)[10]. لازم به ذکر است که نگارندگان از تکنیک روش‌شناسی سطح پاسخ (RSM)[11] برای طراحی استوار الگوریتم‌های فراابتکاری اشاره‌شده استفاده کرده‌اند. آلپان و همکاران[12] [۱۷] یک مساله‌ی زمانبندی را مورد مطالعه قرار داده که بنابر اشاره‌ی نویسندگان دارای دو تفاوت

1 Yu
2 Yu and Elbelu
3 Song
4 Arabani
5 Maknoon
6 Vahdani
7 Alpan
8 Bellanger
9 Electromagnetism-like Algorithm
10 Variable Neighbourhood Search
11 Response Surface Methodology
12 Alpan

[Downloaded from jorlu.ir on 2026-11-30]

[DOI: 10.52547/jamlu.19.4.37]

مهم با کارهای ارایه‌شده‌ی پیش از آن است: الف) تابع هدف مدنظر از جنس هزینه بوده و به زمان ارتباط(سهم)ی ندارد، ب) توقف در عملیات بارگذاری مجاز است. از مفروضات مهم این مساله می‌توان به این نکته اشاره نمود که در این تحقیق کامیون‌های ورودی براساس سیاست FIFO[1] به درب‌ها تخصیص داده می‌شوند می‌شوند و از طرف دیگر هزینه‌های مدنظر مستقل از زمان هستند. در این مساله زمان‌بندی تنها برای کامیون‌های خروجی مورد مطالعه قرار می‌گیرد؛ زیرا توالی کامیون‌های ورودی براساس برنامه‌های تولید از پیش مشخص است. کمینه کردن 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 مساله‌ی فوق الذکر است که جهت دستیابی به آن از یک برنامه‌ریزی پویا با به‌کارگیری یک حد برای جواب بهینه استفاده شده است.

مدنی اصفهانی و همکارانش[۱۹] به زمان‌بندی انبار متقاطع چندگانه با محدود بودن ظرفیت انبار متقاطع در حالی که یک ذخیره‌سازی موقت در جلوی درب حمل وجود دارد، پرداخته‌اند. آن‌ها یک مدل MIP که برگرفته از مدل یو [۱۱] است، پیشنهاد دادند و برای حل مدل مورد نظر از دو الگوریتم فراابتکاری تبرید شبیه‌سازی شده و کرم شبتاب(FA)[2] استفاده کردند. سپس برای ارزیابی عملکرد الگوریتم‌های پیشنهادشده از طراحی آزمایشات تاگوچی استفاده کرده‌اند که الگوریتم کرم شبتاب نسبت به الگوریتم تبرید شبیه‌سازی‌شده عملکرد بهتری داشت. شکل زیر یک انبار متقاطع چندگانه را نشان می‌دهد.

کشتزری و همکارانش[3][۲۰] به ارایه یک مدل بهبوددهنده و یک الگوریتم فراابتکاری ترکیبی برای زمانبندی کامیون‌ها برای مسایل انبار متقاطع پرداخته‌اند. آن‌ها یک مدل MIP جدید برای مساله خود استفاده کرده‌اند. با استفاده از حل‌کننده‌های بهینه‌سازی تجاری، عملکرد مدل پیشنهادشده را با مدل در دسترس برای حل مثال‌های کوچک مقایسه کرده‌اند. و برای حل مثال‌های بزرگ، یک الگوریتم فراابتکاری بهینه‌سازی انبوه ذرات ترکیب‌شده با تبرید شبیه‌سازی‌شده پیشنهاد داده‌اند. با استفاده از روش طراحی آزمایشات تاگوچی به تنظیم پارامتر الگوریتم‌های پیشنهادشده پرداخته و سپس عملکرد این الگوریتم‌ها را با دو الگوریتم فراابتکاری در دسترس یعنی ژنتیک و الکترو شبه مغناطیس مقایسه کرده‌اند.

امینی و توکلی مقدم[4][۲۱] به مساله زمان‌بندی کامیون دو هدفه در یک مرکز انبار متقاطع با احتمال خراب شدن کامیون‌ها پرداخته‌اند. مدل اول آن‌ها برگرفته از مدل یو و اگبلو [۲] می‌باشد. همچنین آن‌ها به این مدل فرض جدیدی یعنی موعد تحویل کامیون‌های خروجی اضافه کرده‌اند. هدف مدل موردنظر به حداقل رساندن دیرکردهای کامیون خروجی می‌باشد. در مدل دوم همان طور که گفته شد، فرض خرابی کامیون‌ها مطرح است. در واقع در مدل دوم، فرض می‌شود که هر کامیون با یک احتمال مشخص خراب‌شده و دارای یک زمان متوسط تعمیر است و تعداد خرابی‌های هر کامیون در واحد زمان از یک توزیع پوآسون پیروی کرده و زمان تعمیر آن

1First in first out
2Firefly algorithm(FA)
3Keshtzari and et al
4 Amini and tavakkoli moghadam

[Downloaded from jorlu.ir on 2026-11-30]

[DOI: 10.52547/jamlu.19.4.37]

صفر در نظر گرفته می‌شود. همچنین از روش RSM برای تنظیم پارامتر استفاده‌شده و در نهایت با استفاده از الگوریتم‌های فراابتکاری به حل دو مدل فوق پرداخته‌اند.

گلشاهی- رودبانه و همکاران[۲۲] به توسعه یک کران پایین[1] و الگوریتم‌های ابتکاری قوی برای مساله زمان‌بندی کامیون‌ها در مراکز انبار متقاطع پرداخته‌اند. مدل آن‌ها برگرفته از مدل یو [۱۱] می‌باشد. هدف این مدل پیدا کردن بهترین توالی کامیون‌های ورودی و خروجی است. برای حل مدل مورد نظر از دو الگوریتم ابتکاری و از پنج الگوریتم فراابتکاری زنتیک، تبرید شبیه‌سازی‌شده، کشتل[2] (KA) ، جستجوی فراکتال تصادفی[3] (SFS) و از یک الگوریتم ترکیبی تبرید شبیه‌سازی شده- بهینه‌سازی تجمعی ذرات (SA-PSO) استفاده شد.

فونسکا و همکاران [4] [۲۳] تحقیقی در مورد زمانبندی کامیون‌ها در انبار متقاطع موازی انجام دادند. آنها ابتداً این مساله را به عنوان یک مساله 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 آرام‌سازی 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 مسایل با اندازه‌های کوچک و بزرگ پیدا می‌کند. وحدانی و شهرام‌فرد [5] [۲۴] مساله زمانبندی کامیون‌ها و اختصاص آنها به درب‌ها به طور همزمان را مورد بررسی قرار دادند. آنها همچنین زمان سرویس درب‌ها و تأثیر آن بر زمانبندی را نیز در نظر گرفتند. در این تحقیق، مدل جدیدی برای مسایل مربوط به زمانبندی و تخصیص کامیون‌ها با محدودیت پنجره زمانی برای ورود و خروج کامیون‌ها، درب‌های بارانداز با سرویس‌دهی مختلط و همچنین توالی کامیون‌ها ارایه‌شده است. برای حل این مدل توسعه‌یافته، دو الگوریتم فراابتکاری، الگوریتم ژنتیک و رقابت امپریالیستی ارایه‌شده است. نتایج محاسباتی نشان می‌دهد که چارچوب پیشنهادی منجر به افزایش کل هزینه‌ها می‌شود، اگر چه برنامه‌ریزی دقیق‌تر ارایه می‌نماید. علاوه بر این، نشان می‌دهد که الگوریتم‌های پیشنهادی دارای عملکرد متفاوت بر اساس معیارهای در نظر گرفته شده برای مقایسه هستند.

سیدی و همکاران [6] [۲۵] 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 ارایه‌شده تاکنون در مدل‌های مشابه به اهداف مورد نظر دست می‌یابند. در رویکردهای ارایه‌شده 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.

1 Lower bound
2 keshtel
3 Stochastic fractal search(SFS)
4 Fonseca et al.
5 Vahdani & Shahramfard
6 Seyedi et al.

[Downloaded from jorlu.ir on 2026-11-30]

[DOI: 10.52547/jamlu.19.4.37]

همچنین به دلیل این‌که در محیط واقعی معمولا فضای کوچکی را برای انبارسازی موقت کالاها در نظر می‌گیرند، سیاست روش‌های ابتکاری ارایه‌شده 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 جامع‌ترین کارهای ارایه‌شده تاکنون می‌توان به بویسن و فیلندر [۳]، ون بل و همکاران [۵] و لیدر و آلپان [۶] اشاره نمود.

۳ بیان مساله

در این پژوهش مساله زمان‌بندی کامیون‌ها در محیط یک انبار متقاطع تک دربی در نظر گرفته شده است. مساله شناخته‌شده زمانبندی ماشین‌ها در انبار متقاطع جزء مسایل بهینه‌سازی ترکیبی با پیچیدگی محاسباتی $O(mn!^m)$ است [۱۳]. بویسن و همکاران[1] [۲۶] ثابت کردند که مساله‌ی برنامه‌ریزی کامیون‌ها در یک سیستم انبار متقاطع قویاً NP-hard می‌باشد. از آن‌جایی که این مسایل از نوع مسایل پیچیده می‌باشند و زمان حل بهینه آن‌ها در رده زمان نمایی است در ادبیات تحقیق برای حل مسایل از الگوریتم‌های فراابتکاری استفاده شده است. در این مقاله نیز از الگوریتم‌های فراابتکاری مانند GA[2]، SA[3] ، ICA[4] و همچنین الگوریتم فراابتکاری ترکیبی جدیدی بنام PCICA [5] استفاده شده است و در نهایت نتایج عددی به‌دست آمده از تمامی الگوریتم های فرا ابتکاری مورد بررسی و تحلیل حساسیت قرار گرفته است.

به منظور افزایش عملکرد الگوریتم‌های بهینه‌سازی، محققان الگوریتم‌های ترکیبی را طراحی نموده‌اند. در این روش‌ها، تعداد دو یا بیشتر الگوریتم با هم ترکیب می‌شوند. الگوریتم‌های ترکیبی تمایل دارند که از مزایای همه الگوریتم‌های ترکیب شده استفاده نمایند.

در این مقاله، یک الگوریتم ترکیبی جدید به نام PCICA، که برای افزایش توانایی اکتشاف ICA طراحی‌شده است، ارایه می‌شود. این الگوریتم، ترکیبی از الگوریتم تکاملی رقابت استعماری (ICA) و روش تجزیه و تحلیل مولفه اصلی (PCA[6]) است. در الگوریتم رقابت استعماری (ICA) سیاست همگون‌سازی (جذب)، با استفاده از یک استراتژی یادگیری خطی، موقعیت هر کشور را از طریق یک ترکیب خطی با استفاده از موقعیت قبلی کشور و فاصله از موقعیت امپریالیست آن با کمی زاویه تصادفی به روز رسانی می‌کند. این چنین استراتژی به‌روزسازی بسیار آسان به نظر می‌رسد. با این حال، در هنگام جستجو در فضای یک مجموعه‌ی دارای محدودیت در اکتشاف اطلاعات ناکارآمد است. بنابراین، پیشنهاد شده است که بر اساس تجزیه و تحلیل مولفه اصلی (PCA) است. که در آن از PCA که عمدتا به نحو احسن اطلاعات جمعیت را برای تعیین جهت مولفه‌های اصلی استخراج می‌کند، استفاده می‌شود. تحلیل مولفه اساسی به بیان ساده، روشی برای استخراج متغیرهای مهم (به شکل مولفه) از مجموعه بزرگی متغیرهای موجود در یک مجموعه داده است. با استفاده از این

1 Boysen
2 Genetic Algorithm
3 Simulated Annealing
4 Imperialist competitive algorithm
5 Principal Component Imperialist competitive algorithm
6 Principal Component Analysis

[Downloaded from jorlu.ir on 2026-11-30]

[DOI: 10.52547/jamlu.19.4.37]

قابلیت PCICA می‌تواند اطلاعات موقعیت کشورها را ذخیره‌سازی کند و با استفاده از آن کشورها را به جهت‌های موثرتری هدایت کند.

٤ مدل

همان‌طور که در بخش قبل گفته شد، زمان‌بندی کامیون‌ها و تعیین بهترین ترکیب و توالی کامیون‌های ورودی و خروجی برای کارا بودن انبار متقاطع ضروری است.

مدل ریاضی زیر بر پایه زمان‌بندی کامیون‌ها می‌باشد. این مدل برگرفته از مدل یو و اگلبو [۱۲] است. آن‌ها مساله زمان‌بندی کامیون‌ها را در یک انبار متقاطع تک دربی در حالتی که یک ذخیره‌سازی موقت در جلوی درب حمل وجود دارد، مورد بررسی قرار دادند. مفروضات ابتدایی مساله مورد نظر به صورت زیر است: کامیون‌های ورودی به بارانداز دریافت می‌رسند، بار خود را در بارانداز دریافت تخلیه می‌کنند. محصولات مورد نیاز کامیون‌های خروجی منتقل شده و سپس انبار را ترک می‌کنند. لازم به‌ذکر است، این مساله به فرآیندهای درون انبار کاری ندارد و تنها زمان‌بندی کامیون‌ها را مورد مطالعه قرار می‌دهد. علاوه بر شرایط ذکرشده، مفروضات دیگری نیز در این مساله وجود دارد که به شرح زیر است:

مفروضات و خصوصیات مدل

- تمامی کامیون‌های ورودی و خروجی در لحظه‌ی صفر آماده و مهیا می‌باشند.
- 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.
- یک درب ورودی و یک درب خروجی وجود دارد.
- ظرفیت انبار موقت نامحدود می‌باشد.

همچنین فرض بر این است که اطلاعات زیر از قبل مشخص می‌باشد:

- تعداد و نوع محصولات موجود درکامیون‌های ورودی
- تعداد و نوع محصولات مورد نیاز کامیون‌های خروجی
- ذخیره‌سازی بلند مدت ممنوع است.
- زمان بارگیری و تخلیه محصولات
- زمان انتقال محصولات از جایگاه دریافت به جایگاه ارسال
- زمان تعویض کامیون‌ها

متغیرها و پارامترهای مدل

R: تعداد کامیون‌های ورودی $i = ۱,...,R$

S: تعداد کامیون‌های خروجی $j = ۱,...,S$

[Downloaded from jorlu.ir on 2026-11-30]

[DOI: 10.52547/jamlu.19.4.37]

N: تعداد کالاها $k = ۱,...,N$

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 کامیون‌ها

M: یک عدد بسیار بزرگ

V: زمان جابجایی محصولات از درب دریافت به به درب ارسال

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 jام باشد=۱ در غیر این‌صورت=۰

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

مدل ریاضی

$$Min\ z = T \qquad (۱)$$

s.t.

$$T \geq L_j \qquad for\ all\ \ j \qquad (۲)$$

$$\sum_{j=۱}^{S} X_{ijk} = r_{ik} \qquad for\ all\ \ i,k \qquad (۳)$$

$$\sum_{i=۱}^{R} X_{ijk} = S_{jk} \qquad for\ all\ \ j,k \qquad (۴)$$

$$X_{ijk} \leq M v_{ij} \qquad for\ all\ \ i,j,k \qquad (۵)$$

$$F_i \geq c_i + \sum_k r_{ik} \qquad for\ all\ \ i \qquad (۶)$$

$$c_j \geq F_i + D - M(۱ - P_{ij}) \qquad for\ all\ \ i,j\ \ and\ \ i \neq j \qquad (۷)$$

$$c_i \geq F_j + D - M P_{ij} \qquad for\ all\ \ i,j\ \ and\ \ i \neq j \qquad (۸)$$

$$P_{ii} = ۰ \qquad for\ all\ \ i \qquad (۹)$$

$$L_j \geq d_j + \sum_k s_{jk} \qquad for\ all\ \ i,j\ \ and\ \ i \neq j \qquad (۱۰)$$

$$d_j \geq L_i + D - M(۱ - q_{ij}) \qquad for\ all\ \ i,j\ \ and\ \ i \neq j \qquad (۱۱)$$

[Downloaded from jorlu.ir on 2026-11-30]

[DOI: 10.52547/jamlu.19.4.37]

$$d_i \geq L_j + D - Mq_{ij} \qquad for \quad all \quad i, j \quad and \quad i \neq j \qquad (۱۲)$$

$$q_{ii} = ۰ \qquad for \quad all \quad i \qquad (۱۳)$$

$$L_j \geq c_i + V + \sum_k X_{ijk} - M(۱ - v_{ij}) \qquad for \quad all \quad i, j \qquad (۱۴)$$

$$All\ variables \geq ۰ \qquad (۱۵)$$

در این مدل 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 زمانی‌که آخرین کالا در آخرین کامیون خروجی بارگیری شود پایان می‌پذیرد. محدودیت (۲)نشان‌دهنده این نکته است که زمان تکمیل عملیات بزرگتر مساوی زمانی است که آخرین ماشین ارسال سکوی ارسال را ترک می‌کند. محدودیت‌های (۳) و (۴) نشان می‌دهد که تعداد کل اقلام دریافت‌شده توسط بارکش‌های دریافت با مقدار کل اقلام ارسال‌شده توسط بارکش‌های ارسال برابر است. محدودیت (۵) 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 می یابد)را نشان می‌دهد. محدودیت‌های(۶)، (۷) و (۸) زمان ورود و خروج ماشین‌های دریافت را بر اساس ترتیب شان در توالی ماشین‌ها مشخص می‌کند. محدودیت (۹) تضمین می‌کند که هیچ کامیون ورودی از خودش در ترتیب توالی‌ها پیشی نگیرد. محدودیت‌های (۱۰)، (۱۱) و (۱۲) زمان ورود و خروج ماشین‌های ارسال را بر اساس ترتیب‌شان در توالی ماشین‌ها مشخص می‌کند. محدودیت (۱۳) تضمین می‌کند که هیچ کامیون خروجی از خودش در ترتیب توالی‌ها پیشی نگیرد. محدودیت (۱۴) بین زمان خروج برای هر کامیون خروجی و زمان ورود برای هر ماشین ورودی برای هر نوع کالا، ارتباط برقرار می‌کند.

۵ روش‌های فراابتکاری برای حل مدل ریاضی

- **نمایش جواب**

نحوه نمایش جواب نقش حیاتی در عملکرد الگوریتم‌ها برای پیدا نمودن نقاط مختلف فضای جواب‌های مساله دارد. همچنین تغییر و منطبق کردن عملگرهای هر الگوریتم با این ساختار جواب جز اجتناب‌ناپذیر استفاده از الگوریتم می‌باشد [۲۷]. روش کدگذاری در شکل ۱ نشان داده شده است. ابعاد کروموزوم با مجموع تعداد کامیون‌های دریافت و ارسال می‌باشد. در شکل ۱ فرض بر این است که چهار کامیون ورودی و چهار کامیون خروجی وجود دارد. ساختار کروموزم متشکل از دو قسمت است. قسمت اول تعیین‌کننده‌ی توالی کامیون‌های دریافت، و قسمت دوم تعیین‌کننده‌ی توالی کامیون‌های ارسال می‌باشد. این طرح کدگذاری در تمام الگوریتم‌های فراابتکاری استفاده شده در این مطالعه به‌کار رفته است.

[Downloaded from jorlu.ir on 2026-11-30]

[DOI: 10.52547/jamlu.19.4.37]



شکل ۱. مثالی از طرح کدگذاری

در این طرح، از روش کلید تصادفی[1] برای ایجاد یک جواب شدنی استفاده شده است.

- **الگوریتم ژنتیک**

نخستین بار هلند [2] [۲۸] ایده‌ی استفاده از الگوریتم ژنتیک را برای مسایل بهینه‌سازی مطرح نمود. سپس این الگوریتم به طور گسترده در طیف وسیعی از مسایل بهینه‌سازی مورد استفاده قرار گرفت. الگوریتم ژنتیک، الهامی از علم ژنتیک و نظریه‌ی تکامل داروین است و بر اساس بقای برترین‌ها یا انتخاب طبیعی استوار است. GA یک الگوریتم جمعیت محور[3] 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 به‌صورت تصادفی تولید می‌شود. سپس میزان برازندگی هر یک از اعضای جمعیت (کروموزوم) تعیین می‌شود. در مسایل بهینه‌سازی، معمولا 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[4] و جهش است. شبه کد الگوریتم الگوریتم ژنتیک به طور خلاصه در شکل ۲ نشان داده شده است.

۱	جمعیت اولیه را به صورت تصادفی ایجاد کنید
۲	محاسبه کیفیت پاسخ ها با استفاده از تابع برازندگی
۳	انتخاب والدها براساس کیفیت آنها
۴	عملیات عملگرهای الگوریتم ژنتیک، لقاح و جهش
۵	جایگزینی نسل جدید با نسل قدیم
۶	بررسی معیارهای توقف الگوریتم ژنتیک
۷	انتخاب بهترین کروموزوم به عنوان پاسخ نهایی

شکل ۲. شبه کد الگوریتم ژنتیک

- **الگوریتم تبرید شبیه‌سازی‌شده[5]**

به کارگیری الگوی تبرید شبیه‌سازی شده برای مسایل بهینه‌سازی اولین بار توسط کریک پاتریک و همکاران [6] [۲۹] انجام شد. به منظور حل مسایل بهینه‌سازی توسط این الگوریتم، ابتدا یک راه‌حل اولیه به صورت تصادفی تولید می‌گردد و مطابق با تابع هدف مورد ارزیابی قرار می‌گیرد. شایان ذکر است که در ابتدا این جواب اولیه به عنوان بهترین جواب‌یافته‌شده در نظر گرفته می‌شود. سپس یک همسایگی در اطراف آن تولید و مورد ارزیابی قرار می‌گیرد. در این صورت یکی از سه حالت زیر اتفاق می‌افتد:

[1] random-key
[2] Holland
[3] Population-Base
[4] Crossover
[5] Simulated Annealing (SA)
[6] Kirkpatrick et al.

[Downloaded from jorlu.ir on 2026-11-30]
[DOI: 10.52547/jamlu.19.4.37]

۱. جواب همسایه بهتر از جواب فعلی است. در این حالت جواب همسایه جایگزین آن می‌گردد.
۲. جواب همسایه بدتر از جواب فعلی است. در این حالت سیستم مجاز به پذیرش جواب همسایه با یک احتمالی است. مقدار این احتمال از رابطه‌ی زیر به‌دست می‌آید:

$$p = e^{\frac{\Delta f}{T}} \qquad (۱۶)$$

که در آن **T** یک پارامتر به نام دماست. در هر دما سیستم مجاز به کاوش در فضای جستجوی مساله در تعداد تکرار مشخصی می‌باشد. پس از آن دما به آرامی کاهش می‌یابد. **Δf** تفاوت بین مقدار تابع هدف راه‌حل فعلی و راه‌حل همسایه است که مقدار آن از رابطه‌ی زیر به‌دست می‌آید:

$$\Delta f = f(x^{new}) - f(x) \qquad (۱۷)$$

در رابطه بالا، **f(xnew)** برابر با مقدار تابع هدف راه‌حل جدید و **f(x)** برابر با مقدار تابع هدف راه‌حل فعلی است.

در صورتی که جواب همسایه مطابق با حالت ۱ یا ۲ مورد پذیرش واقع نشود حذف می‌گردد و یک همسایگی جدید بر روی جواب فعلی ایجاد می‌گردد. در تکرارهای اول مقدار دما بالا در نظر گرفته می‌شود تا شانس پذیرش جواب‌های بدتر بیشتر باشد. سپس در هر تکرار دما به آرامی کاهش می‌یابد و در نهایت الگوریتم به سمت یک جواب خوب همگرا می‌شود.در شکل ۳ شبه کد الگوریتم تبرید شبیه‌سازی شده نشان داده شده است.

۱	راه حل اولیه **S** را به صورت تصادفی ایجاد کنید
۲	دمای اولیه **T** را تعیین کنید
۳	تابع هزینه را برای این راه حل $f(S)$ ارزیابی کنید
۴	تا زمانی که شرایط توقف محقق نشده مراحل زیر را تکرار کنید
۵	برای **j=1** تا حداکثر تکرار داخلی مراحل زیر را تکرار کنید
۵-۱	بر اساس ساختار همسایگی راه حل جدید S' را تولید نمایید
۵-۲	تابع هزینه را برای این راه حل جدید $f(S')$ ارزیابی کنید
۵-۳	اگر $f(S') \le f(S)$ آنگاه $S = S'$
۵-۴	در غیر اینصورت اگر عدد تصادفی **r** از $p = e^{\frac{\Delta f}{T}}$ کوچکتر بود آنگاه $S = S'$
۵-۵	پایان تکرار حلقه داخلی
۶	دمای اولیه **T** را کاهش دهید
۷	در صورت تحقق شرایط توقف **S** را به عنوان جواب نهایی به خروجی بفرستید

شکل ۳. شبه کد الگوریتم تبرید شبیه‌سازی‌شده

- **الگوریتم رقابت استعماری**[1]

اکثر الگوریتم‌های بهینه‌سازی مانند الگوریتم ژنتیک، به طور عمده الهام گرفته از فرایندهای طبیعی می‌باشند و در ارایه این الگوریتم‌ها به سایر نمودهای تکامل انسانی توجهی نشده است. الگوریتم رقابت استعماری، الگوریتم جدیدی برای بهینه‌سازی است که نه از یک پدیده طبیعی، بلکه از یک پدیده اجتماعی انسانی الهام گرفته است.

[1] Imperialist Competitive Algorithm (ICA)

[Downloaded from jorlu.ir on 2026-11-30]
[DOI: 10.52547/jamlu.19.4.37]

به طور ویژه، این الگوریتم به فرایند استعمار به عنوان مرحله‌ای از تکامل اجتماعی سیاسی بشر نگریسته و با مدل‌سازی ریاضی این پدیده تاریخی از آن به عنوان منشا الهام یک الگوریتم قدرتمند در زمینه بهینه‌سازی بهره می‌گیرد [۳۰].

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 بهینه‌سازی ارایه نموده است. همانند همه الگوریتم‌های قرار گرفته در دسته الگوریتم های بهینه سازی تکاملی، الگوریتم رقابت استعماری نیز مجموعه اولیه‌ای 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 گروه بندی 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 مساله بهینه‌سازی (کشور مطلوب) را در اختیار می‌گذارد. مراحل الگوریتم ICA را می‌توان به صورت شبه کد زیر خلاصه کرد:

۱	مقداردهی اولیه الگوریتم. چند راه حل تصادفی در فضای جستجو ایجاد کنید و امپراتوری های اولیه ایجاد کنید.
۲	همگون‌سازی: مستعمرات در جهت‌های مختلف به سمت امپریالیست‌ها حرکت می‌کنند.
۳	انقلاب: تغییرات تصادفی در ویژگی‌های برخی کشورها رخ می‌دهد
۴	تغییر موضع بین مستعمره و امپریالیست. مستعمره‌ای که موقعیت بهتری نسبت به امپریالیست دارد، این شانس را دارد که با جایگزینی با امپریالیست موجود، کنترل امپراتوری را در دست بگیرد.
۵	رقابت امپریالیستی: همه امپریالیست‌ها برای تصاحب مستعمرات یکدیگر رقابت می‌کنند.
۶	امپراتوری‌های بی‌قدرت را از بین ببرید. امپراتوری‌های ضعیف به تدریج قدرت خود را از دست می‌دهند و در نهایت حذف خواهند شد.
۷	اگر شرط توقف برآورده شد، توقف کنید، اگر نه به گام ۲ بروید.
۸	پایان

شکل ٤. شبه کد الگوریتم ICA

یکی از مهم‌ترین ابزارهای الگوریتم ICA این است که کشورهای امپریالیستی مستعمرات خود با استفاده از سیاست جذب نشان داده شده در شکل ۱را جذب کند. برای به‌روز رسانی موقعیت خود، هر مستعمره حرکت به سوی امپریالیست خود را با استفاده از فرمول (۱۸) انجام می‌دهد:

$$x_{t+1} = x_t + \beta d . \gamma U(-\theta, \theta) \qquad (۱۸)$$

که در آن $\beta > ۱$ باعث می‌شود مستعمرات به امپریالیست نزدیک‌تر شوند. d فاصله بین مستعمره و امپریالیست است. γ ضریب جذب است $(\gamma < ۱)$ و θ انحراف از جهت اصلی، که قابلیت جستجو در اطراف امپریالیست را تنظیم می‌کند.

- **الگوریتم ترکیبی PCICA**[1]

[1] PRINCIPAL COMPONENT IMPERIALIST COMPETETIVE ALGORITHM

[Downloaded from jorlu.ir on 2026-11-30]

[DOI: 10.52547/jamlu.19.4.37]

در این مقاله، یک الگوریتم ترکیبی پیشنهادی جدید برای اولین بار به نام PCICA، که برای افزایش توانایی اکتشاف ICA طراحی شده است، ارایه می‌شود. این الگوریتم ترکیبی از الگوریتم تکاملی رقابت استعماری (ICA) و روش تجزیه و تحلیل مولفه اصلی (PCA) است. در الگوریتم رقابت استعماری (ICA) سیاست همگون‌سازی (جذب)، با استفاده از یک استراتژی یادگیری خطی، موقعیت هر کشور را از طریق یک ترکیب خطی با استفاده از موقعیت قبلی کشور و فاصله از موقعیت امپریالیست آن با کمی زاویه تصادفی به‌روزرسانی می‌کند. علیرغم سادگی این استراتژی در بروزرسانی موقعیت، هنگام جستجو در یک فضای چند بعدی ناکارآمد بوده و دارای محدودیت در اکتشاف اطلاعات است. بنابراین، در الگوریتم ترکیبی پیشنهادی در استفاده از ICA ،یک مکانیسم جستجو جدید پیشنهاد شده است که بر اساس تجزیه و تحلیل مولفه اصلی (PCA) است که در آن از PCA که عمدتا به نحو احسن اطلاعات جمعیت را برای تعیین جهت مولفه‌های اصلی استخراج می‌کند، استفاده می‌شود. PCICA می‌توانید بسیاری از اطلاعات موقعیت کشورها را ذخیره‌سازی کند که با استفاده از آن کشورها به جهت‌های امیدوار کننده‌تری هدایت شوند. گام‌های الگوریتم پیشنهادی در زیر نشان داده شده است.

گام ۱: مقداردهی اولیه

(الف) مقداردهی اولیه امپراتوری‌ها

مانند تمام الگوریتم‌های تکاملی، ICA نیز با یک جمعیت اولیه از راه حل که کشورها نامیده می‌شوند، با اندازه $\boldsymbol{N_{pop}}$شروع می‌شود. از این کشورها تعداد $\boldsymbol{N_{imp}}$ از بهترین راه‌حل‌ها به عنوان امپریالیست انتخاب می‌شوند و تعداد $\boldsymbol{N_{col}}$ از کشورهای باقی‌مانده تشکیل مستعمرات این امپریالیست‌ها را می‌دهند. امپراطوری‌های اولیه با تقسیم مستعمرات در میان استعمارگران با توجه به قدرت نرمالایز شده آنها شکل می‌گیرند:

$$P_n = \left| \frac{C_n}{\sum_{i=1}^{N_{imp}} C_i} \right| \qquad (۱۹)$$

که در آن $C_n = c_n - \max c_i$. $\boldsymbol{c_n}$ هزینه $\mathbf{n}$ امین امپریالیست است و $\boldsymbol{C_n}$ هزینه نرمالیزه شده آن است. تعداد مستعمرات $\boldsymbol{N_{Cn}}$ یک امپراتوری با توجه به فرمول زیر محاسبه می‌شود:

$$NC_n = round\,(P_n . N_{col}) \qquad (۲۰)$$

ب) مقداردهی اولیه ماتریس کوواریانس

[Downloaded from jorlu.ir on 2026-11-30]

[DOI: 10.52547/jamlu.19.4.37]

$$S = \begin{bmatrix} S_1^2 & S_{12} & \dots & S_{1n} \\ S_{12} & S_2^2 & \dots & S_{2n} \\ \vdots & \vdots & & \vdots \\ S_{1n} & S_{2n} & \dots & S_n^2 \end{bmatrix} \quad (۲۱)$$

$$S_{jk} = \frac{\sum_{i=1}^{N} W_i (x_{ji} - \bar{x}_j)(x_{ki} - \bar{x}_k)}{n-1}$$

که در آن **n** نشان‌دهنده تعداد ابعاد است.

ج) محاسبه مولفه‌های اصلی

$$u_i' S u = L \quad (۲۲)$$

د) انتقال کشورها به فضای **Z**

مولفه های اصلی سپس به صورت زیر تعریف می شود:

$$z = U' \left[x - \bar{x} \right] \quad (۲۳)$$

متغیرهای فضای z_i (مولفه‌های اصلی) هستند که توسط یک ترکیب خطی از متغیرهای فضای **X** تشکیل شده‌اند که متوسط آن صفر و واریانس آنها **li** است.

$$z_{i\in\{1\dots n\}} = u_i'[x - \bar{x}] \quad (۲۴)$$

عکس این عملیات نیز امکانپذیر می‌باشد. هر ذره در فضا **Z** را می‌توان با استفاده از معادله زیر به فضای **X** بازگرداند:

$$x = \bar{x} + Uz \quad (۲۵)$$

فضای کاهش بعد یافته ~**z** (که نشان‌دهنده کاهش ابعاد است) را می‌توان به صورت زیر تعریف کرد:

$$\tilde{z} = \tilde{U}'[x - \tilde{x}] \quad (۲۶)$$

که در آن $\tilde{U}$ با استفاده از بردارهای ویژه مرتبط با بزرگ‌ترین مقادیر ویژه تعریف شده است. این رابطه را می‌تواند برای هر متغیر $\tilde{z}_i$ نوشته شود:

$$\tilde{z}_{i\in\{1\dots n\}} = \tilde{U}_j'[x - \tilde{x}] \quad (۲۷)$$

همین‌طور که پیشتر نیز گفته شد فضا **Z** را می‌توان با استفاده از معادله زیر به فضای **X** بازگرداند:

$$x = \bar{x} + \tilde{U}\tilde{z} + (x - \tilde{x}) \quad (۲۸)$$

اولین مولفه اساسی استخراج‌شده بیشترین مقدار پراکندگی داده‌ها را در کل مجموعه داده‌ها در نظر می‌گیرد. این امر بدان معنی است که اولین مولفه حداقل با تعدادی از متغیرها همبسته است. دومین مولفه استخراج‌شده دو ویژگی مهم دارد، این مولفه بیشترین واریانس مجموعه داده‌ها که توسط مولفه اول محاسبه نشده است را در نظر می‌گیرد. یعنی دومین مولفه با تعدادی از متغیرهای مشاهده شده که همبستگی بالایی با جزء اول ندارند،

[Downloaded from jorlu.ir on 2026-11-30]

[DOI: 10.52547/jamlu.19.4.37]

همبسته است. ویژگی دوم این است که مولفه دوم با مولفه اول همبستگی ندارد، یعنی همبستگی بین دو مولفه صفر است. سایر مولفه‌های استخراج‌شده در این روش نیز دو ویژگی مذکور را دارا می‌باشند.

ه) محاسبه قدرت تمامی کشورها در فضای جدید

و) جابجایی کلونی و امپریالیست در صورتی که قدرت کلونی از امپریالیست بیشتر شود.

۲. حرکت کشورها (سیاست جذب)

الف) حرکت کشورها در فضا $\mathbf{Z}$

موقعیت کشورها در فضای $\mathbf{Z}$ است با توجه به فرمول زیر به‌روز رسانی می‌شود:

$$\tilde{z}_{i,j}^{k+1} = \tilde{z}_{i,j}^{k} + \tilde{\beta}.(\tilde{z}_{i,g}^{k} - \tilde{z}_{i,j}^{k}).\tilde{\gamma}.\tilde{U}(-\theta,\theta) \quad (۲۹)$$

در این فرمول $\boldsymbol{i \le m \le n}$ شاخص بعد $\tilde{\mathbf{z}}$ است، $\mathbf{j}$ نشان‌دهنده شاخص کشور است، $\mathbf{g}$ تعیین‌کننده امپریالیست است، $\mathbf{U}$ عدد تصادفی یکنواخت بین $-\theta$ و θ است. $\tilde{z}_{i,j}^{k}$ نشان‌دهنده $\mathbf{i}$ ام بعد تغییر یافته امپریالیست در طول تکرار $\mathbf{K}$ است.

ب) انتقال کشورها از فضای جدید $\mathbf{Z}$ به فضای $(\mathbf{X})$

مکان فضای به‌روز شده $\mathbf{z}$ با استفاده از بردار متعامد جزیی $\tilde{U}$ به فضا $\mathbf{x}$ قابل بازگشت است، که در آن $\tilde{\mathbf{U}}$ شامل اولین $\mathbf{m}$ بردارهای ویژه مربوط به بزرگ‌ترین مقادیر ویژه است. هنگامی که $\mathbf{m}$ برابر با ابعاد فضای $\mathbf{X}$ باشد یعنی $\mathbf{M = n}$ در نتیجه $\tilde{U} = U$ خواهد بود.

$$\tilde{x}_{i,j}^{k+1} = \tilde{\tilde{x}}_{i,j}^{k} + \tilde{U}[\tilde{z}_{i,j}^{k+1}] \quad (۳۰)$$

۳. انقلاب

در این قسمت تغییرات تصادفی در ویژگی‌های برخی از کشورها رخ می‌دهد. در این جابجایی به صورت ناگهانی یک کشور بدون تبعیت از قانون و قاعده خاصی که تاکنون تعریف‌شده، موقعیت خود را تغییر می‌دهد که به این عمل انقلاب می‌گویند. این عمل می‌تواند در جهت بهتر یا بدتر کردن یک یا چند عامل صورت گیرد.

۴. جایگزینی

بعضی از کشورها در حین حرکت به سمت کشور امپریالیست ممکن است به موقعیت بهتری نسبت به امپریالیست دست یابند. یک کلونی با موقعیت بهتری نسبت به امپریالیست این شانس را دارد که کنترل امپراتوری را با گرفتن جایگاه امپریالیست به‌دست آورد.

۵. قدرت کل یک امپراطوری

کل قدرت هر امپراطوری توسط هزینه‌ی امپریالیست آن به علاوه هزینه متوسط مستعمرات آن تعریف شده است، به این ترتیب قدرت کل یک امپراطوری از طریق فرمول زیر قابل دستیابی است:

$$TC_n = \cos t(imperialist_n) + \varepsilon \times mean(coloniesofempires_n) \quad (۳۱)$$

که در آن $\boldsymbol{\varepsilon}$ یک عدد مثبت کمتر از ۱ در نظر گرفته می‌شود.

۶. رقابت امپریالیستی

[Downloaded from jorlu.ir on 2026-11-30]

[DOI: 10.52547/jamlu.19.4.37]

تمام امپراتوری‌ها سعی در به اختیار گرفتن مستعمرات دیگر امپراتوری ها و کنترل آنها را دارند. هر امپراطوری که نتواند بر قدرت خود بیافزاید و قدرت رقابت خود را از دست بدهد در جریان رقابت امپریالیستی حذف خواهد شد. این رقابت امپریالیستی با انتخاب ضعیف ترین مستعمره از ضعیف‌ترین امپراتوری و به انتقال آن (یا آنها) به امپراتوری که دارای بالاترین احتمال برای تصاحب آن (یا آنها) است، مدل شده است. مستعمرات مذکور لزوما توسط قوی ترین امپراطوری تصاحب نخواهند شد بلکه امپراطوری‌های قوی تر، احتمال تصاحب بیشتری دارند.

برای شروع رقابت $\boldsymbol{P_{pn}}$ به عنوان احتمال تصاحب هر امپراتوری تعریف می‌شود:

(۳۲)

$$P_{Pn} = \left| \frac{NTC_n}{\sum_{i=1}^{N_{imp}} NTC_i} \right|$$

که در آن $NTC_n = TC_n - \max_i TC_i$. TC_n و NTC_n به ترتیب هزینه کل و هزینه نرمال کل امپراتوری $\mathbf{n}$ ام هستند.

برای اختصاص این مستعمره به یک امپراطوری بر اساس احتمال تصاحب، یک بردار $\mathbf{P}$ تشکیل می‌دهیم:

$$P = [P_{p1}; P_{p2}; P_{p3}; ...; P_{pNimp}] \quad (۳۳)$$

سپس یک بردار با همان اندازه بردار $\mathbf{P}$ که از اعداد تصادفی تشکیل شده است ایجاد می کنیم:

$$R = [R_1; R_2; R_3; ...; R_{Nimp}] \quad (۳۴)$$

که در آن $\mathbf{r_i}$ اعداد تصادفی بین ۰ و ۱ است. در نهایت، بردار $\mathbf{A}$ که حاصل از کم کردن $\mathbf{R}$ از $\mathbf{P}$ است را تشکیل می‌دهیم.

$$A = [P_{p1} - r_1; P_{p2} - r_2; P_{p3} - r_3; ...; P_{pNimp} - r_{Nimp}] \quad (۳۵)$$

با توجه به بردار $\mathbf{A}$، مستعمره اشاره شده به امپراطوری که شاخص مربوط به آن حداکثر است اختصاص داده می‌شود.

۷. از بین بردن امپراطوری‌های ضعیف

۸. اگر شرط توقف برقرار است توقف کنید، در غیر این‌صورت به گام ۲ بروید.

۹. پایان

٦ نتایج محاسباتی

در این قسمت ابتدا به طراحی مساله‌هایی جهت بررسی عملکرد الگوریتم‌های ارایه‌شده می‌پردازیم. سپس پارامترهای هر الگوریتم تعیین و تنظیم می‌گردد. تمامی مسایل توسط الگوریتم‌ها اجرا، و نتایج به‌دست 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 از ۱۵ مساله در ابعاد متوسط و بزرگ که توسط گلشاهی رودبنه و دیگران [۲۲] و سیدی

[Downloaded from jorlu.ir on 2026-11-30]

[DOI: 10.52547/jamlu.19.4.37]

و دیگران [۳۱,۳۲] استفاده گردید، استفاده شده است. مشخصات مربوط به آن‌ها در مقاله گلشاهی رودبنه و دیگران [۲۲] قابل دسترسی می‌باشد.

- **تنظیم پارامتر**

هر کدام از روش‌های فراابتکاری به پارامترهایی برای انجام فرایند مخصوص خود نیاز دارند تا نتایج به‌دست آمده از روش‌های فراابتکاری مورد نظر را مورد بررسی قرار دهد. به عنوان مثال در الگوریتم ***GA***، پارامترهای تعداد تکرار[۱](***Maxit***)، تعداد جمعیت[۲](***Npop***)، نرخ تقاطع[۳](***Pc***)، نرخ جهش[۴](***Pm***)، در الگوریتم رقابت استعماری، تعداد تکرار، تعداد جمعیت یا تعداد کشور، تعداد امپراطور[۵](***Nimp***)، بتا[۶]، نرخ انقلاب[۷] و زتا[۸]. تمامی الگوریتم‌های فراابتکاری معرفی‌شده در قسمت قبل دارای پارامترهایی می باشند 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.

به دلیل تصادفی بودن ماهیت الگوریتم‌های فراابتکاری، این امکان وجود دارد که یک الگوریتم در یک اجرا جواب خوبی را به‌دست بیاورد، اما همان الگوریتم در یک اجرای دیگر جوابی با کیفیت نامطلوب را ارایه دهد. 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 مسایل آزمایشی متفاوت است، ما نمی‌توانیم نتیجه به‌دست آمده را به طور مستقیم اعمال کنیم. برای از بین بردن این مشکل، با استفاده از RPD[۹] بر روی مقادیری که از طریق آزمایشات برای هر مساله به‌دست می‌آید، بی‌مقیاس‌سازی صورت می‌گیرد. مقدار ***RPD*** با استفاده از رابطه‌ی زیر به‌دست می‌آید:

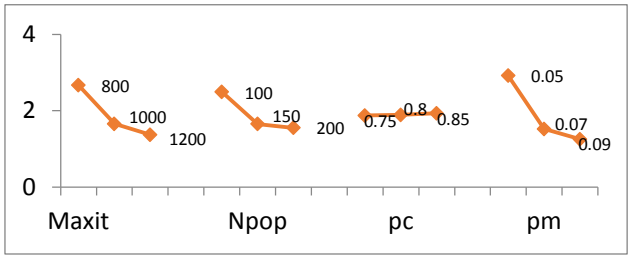
$$RPD = \frac{Sol_{ij}^{k} - Bestsol^{k}}{Bestsol^{k}} \qquad (۳۶)$$

در رابطه‌ی بالا Sol_{ij}^{k}، مقدار به‌دست آمده برای مسأله‌ی Kام در تکرار jام از آزمایش i است. $Bestsol^{k}$ بهترین مقدار به‌دست آمده برای مسأله ی kام است. شکل‌های ۷، ۸، ۹، ۱۰ و ۱۱ مقادیر RPD به‌دست آمده برای پارامترهای مختلف را نشان می‌دهند.

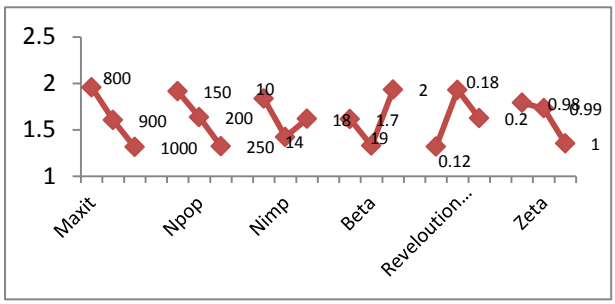
[1] Number of Maxiteration(Maxit)
[2] Number of population(Npop)
[3] Crossover rate(Pc R)
[4] Mutation rate(Pm R)
[5] Number of imperialist(Nimp)
[6] Beta
[7] Revolution rate
[8] Zeta
[9] Relative Percentage Deviation

[Downloaded from jorlu.ir on 2026-11-30]

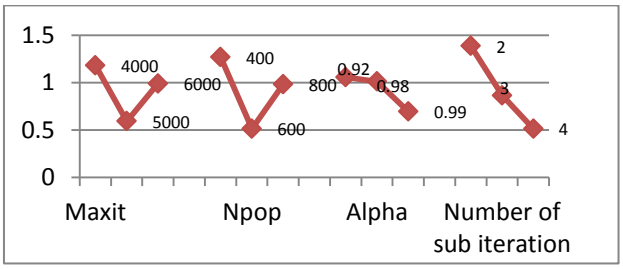
[DOI: 10.52547/jamlu.19.4.37]



شکل ۵. نمودار RPD پارامترهای الگوریتم GA



شکل ٦. نمودار RPD برای پارامترهای الگوریتم ICA



شکل ۷. نمودار RPD برای پارامترهای الگوریتم SA

میانگین RPD یک پاسخ متوسط برای ترکیب‌های مختلف از سطوح فاکتورهای کنترل در یک طرح تاگوچی می‌باشد. با توجه به هدف این تحقیق که مینیمم‌سازی است، در صدد تعیین سطح عاملی هستیم که میانگین را به حداقل برساند. به عنوان مثال در شکل ۵ 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 Makespan 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
GA	Max it	۱۲۰۰
	N pop	۲۰۰
	Pc	۰/۷۵
	Pm	۰/۰۹

[Downloaded from jorlu.ir on 2026-11-30]

[DOI: 10.52547/jamlu.19.4.37]

SA	Max it	۵۰۰۰
	T0	۶۰۰
	Alpha	۰/۹۹
	Number of sub iteration	۴
ICA	Max it	۱۰۰۰
	N pop	۲۵۰
	N imp	۱۴
	Beta	۱/۹
	Revolution rate(R)	۰/۱۲
	Zeta	۱

- **نتایج الگوریتم‌های فراابتکاری**

برای بررسی عملکرد الگوریتم های فراابتکاری از ۱۵ مساله در ابعاد متوسط و بزرگ که توسط گلشاهی رودبنه و دیگران (۲۰۱۷) طراحی گردید، استفاده شده است. در هر اجرا نتایج ثبت گردید. بهترین، بدترین و میانگین نتایج به‌دست آمده توسط هر الگوریتم در جدول ۲ نشان داده شده است.

جدول۲. نتایج بدست آمده توسط الگوریتم‌های فراابتکاری

مسأله	PCICA			ICA			SA			GA		
	میانگین	بهترین	بدترین	میانگین	بهترین	بدترین	میانگین	بهترین	بدترین	میانگین	بهترین	بدترین
۱	۳۱۳۴/۵	۳۰۸۵	۳۲۳۳	۳۱۷۶/۴	۳۰۸۵	۳۲۱۹	۳۱۷۳/۱	۳۰۸۱	۳۲۶۵	۳۱۹۶/۱	۳۱۰۹	۳۲۶۰
۲	۳۵۳۵/۵	۳۵۲۲	۳۵۶۷	۳۵۵۳/۲	۳۵۲۶	۳۵۶۳	۳۵۷۰/۶	۳۵۱۲	۳۵۹۷	۳۵۷۳/۳	۳۵۶۱	۳۵۹۵
۳	۵۱۴۶/۸	۵۰۵۶	۵۲۷۸	۵۱۹۰/۳	۵۰۵۶	۵۲۴۹	۵۲۰۳/۱	۵۰۵۶	۵۸۳۸	۵۲۷۲/۷	۵۰۵۶	۵۳۵۳
۴	۳۹۶۶/۴	۳۸۷۷	۴۰۲۸	۳۹۹۴/۸	۳۸۷۹	۴۰۲۷	۳۹۶۱/۲	۳۸۶۴	۴۰۱۹	۳۹۷۲/۷	۳۸۸۴	۴۰۹۸
۵	۵۲۵۷/۱	۵۱۸۴	۵۳۶۹	۵۲۹۴/۸	۵۱۸۴	۵۳۶۶	۵۲۴۰/۳	۵۱۷۸	۵۳۵۵	۵۳۰۶/۹	۵۲۰۳	۵۴۱۹
۶	۷۹۹۳/۵	۷۸۷۲	۸۱۴۱	۷۹۹۰/۲	۷۸۷۷	۸۱۲۸	۷۹۵۰/۸	۷۸۰۳	۸۰۹۷	۷۹۹۹/۳	۷۸۵۴	۸۱۳۰
۷	۷۱۵۵/۱	۶۹۴۸	۷۲۶۷	۷۱۵۵/۱	۷۰۱۲	۷۲۳۷	۷۱۶۳/۶	۷۰۹۱	۷۳۳۸	۷۱۵۷/۵	۷۰۰۳	۷۲۴۱
۸	۷۷۲۰/۹	۷۵۸۷	۷۸۶۰	۷۷۰۴/۸	۷۶۰۰	۷۸۴۰	۷۸۰۹	۷۶۶۲	۷۹۶۶	۷۷۲۵/۶	۷۵۹۶	۷۹۳۵
۹	۶۲۹۱/۶	۶۱۲۴	۶۳۷۰	۶۲۸۳/۶	۶۱۹۷	۶۳۶۳	۶۳۱۹/۸	۶۲۴۹	۶۳۹۵	۶۳۰۵/۸	۶۱۹۹	۶۴۱۰
۱۰	۵۶۰۴/۲	۵۴۸۹	۵۶۹۵	۵۶۰۱/۹	۵۵۰۹	۵۶۹۰	۵۵۳۵/۳	۵۴۹۰	۵۶۷۵	۵۷۳۱/۸	۵۶۳۲	۵۸۲۹
۱۱	۸۵۷۴/۶	۸۳۹۷	۸۷۰۷	۸۵۸۳/۷	۸۴۱۴	۸۷۰۷	۸۴۴۴/۵	۸۳۲۹	۸۷۱۳	۸۷۶۹/۹	۸۴۷۸	۸۹۳۴
۱۲	۸۳۱۰/۵	۸۱۵۴	۸۵۴۵	۸۳۸۶/۱	۸۲۳۳	۸۵۳۹	۸۳۱۸/۱	۸۱۹۶	۸۵۲۸	۸۵۱۰/۹	۸۳۰۲	۸۷۲۵
۱۳	۹۴۵۲/۶	۹۳۰۶	۹۶۶۱	۹۴۸۸/۳	۹۳۴۹	۹۶۵۶	۹۴۱۲/۹	۹۳۳۹	۹۶۹۶	۹۶۴۸/۱	۹۴۷۵	۹۹۲۸
۱۴	۱۱۰۷۴/۷	۱۰۶۹۴	۱۱۳۴۸	۱۱۰۸۶/۵	۱۰۹۰۲	۱۱۳۷۰	۱۱۰۵۳/۷	۱۰۸۹۸	۱۱۲۸۴	۱۱۲۱۱/۸	۱۰۹۵۶	۱۱۶۹۶
۱۵	۹۵۵۴/۹	۹۳۳۶	۹۷۰۱	۹۵۵۴	۹۴۴۴	۹۷۰۲	۹۴۶۶/۶	۹۴۲۳	۹۷۰۷	۹۶۷۱/۹	۹۵۵۷	۹۸۲۴

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 دو معیار عملکرد انحراف درصد نسبی (RPD) و زمان پردازش

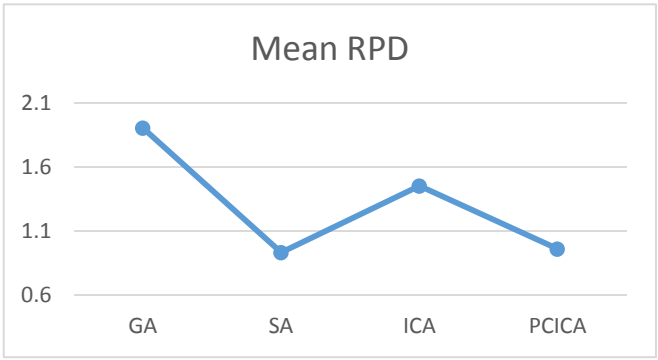
[Downloaded from jorlu.ir on 2026-11-30]

[DOI: 10.52547/jamlu.19.4.37]

استفاده شده است. میانگین مقدار RPD برای جواب‌های به‌دست آمده توسط الگوریتم‌ها در هر مساله در سی بار اجرای الگوریتم در جدول ۳ نشان داده شده است.

جدول ۳. عملکرد انحراف درصد نسبی(RPD)

مسأله	GA	SA	ICA	PCICA
۱	۲/۱۱۱۶۶	۰/۸۹۹۱۳	۱/۶۳۲۷	۰/۹۴۴
۲	۱/۵۲۰۱۸	۰/۹۵۱۲۸	۱/۴۸۱	۰/۹۵۲
۳	۱/۴۱۳۸۱	۰/۹۵۸۹۹	۱/۸۲۰۵	۰/۹۸۷
۴	۲۰۳۲۶۳۶	۰/۹۲۸۵۵	۱/۴۳۹۵	۱/۲۹۵
۵	۱/۹۹۰۵۱	۰/۹۲۵۴۹	۰/۹۷۹۹	۱/۲۰۸
۶	۱/۴۳۰۱۵	۰/۸۹۵۹۶	۱/۴۴۰۱	۰/۸۳۰
۷	۱/۹۶۸۸۹	۰/۸۸۴۸۶	۱/۲۰۶۴	۰/۹۱۶
۸	۱/۹۰۷۳۳	۰/۹۷۹۸۵	۱/۱۴۴۴	۰/۸۷۰
۹	۱/۴۵۷۵۱	۰/۹۲۰۸۳	۱/۲۸۳	۰/۸۷۶
۱۰	۲/۱۸۵۲۳	۱/۶۶۸۸	۱/۷۷۰۳	۰/۸۷۹
۱۱	۲/۳۰۹۱	۱/۴۵۵۷	۰/۹۸۰۱	۰/۸۷۱
۱۲	۲/۳۵۴۶۷	۱/۰۹۳۶	۱/۶۲۱۸	۰/۹۱۲
۱۳	۱/۷۳۳۴۵	۱/۱۵۴۹	۱/۷۱۸۷	۰/۹۶۷
۱۴	۱/۹۴۳۲۸	۱/۰۰۶	۱/۵۳۰۴	۰/۹۱۹
۱۵	۱/۹۰۳۵۳	۰/۹۵۸۵۱	۱/۷۲۶۸	۰/۷۳۶
میانگین	۱/۹۰۳۷	۰/۸۹۹۱۳	۱/۶۳۲۷	۰/۹۴۴



شکل ۸. معیار عملکرد انحراف درصد نسبی(RPD)

RPD با استفاده از رابطه (۳۶) محاسبه می‌شود. همچنین جهت بررسی زمان پردازش میانگین زمان صرف‌شده جهت یافتن بهترین جواب‌ها در سی بار اجرای الگوریتم محاسبه شده است. این نتایج در جدول ۴ نشان داده شده است.

میانگین RPD در شکل ۸ نشان داده شده است. همان‌طور که از این شکل نیز مشخص است در این معیار الگوریتم PCICA بهترین عملکرد را به خود اختصاص داده است. از لحاظ زمان پردازش نیز الگوریتم SA عملکرد مطلوب‌تری داشته است و دارای کمترین زمان پردازش می‌باشد.

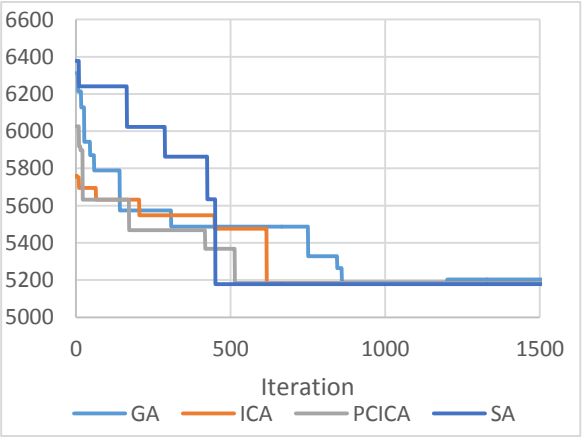
[Downloaded from jorlu.ir on 2026-11-30]

[DOI: 10.52547/jamlu.19.4.37]

جدول ٤. میانگین زمان پردازش

مسأله	GA	SA	ICA	PCICA
۱	۱۸/۲۷	۱/۹۱	۲۵/۰۵	۲۷/۵۶
۲	۱۹/۹۲	۱/۳۸	۲۴/۱۵	۲۵/۶۰
۳	۱۵/۰۴	۲/۵۵	۳۴/۵۹	۳۸/۰۵
۴	۳۱/۶۵	۵/۴۶	۷۳/۷۵	۷۹/۶۴
۵	۲۴/۴۰	۴/۸۳	۸۲/۴۲	۹۰/۶۶
۶	۳۴/۳۷	۹/۹۹	۷۰/۷۷	۷۲/۹۰
۷	۶۱/۴۰	۹/۷۵	۹۸/۴۴	۱۰۰/۴۰
۸	۵۷/۴۸	۸/۶۰	۱۱۹/۷۳	۱۲۶/۹۲
۹	۶۵/۶۱	۱۰/۳۳	۱۱۳/۴۳	۱۲۰/۲۴
۱۰	۱۱۵/۰۵	۱۶/۳۵	۱۳۲/۴۰	۱۴۴/۳۲
۱۱	۱۱۶/۷۶	۲۱/۸۵	۱۷۳/۸۷	۱۸۹/۵۲
۱۲	۱۳۳/۵۷	۱۲/۸۵	۱۷۸/۵۵	۱۸۹/۲۷
۱۳	۱۰۰/۰۹	۱۸/۲۶	۱۷۹/۶۱	۱۸۸/۶۰
۱۴	۱۳۰/۷۷	۲۵/۷۵	۲۹۹/۸۳	۳۱۱/۸۳
۱۵	۱۲۶/۹۴	۲۳/۵۶	۳۵۵/۸۵	۳۹۱/۴۴

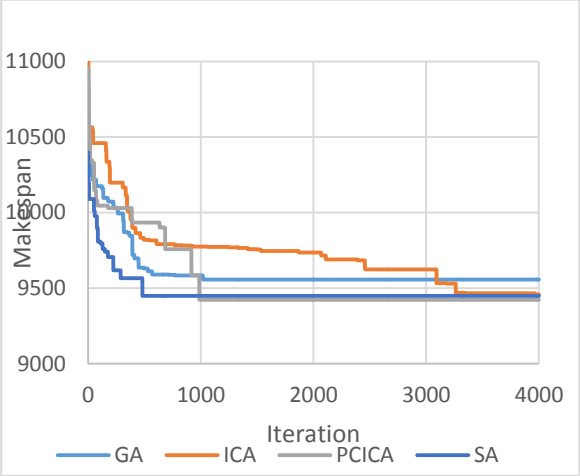
برای دیدن چگونگی همگرایی الگوریتم‌ها به راه حل نهایی آنها، مسایل (۵) و (۱۵) در گروه مسایل متوسط و بزرگ به طور تصادفی انتخاب شده است. شکل‌های ۹ و ۱۰ مشخصات همگرایی را برای الگوریتم‌ها نشان می‌دهد. همان‌طور که مشاهده می‌شود، الگوریتم SA دارای بالاترین سرعت همگرایی می‌باشد.



شکل ۹. رفتار همگرایی الگوریتم‌ها در مساله (۵)

[Downloaded from jorlu.ir on 2026-11-30]

[DOI: 10.52547/jamlu.19.4.37]



شکل ۱۰. رفتار همگرایی الگوریتم‌ها در مساله (۱۵)

۷ بحث و جمع‌بندی نتایج

به طور کلی با توجه به بررسی دو معیار عملکرد انحراف درصد نسبی (RPD) و زمان پردازش الگوریتم PCICA بهترین عملکرد را از نظر کیفیت پاسخ به خود اختصاص داده است. از لحاظ زمان پردازش نیز الگوریتم SA عملکرد مطلوب‌تری را نشان داده است. جدول ۵ به طور خلاصه بهترین نتایج به‌دست آمده توسط الگوریتم‌های فراابتکاری برای مسایل در ابعاد متوسط و بزرگ را نشان می‌دهد. نکته حایز اهمیت این است که روش فراابتکاری معرفی‌شده در این مقاله در اکثر مسایل نتایج بسیار خوب و بهتر نسبت به الگوریتم‌های فراابتکاری دیگر به‌دست آورده است.

جدول ۵. بهترین نتایج به‌دست آمده توسط الگوریتم‌های فراابتکاری

مساله	مقادیر به‌دست آمده	توسط الگوریتم
۱	۳۰۸۱	SA
۲	۳۵۱۲	SA
۳	۵۰۵۶	PCICA& SA
۴	۳۸۶۴	SA
۵	۵۱۷۸	PCICA& SA
۶	۷۸۰۳	SA
۷	۶۹۴۸	PCICA
۸	۷۵۸۷	PCICA
۹	۶۱۲۴	PCICA & SA
۱۰	۵۴۸۹	PCICA
۱۱	۸۳۲۹	PCICA
۱۲	۸۱۵۴	PCICA
۱۳	۹۳۰۶	PCICA
۱۴	۱۰۶۹۴	PCICA
۱۵	۹۳۳۶	PCICA

[Downloaded from jorlu.ir on 2026-11-30]

[DOI: 10.52547/jamlu.19.4.37]

۸ نتیجه‌گیری

یکی از مهم‌ترین مسایلی که در سال‌های اخیر در رابطه با انبارهای متقاطع به طور گسترده‌ای مورد بررسی قرار گرفته است، زمان‌بندی و تعیین ترتیب وسایل حمل در انبارهای متقاطع می‌باشد، که هم زمان‌بندی منابع درون انبار و هم زمان‌بندی وسایل حمل ورودی و خروجی را در بر می‌گیرد. این مطالعه به بررسی مساله زمانبندی کامیون‌ها در سیستم انبار متقاطع با یک انبار موقت در نزدیکی درب خروجی می‌پردازد. در واقع هدف مساله حداقل‌سازی زمان اجرای کل عملیات و تعیین بهترین توالی برای کامیون‌های ورودی و خروجی می‌باشد.

در این مقاله با بررسی مرور ادبیات کارهای گذشته، برای حل مسایل در مقیاس متوسط و بزرگ از سه الگوریتم فراابتکاری شامل الگوریتم‌های ژنتیک (GA)، رقابت استعماری (ICA) و الگوریتم تبرید شبیه‌سازی‌شده (SA) استفاده شده است. علاوه بر این در این مقاله برای اولین بار یک روش فراابتکاری ترکیبی بسیار قوی ارایه‌شده است که با بهترین عملکرد نسبت به دیگر رویکردهای فراابتکاری ارایه‌شده در مدل‌های مشابه به اهداف دست می‌یابد. این الگوریتم فراابتکاری ترکیبی از ترکیب دو الگوریتم ICA و PCA نتیجه شده است. مقایسه‌ها نشان می‌دهد که الگوریتم مبتنی بر مکانیسم یادگیری مؤلفه اصلی، PCICA، به طور قابل توجهی نسبت به الگوریتم‌های موجود ICA پیشی می‌گیرد.

سپس با استفاده از روش تاگوچی به تنظیم پارامتر الگوریتم‌های فوق پرداخته‌ایم. در نهایت نتایج عددی به‌دست آمده از تمامی الگوریتم‌های فراابتکاری مورد بررسی و تحلیل حساسیت قرار گرفته‌اند. الگوریتم‌های فراابتکاری را بر اساس معیارهای بهترین، بدترین، میانگین جواب‌ها، Rpd و زمان مورد مقایسه قرار داده‌ایم. در نتیجه الگوریتم ترکیبی PCICA از نظر کیفیت جواب بهتر از سایر الگوریتم‌ها است. همچنین از نظر زمان حل نیز الگوریتم SA بهتر از سایرین عمل نموده است. این آزمایشات نشان می‌دهد که مکانیسم انباشت اطلاعات مبتنی بر PCA می‌تواند اکثر اطلاعات اکتشافی همه پاسخ‌ها را کشف، حفظ و استفاده کند و به راه حل‌های با کیفیت بالاتر، همگرایی سریع‌تر و قابلیت اطمینان قوی‌تر کمک می‌کند.

مسایل مطرح‌شده برای سیستم‌های فراباراندار و مدل‌های آن‌ها بسیار گسترده هستند و می‌توان با در نظر گرفتن فرضیات جدید، مدل‌های سابق را توسعه داد و آن‌ها را به شرایط واقعی و کاربردی نزدیک‌تر کرد. علاوه بر آن مکانیسم PCA به‌کار گرفته‌شده در روش ICA به‌دلیل داشتن مکانیسم ذخیره‌سازی اطلاعات مفید برای یافتن جهت‌های موثرتر، برای الگوریتم بهینه‌سازی رقابت استعماری بسیار کارآمد و مؤثر است. از این رو، می‌توان به طور موثر این مکانیسم را در سایر الگوریتم‌های فراابتکاری نیز به کار گرفت.

منابع

[1] W. Wisittipanich and P. Hengmeechai. (2017). Truck scheduling in multi-door cross docking terminal by modified particle swarm optimization. Computers & Industrial Engineering, 113, 793-802.

[2] 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), 377-396.

[3] Boysen, N., & Fliedner, M. (2010). Cross dock scheduling: Classification, literature review and research agenda. Omega, 38(6), 413-422.

[4] Stephan, K., & Boysen, N. (2011). Cross-docking. Journal of Management Control, 22(1), 129-137.

[Downloaded from jorlu.ir on 2026-11-30]

[DOI: 10.52547/jamlu.19.4.37]

[5] Van Belle, J., Valckenaers, P., & Cattrysse, D. (2012). Cross-docking: State of the art. Omega, 40(6), 827-846.
[6] Ladier, A. L., & Alpan, G. (2016). Cross-docking operations: Current research versus industry practice. Omega, 62, 145-162.
[7] Agustina, D., Lee, C. K. M., & Piplani, R. (2014). Vehicle scheduling and routing at a cross docking center for food supply chains. International Journal of production economics, 152, 29-41.
[8] 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), 416-430.
[9] Shabani, A., Tavakkoli-Moghaddam, R., & Hajiaghaei-Keshteli, M. (2019). Solving a Fuzzy Fixed-Charge Transportation Problem by Meta-Heuristics with a New Encoding Scheme. Journal of Operational Research In Its Applications (Applied Mathematics)-Lahijan Azad University, 15(4), 15-35.
[10] 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), 67-78.
[11] Yu, W. (2002). Operational strategies for cross docking systems. Iowa State University.
[12] S. Ley and S. Elfayoumy, (2007) "Cross Dock Scheduling Using Genetic Algorithms", IEEE International Symposium on Computational Intelligence in Robotics and Automation, FL, USA, 416-420
[13] K. Song and F. Chen, (2007) "Scheduling cross docking logistics optimization problem with multiple inbound vehicles and one outbound vehicle", IEEE International Conference on Automation and Logistics, Shandong, China, August, 3089-3094.
[14] 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(3), 1964-1979.
[15] Maknoon, M. Y., & Baptiste, P. (2010)"Moving freight inside cross docking terminals", 8th International Conference on Supply Chain Management and Information IEEE, 1-6.
[16] Vahdani, B., & Zandieh, M. (2010). Scheduling trucks in cross-docking systems: Robust meta-heuristics. Computers & Industrial Engineering, 58(1), 12-24.
[17] Alpan, G., Larbi, R., & Penz, B. (2011). A bounded dynamic programming approach to schedule operations in a cross docking platform. Computers & Industrial Engineering, 60(3), 385-396.
[18] Bellanger, A., Hanafi, S., & Wilbaut, C. (2013). Three-stage hybrid-flowshop model for cross-docking. Computers & Operations Research, 40(4), 1109-1121.
[19] Madani-Isfahani, M., Tavakkoli-Moghaddam, R., & Naderi, B. (2014). Multiple cross-docks scheduling using two meta-heuristic algorithms. Computers & Industrial Engineering, 74, 129-138.
[20] Keshtzari, M., Naderi, B., & Mehdizadeh, E. (2016). An improved mathematical model and a hybrid metaheuristic for truck scheduling in cross-dock problems. Computers & Industrial Engineering, 91, 197-204.
[21] 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, 180-191.
[22] 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, 17-38.
[23] 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), 139-154.
[24] Vahdani, B., & Shahramfard, S. (2019). A truck scheduling problem at a cross-docking facility with mixed service mode dock doors. Engineering Computations. 36(6), 1977-2009.
[25] 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), 296-305.
[26] Boysen, N., Fliedner, M., & Scholl, A. (2010). Scheduling inbound and outbound trucks at cross docking terminals. OR spectrum, 32(1), 135-161.
[27] Mousavi, M., HAJIAGHAEI, K. M., & TAVAKKOLI, M. R. (2018). Integrated scheduling of production on single machine environment and air transportation considering charter flights and time windows for due date with two efficient meta-heuristics algorithm., Journal of Operational Research In Its Applications (Applied Mathematics), 15(2), 19-40.

[Downloaded from jorlu.ir on 2026-11-30]

[DOI: 10.52547/jamlu.19.4.37]

[28] Holland, J. H. (1992). Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. MIT press.
[29] Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. science, 220(4598), 671-680.
[30] Atashpaz-Gargari, E., & Lucas, (2007). Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition, Proceedings of IEEE Congress on Evolutionary Computation, Singapore, 4661–4667.
[31] 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), 180-201.
[32] Seyedi, I., Hamedi, M., Tavakkoli-Moghaddam, R. (2022). Optimization for a truck scheduling problem in multi-door cross dokcing with learning effect and deteriorating jobs. Journal of Transportation Research, 19(71), 183-206.

[Downloaded from jorlu.ir on 2026-11-30]

[DOI: 10.52547/jamlu.19.4.37]