

الگوریتم جدیدی برای حل مسأله مسیریابی-موجودی با ارسال مستقیم

علی حسین میرزایی^۱، عیسی نخعی کمال آبادی^{۲*}، سید حسام‌الدین ذگردی^۳

۱- دانشجوی دکتری مهندسی صنایع دانشکده فنی و مهندسی دانشگاه تربیت مدرس

۲- دانشیار دانشکده فنی و مهندسی دانشگاه تربیت مدرس

۳- دانشیار دانشکده فنی و مهندسی دانشگاه تربیت مدرس

چکیده

این مقاله به بررسی مسأله مسیریابی-موجودی چند محصولی چند دوره‌ای در یک زنجیره تأمین دو سطحی؛ شامل یک تولیدکننده و مجموعه‌ای از خرده‌فروشان اختصاص دارد. در مسأله مورد بررسی، علاوه بر مدیریت موجودی و برنامه‌ریزی توزیع، برنامه‌ریزی تولید نیز در نظر گرفته شده است. مسأله با هدف کمینه‌سازی مجموع هزینه‌های سیستم شامل هزینه‌های راه‌اندازی، توزیع و نگهداری موجودی مدل‌سازی شده است. محصولات توسط ناوگانی از وسایل حمل همسان با ظرفیت محدود تحت استراتژی ارسال مستقیم به خرده‌فروشان تحویل داده می‌شوند. همچنین، ظرفیت تولید و نگهداری محدود و کمبود غیرمجاز فرض شده است. نشان داده شده است که مسایل مشابه بدون در نظر داشتن برنامه‌ریزی توزیع در زمره مسایل با پیچیدگی سخت قرار دارند، بنابراین مسأله فوق نیز، مسأله‌ای با پیچیدگی سخت است. از این رو، در این مقاله الگوریتم بهینه‌سازی گروه ذرات بهبودیافته جدیدی برای حل آن توسعه داده شده است. الگوریتم پیشنهادی از دو بخش مجزا تشکیل شده است. نخست، مقادیر متغیرهای صفرویک با استفاده از الگوریتم پیشنهادی تعیین و سپس با حل یک مدل برنامه‌ریزی خطی، مقادیر متغیرهای

پیوسته محاسبه می‌شود. کارایی الگوریتم پیشنهادی با استفاده از مسایل نمونه تصادفی متعددی با الگوریتم‌های ژنتیک و بهینه‌سازی گروه ذرات مقایسه شده است. نتایج محاسباتی بیانگر عملکرد بهتر الگوریتم پیشنهادی است.

واژه‌های کلیدی: زنجیره تأمین، مسأله مسیریابی-موجودی، استراتژی ارسال مستقیم، بهینه‌سازی گروه ذرات، برنامه‌ریزی تولید-توزیع.

۱- مقدمه

امکانپذیر است. در مسأله مسیریابی-موجودی با تعیین برنامه توزیع بهینه‌ای که مجموع هزینه‌ها را کمینه سازد، می‌توان به این هدف دست یافت (کوردیو و همکاران، ۲۰۰۷). مسأله مسیریابی-موجودی در پژوهش‌های متعددی بررسی شده است که مرور جامعی از پژوهش‌های پیشین توسط اندرسون و همکاران^۵ (۲۰۱۰) ارائه شده است. نویسندگان با بررسی ابعاد صنعتی مسأله، طبقه‌بندی و مرور جامعی از پژوهش‌های موجود ارائه داده‌اند.

از جمله استراتژی‌های توزیعی که می‌توان در مسأله مسیریابی-موجودی به کار گرفت، استراتژی توزیع ارسال مستقیم^۶ است. ارسال مستقیم به حالتی اشاره می‌کند که در آن در هر دوره هر وسیله حمل تنها به یک خرده‌فروش محصول تحویل می‌دهد. به دلیل سادگی پیاده‌سازی این استراتژی توزیع در عمل، طبیعی است که در مسأله مسیریابی-موجودی ابتدا چنین استراتژیی بررسی شود. از این رو، استراتژی ارسال مستقیم در مسأله مسیریابی-موجودی، بطور مکرر در سیستم‌های توزیع صنعتی استفاده می‌شود. کارایی استراتژی توزیع ارسال مستقیم حداقل به بزرگی جذر کوچکترین نرخ بهره-برداری از وسایل حمل است. به بیان دیگر، در

مسأله مسیریابی-موجودی^۱ بسط مهمی از مسأله مسیریابی وسیله نقلیه^۲ است که در آن تصمیمات کنترل موجودی و مسیریابی در هم ادغام می‌شوند (کوردیو و همکاران^۳، ۲۰۰۷). مسأله مسیریابی-موجودی بیشتر در سیستم‌های مدیریت موجودی توسط فروشنده^۴ (VMI) کاربرد دارد. در سیستم‌های مدیریت موجودی توسط فروشنده، فروشنده قادر است تا زمانبندی و اندازه تحویل محصول به خرده‌فروشان را کنترل نماید. در قبال این آزادی عمل، فروشنده تضمین می‌کند که مشتریان با کمبود مواجه نمی‌شوند. در روابط سنتی‌تر میان فروشنده و مشتری که در آن مشتریان درخواست سفارش محصولات را به فروشنده می‌دادند، به دلیل زمانبندی سفارشات مشتریان، ممکن است کارایی به شدت کاهش و به نوبه آن هزینه‌های موجودی و توزیع به شدت افزایش یابد. با وجود این، تحقق کاهش هزینه‌های ناشی از به کارگیری سیستم‌های VMI در عمل ساده نیست به ویژه با افزایش تعداد و تنوع مشتریان این امر دشوارتر نیز می‌شود. با استفاده از مسأله مسیریابی-موجودی دستیابی به این هدف

1 Inventory Routing Problem (IRP)

2 Vehicle Routing Problem (VRP)

3 Cordeau et al.

4 Vendor Managed Inventory (VMI)

5 Andersson et al.

6 Direct shipment

تصمیم‌گیری برای طراحی و کنترل سیستم موجودی و حمل توسعه داده شده است. کلیوگت و همکاران^۵ (۲۰۰۲) با بررسی مسأله مسیریابی-موجودی تصادفی با ارسال‌های مستقیم و مدلسازی آن به صورت فرآیند تصمیم‌گیری مارکوفی زمان گسسته، روشی تخمینی بر مبنای برنامه‌ریزی پویا برای حل ارائه کرده‌اند. بارنز شوستر و بسک^۶ (۱۹۹۷) به ارزیابی کارایی استراتژی ارسال مستقیم در مسأله مسیریابی-موجودی با افق زمانی نامحدود پرداخته‌اند که در آن تقاضای خرده‌فروشان احتمالی با تابع توزیع معین و کمبود مجاز است.

مسأله مسیریابی وسایل نقلیه در زمره مسایل با پیچیدگی سخت^۷ قرار دارد (لنسترا و رینوی^۸، ۱۹۸۱)؛ در نتیجه، مسأله مسیریابی-موجودی که توسعه‌ای از مسأله مسیریابی وسایل نقلیه است، یک مسأله با پیچیدگی سخت است. بنابراین، در مقالات بسیاری، با استفاده از الگوریتم‌های فراابتکاری، روش‌های حل تقریبی برای مسأله توسعه داده شده است. عزیز و معین^۹ (۲۰۰۷) مسأله را در حالت چند چند محصولی، چند دوره‌ای با چند تأمین‌کننده و یک کارخانه مونتاژ و با هدف کمینه‌سازی مجموع هزینه‌های حمل و نگهداری موجودی بررسی کرده‌اند. آنان با بررسی دو نحوه نمایش مختلف جواب، یک الگوریتم ژنتیک ترکیبی بر اساس رویکرد نخست تخصیص و سپس مسیریابی ارائه کرده‌اند. معین و همکاران^{۱۰} (۲۰۱۰) در تحقیق مشابهی، الگوریتم ژنتیک ترکیبی بهبودیافته‌ای ارائه داده‌اند.

صورتی که نرخ تقاضای هر خرده‌فروش ۹۰ درصد ظرفیت وسایل حمل ضرب در کران بالای فرکانس تحویل به خرده‌فروشان باشد، آنگاه کارایی استراتژی ارسال مستقیم در حدود ۹۴/۸۶ درصد خواهد بود. کارایی ارسال مستقیم در شرایط خاص، حتی به ۱۰۰ درصد نیز می‌رسد. از این رو، در بسیاری از شرایط استفاده از استراتژی ارسال مستقیم توجیه منطقی دارد (لی و همکاران^۱، ۲۰۱۰). مقالات مختلفی با تمرکز بر استراتژی ارسال مستقیم، به بررسی مسأله مسیریابی-موجودی پرداخته‌اند که از آن جمله می‌توان به موارد زیر اشاره کرد. لی و همکاران (۲۰۱۰) به بررسی تحلیلی استراتژی ارسال مستقیم پرداخته‌اند و کارایی ارسال مستقیم را در شرایط مختلف محاسبه کرده‌اند. لی و همکاران (۲۰۰۸) مسأله مسیریابی-موجودی را در حالتی در نظر گرفته‌اند که تأمین‌کننده تنها یک وسیله حمل در اختیار داشته و در هر دوره تنها می‌تواند برای یک مشتری موجودی ارسال کند. نویسندگان الگوریتمی ابتکاری برای یافتن توالی شدنی ارائه کرده‌اند. کمبل و هاردین^۲ (۲۰۰۵) با بررسی حداقل تعداد وسایل حمل مورد نیاز در مسأله مسیریابی-موجودی با ارسال مستقیم، برای حل مسأله، الگوریتمی حریصانه^۳ ارائه داده‌اند. چنگ و دوران^۴ (۲۰۰۴) مدلی برای مسأله مسیریابی-موجودی در زنجیره تأمین جهانی نفت خام ارائه کرده‌اند که در آن تقاضای مشتریان و طول زمان سفر، غیر قطعی است. همچنین، تقاضای مشتریان متغیر با زمان (پویا) است. در این مقاله یک سیستم پشتیبانی

5 Kleywegt et al.

6 Barnes-Schuster and Bassok

7 NP-hard

8 Lenstra and Rinnooy

9 Aziz and Moin

10 Moin et al.

1 Li et al.

2 Campbell and Hardin

3 Greedy

4 Cheng and Duran

این مقاله به بررسی مسأله مسیریابی-موجودی در حالت چند دوره‌ای چند محصولی و با هدف کمینه‌سازی مجموع هزینه‌های سیستم می‌پردازد. هزینه‌های سیستم شامل هزینه‌های راه‌اندازی، توزیع و نگهداری موجودی است. علاوه بر مدیریت موجودی و برنامه‌ریزی توزیع، برنامه‌ریزی تولید نیز در مسأله، ادغام شده است. در این مقاله، مسأله مسیریابی-موجودی در یک زنجیره تأمین دوبخشی متشکل از یک تولیدکننده و مجموعه‌ای از خرده‌فروشان بررسی شده است که در آن، در هر دوره مقدار مشخصی از هریک از محصولات با استفاده از ناوگان همسانی از وسایل حمل با ظرفیت محدود و تحت استراتژی ارسال مستقیم، بین خرده-فروشان توزیع می‌گردد. خدمت‌رسانی به هریک از خرده‌فروشان توسط وسایل حمل، یک دوره زمانی طول می‌کشد. بنابراین، هر وسیله حمل در هر دوره تنها می‌تواند به یک خرده‌فروش محصول تحویل دهد. همچنین فرض شده است که محصول ارسالی به هریک از خرده‌فروشان در هر دوره تنها باید طی یک ارسال به خرده‌فروش تحویل داده شود. ظرفیت انباشت و تولید محدود است و کمبود مجاز نیست. مسأله مسیریابی-موجودی چند دوره‌ای چند محصولی با ارسال مستقیم تحت مفروضات فوق، حالت توسعه یافته‌ای از مسأله تعیین اندازه انباشته با ظرفیت محدود^{۱۱} در حالت چند محصولی، با ظرفیت تولید و هزینه‌های نگهداری ثابت است. در پژوهش‌های پیشین نشان داده شده است که مسأله تعیین اندازه انباشته در شرایط فوق، یک مسأله با

بودیا و همکاران^۱ (۲۰۰۷) در تحقیق خود با بررسی یک مسأله مسیریابی-موجودی که در آن برنامه‌ریزی تولید نیز لحاظ شده است، الگوریتم‌های حل مبتنی بر رویه جستجوی انطباقی حریصانه تصادفی^۲ توسعه داده‌اند. بودیا و پرینز^۳ (۲۰۰۹) در تحقیق مشابه‌ای، ساختاری مبتنی بر الگوریتم ممیتیک^۴ با مدیریت جمعیت جواب‌ها توسعه داده‌اند. ژائو و همکاران^۵ (۲۰۰۸) مدل یکپارچه‌ای برای مسأله مسیریابی-موجودی در زنجیره تأمین سه رده‌ای ارائه کرده و برای حل آن الگوریتم جستجوی همسایگی بزرگ متغیری^۶ توسعه داده‌اند. ژائو و همکاران (۲۰۰۷) رویکرد حل جدیدی بر اساس الگوریتم فراابتکاری جستجوی ممنوع^۷ برای حل مسأله مسیریابی-موجودی در یک زنجیره تأمین دو رده‌ای توسعه داده‌اند. اسپارچی-الکازار و همکاران^۸ (۲۰۰۷) با پیاده‌سازی الگوریتم ژنتیک برای حل مسأله مسیریابی-موجودی با چند محصول، تأثیر مقادیر مختلف پارامترهای ورودی الگوریتم ژنتیک را ارزیابی کرده‌اند تا بهترین مجموع مقادیر پارامترهای الگوریتم بدست آید. راسدیانسیا و سائو^۹ (۲۰۰۵) با توسعه مدلی برای مسأله مسیریابی-موجودی ماشین-های سکه‌ای، به ارائه الگوریتمی دو مرحله‌ای مبتنی بر الگوریتم‌های الحاق^{۱۰} و جستجوی ممنوع پرداخته‌اند.

1 Boudia et al.

2 Greedy randomized adaptive search procedure (GRASP)

3 Boudia and Prins

4 Memetic algorithm

5 Zhao et al.

6 Variable Large Neighborhood Search (VLNS)

7 Tabu Search (TS)

8 Esparcia-Alcazar et al.

9 Rusdiansyah and Tsao

10 Insertion

11 The capacitated lot size problem

۲- مدل ریاضی

مسأله مسیریابی-موجودی چند دوره‌ای چند محصولی با ارسال مستقیم را می‌توان در قالب مدل برنامه‌ریزی عدد صحیح آمیخته^۵ فرموله کرد. پیش از بیان مدل، ابتدا نمادهای استفاده شده برای توصیف مدل ارائه می‌شود:

p : نشانگر محصول

j : نشانگر گره‌ها (گره ۰ نشانگر تولیدکننده و گره‌های ۱ تا N نشانگر خرده‌فروشان)

t : نشانگر دوره‌های برنامه‌ریزی

N : تعداد خرده‌فروشان

f_t : هزینه ثابت راه‌اندازی تولید برای تولیدکننده در دوره t

I_{pjt} : میزان موجودی محصول نوع p در پایان دوره t در مرکز j

C_j : هزینه حمل محصول از تولیدکننده به خرده‌فروشی j

a_p : ضریب مصرف ظرفیت انباشت توسط محصول نوع p از فضای انبار یا وسیله حمل.

k_p : ضریب مصرف ظرفیت تولید توسط محصول نوع p

$Imax_j$: ظرفیت انباشت در مرکز j

d_{pjt} : مقدار تقاضای محصول نوع p در خرده‌فروشی j در دوره t

P_{max} : ظرفیت تولیدکننده برای تولید محصول.

Q : ظرفیت وسیله حمل و نقل.

h_{pj} : هزینه نگهداری یک واحد محصول نوع p در مرکز j .

w_{pjt} : میزان محصول نوع p ارسال شده از تولیدکننده به مرکز توزیع j در طی دوره t

P_{jt} : میزان تولید محصول نوع p در طی دوره t توسط تولیدکننده.

X_{jt} : متغیر صفر و یک نشانگر ارسال محصول به خرده‌فروش j در دوره t

Z_t : متغیر صفر و یک نشانگر تولید محصول در دوره t

مدل ریاضی مسأله، با استفاده از معادله‌های (۱) الی (۸) ارائه شده است:

پیچیدگی سخت است (بیترن و یاناس^۱، ۱۹۸۱). بنابراین، می‌توان نتیجه گرفت که مسأله بررسی شده در این مقاله نیز، با پیچیدگی سخت است. در نتیجه، حل مسأله مذکور با استفاده از روش‌های حل دقیق^۲ به ویژه برای مسایل با ابعاد بزرگ، در زمان محاسباتی معقول امکانپذیر نیست. در این صورت، می‌توان با استفاده از الگوریتم‌های فراابتکاری به جواب‌های تقریبی مناسب در زمان محاسباتی معقول دست یافت. بنا به دانش نویسندگان، در پژوهش‌های پیشین الگوریتم ابتکاری یا فراابتکاری برای حل مسأله مسیریابی-موجودی با مفروضات فوق، توسعه داده نشده است. از این رو، برای حل مسأله، الگوریتم بهینه‌سازی گروه ذرات بهبودیافته^۳ (IPSO) کارایی (IPSO) کارایی توسعه داده شده است. کارایی الگوریتم پیشنهادی با استفاده از مسایل نمونه تصادفی متعددی با الگوریتم‌های ژنتیک و بهینه‌سازی گروه ذرات^۴ (PSO) مقایسه شده است.

ساختار ارائه مطالب بدین صورت خواهد بود که پس از مقدمه ارائه شده در این بخش، در بخش دوم، مدل ریاضی مسأله ارائه خواهد شد، در بخش سوم ساختار الگوریتم حل پیشنهادی تشریح خواهد گردید، بخش چهارم به بیان نتایج محاسباتی اختصاص یافته است و سرانجام در بخش پنجم نتیجه‌گیری ارائه شده است.

1 Bitran and Yanasse

2 Exact methods

3 Improved Particle Swarm Optimization (IPSO)

4 Particle Swarm Optimization (PSO)

متغیرهای تصمیم از طریق معادله‌های (۷) و (۸) برقرار شده است.

۳- الگوریتم بهینه‌سازی گروه ذرات پیشنهادی

الگوریتم فراابتکاری بهینه‌سازی گروه ذرات روش محاسباتی تکاملی مبتنی بر جمعیت جواب‌ها است. مانند سایر الگوریتم‌های فراابتکاری، الگوریتم مذکور ابزار بهینه‌سازی است که می‌تواند برای حل انواع مختلفی از مسایل بهینه‌سازی به کار گرفته شود. این الگوریتم از جدیدترین روش‌های فراابتکاری است که با الهام‌گیری از رفتار اجتماعی گروهی از پرندگان مهاجر که در تلاش برای دستیابی به مقصد ناشناخته‌ای هستند، توسط کندی و ابره‌ارت^۱ (۱۹۹۵) در سال ۱۹۹۵ میلادی توسعه داده شده است. در الگوریتم PSO، جمعیت جواب‌ها، گروه^۲ نامیده می‌شود و هر جواب مانند یک پرنده در گروهی از پرندگان است و ذره^۳ نام دارد و شبیه کرموزوم در الگوریتم ژنتیک است. تمامی ذرات دارای مقدار شایستگی^۴ هستند که با استفاده از تابع شایستگی^۵ محاسبه می‌گردند و تابع شایستگی ذرات باید بهینه گردد. جهت حرکت هر ذره توسط بردار سرعت^۶ آن ذره معین می‌شود. برخلاف الگوریتم ژنتیک، در فرآیند تکاملی الگوریتم مذکور، پرندگان جدیدی از نسل قبل (جواب‌های جدید از جواب‌های قبلی) ایجاد نمی‌گردد، بلکه هر پرنده رفتار اجتماعی خود را با توجه به تجربیاتش و رفتار

$$\min Z = \sum_t f_t \cdot Z_t + \sum_{t, j, j \neq 0} c_j \cdot X_{jt} + \sum_{p, j, t} h_{pj} \cdot I_{pjt} \quad (1)$$

$$I_{pj(t-1)} + w_{pj} - I_{pj} = d_{pj} \quad \forall p, j | j \neq 0, t \quad (2)$$

$$I_{p0(t-1)} + P_{pt} - I_{p0t} = \sum_j w_{pj} \quad \forall p, t \quad (3)$$

$$\sum_p k_p \cdot P_{pt} \leq P_{\max} \cdot Z_t \quad \forall t \quad (4)$$

$$\sum_p a_p \cdot w_{pj} \leq Q \cdot X_{jt} \quad \forall j | j \neq 0, t \quad (5)$$

$$\sum_p a_p \cdot I_{pj} \leq \text{Imax}_j \quad \forall j, t \quad (6)$$

$$X_{jt}, Z_t \in \{0, 1\} \quad \forall j, t \quad (7)$$

$$P_{pt}, w_{pj}, I_{pj} \geq 0 \quad \forall p, j, t \quad (8)$$

معادله (۱) بیانگر تابع هدف مدل پیشنهادی

است که شامل هزینه‌های ثابت راه‌اندازی، هزینه ارسال محصول به خرده‌فروشان، هزینه‌های نگهداری محصول توسط تولیدکننده و خرده‌فروشان است.

معادله (۲) بیانگر توازن موجودی بین تقاضا برای محصول نوع p در خرده‌فروشی j در دوره t و مجموع محصول نوع p ارسال شده از تولیدکننده به خرده‌فروشی l در دوره t است. توازن میان مجموع محصول نوع p ارسال شده از تولیدکننده به تمامی خرده‌فروشی‌ها در دوره t و مجموع محصول نوع p که در دوره t ام تولید شده است، با استفاده از معادله (۳) بیان می‌شود. معادله (۴) محدودیت ظرفیت تولید را تضمین می‌کند. محدودیت ظرفیت وسایل حمل و نقل با استفاده از معادله (۵) ارائه شده است. معادله‌های (۴) و (۵) بطور ضمنی نشان می‌دهند که آیا در دوره t به ترتیب، توسط تولیدکننده تولیدی صورت گرفته است یا محصولی برای خرده‌فروشی j ام ارسال شده است. معادله (۶) تضمین‌کننده محدودیت ظرفیت نگهداری برای تولیدکننده و خرده‌فروشی‌ها است. محدودیت‌های فنی روی

1 Kennedy and Eberhart

2 Swarm

3 Particle

4 Fitness value

5 Fitness function

6 Velocity

(۹)

$$V_{it} = w.V_{it-1} + c_1.r_1.(pBest_i - x_{it}) + c_2.r_2.(nBest_i - x_{it})$$

$$-V_{max} \leq V_{it} \leq V_{max} \quad (10)$$

$$s_i = 1/1 + e^{-V_{it}} \quad (11)$$

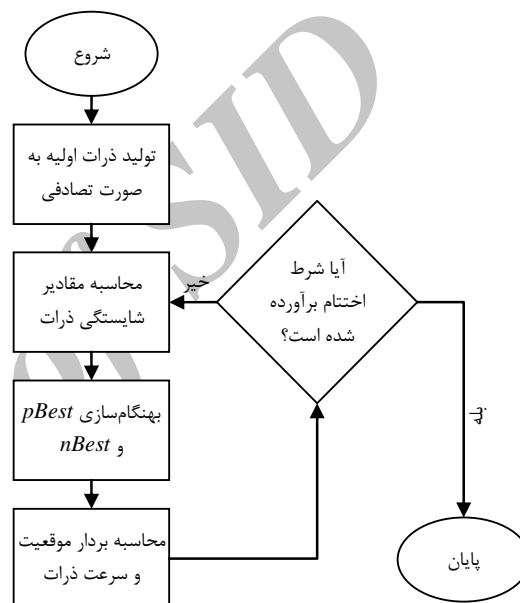
$$x_{it} = \begin{cases} 1 & \rho \leq s_i \\ 0 & otherwise \end{cases} \quad (12)$$

طبق رابطه (۹) بردار سرعت جدید هر ذره بر

اساس سرعت قبلی خود ذره ($V_{i(t-1)}$)، بهترین موقعیتی که ذره تاکنون به آن دست یافته است ($pBest_i$) و موقعیت بهترین ذره در همسایگی ذره که تابحال بدست آمده است ($nBest_i$)، محاسبه می‌گردد. در صورتی که همسایگی هر ذره شامل تمام ذرات گروه باشد، آنگاه $nBest_i$ بیانگر موقعیت بهترین ذره در میان گروه است که با $gBest$ به آن اشاره می‌شود. r_1 و r_2 دو عدد تصادفی (با توزیع یکنواخت بین $[0,1]$) هستند که مستقل از یکدیگر تولید می‌شوند. c_1 و c_2 که با نام ضرایب یادگیری به آنان اشاره شده است، تأثیر $pBest$ و $nBest$ را بر فرآیند جستجو کنترل می‌نمایند. w بیانگر ضریب وزنی اینرسی است. بردار سرعت ذرات با مقدار V_{max} محدود شده است. V_{max} به عنوان محدودیتی است که قابلیت جستجوی جهانی گروه ذرات را کنترل می‌کند. با استفاده از رابطه (۱۱) بردار سرعت هریک از ذرات به بردار احتمال تغییر، تبدیل می‌شود. در رابطه فوق، s_i بیانگر احتمال آن است که x_{it} برابر با ۱ شود. سپس با استفاده از رابطه (۱۲) بردار موقعیت هریک از ذرات بهنگام می‌گردد. در رابطه فوق، ρ عددی تصادفی با توزیع یکنواخت بین صفر و یک است.

مسأله مسیریابی-موجودی چند دوره‌ای چند محصولی با ارسال مستقیم تحت مفروضات مورد

سایر پرندگان گروه تکامل بخشیده و مطابق آن حرکت خود را به سوی مقصد بهبود می‌دهد. به عبارت دیگر، در این الگوریتم عملگرهای تکاملی چون تقاطع^۱ و جهش^۲ وجود ندارد (هو و همکاران، ۲۰۰۴؛ جائو و همکاران، ۲۰۰۶). ساختار کلی الگوریتم بهینه‌سازی گروه ذرات در شکل ۱ ارائه شده است.



شکل ۱. ساختار الگوریتم بهینه‌سازی گروه ذرات

الگوریتم PSO با گروهی از جواب‌های (ذرات) تصادفی آغاز و سپس با به‌نگام‌سازی ذرات در هر تکرار به دنبال جواب بهینه می‌گردد (هو و همکاران، ۲۰۰۴). اگر متغیرهای تصمیم و به نوبه آن موقعیت ذرات، از نوع صفر و یک باشند؛ بردارهای سرعت و موقعیت هریک از ذرات در هر تکرار الگوریتم، طبق روابط (۹) الی (۱۲) محاسبه می‌شوند (انگلبرت، ۲۰۰۵):

- 1 Crossover
- 2 Mutation
- 3 Hu et al.
- 4 Jiao et al.
- 5 Engelbrecht

مجازست. در بخش نخست، ابتدا به تعداد $SSize$ ذره با استفاده از روشی شبه تصادفی به عنوان ذرات اولیه، ایجاد می‌شوند. پس از آن، همسایگی تصادفی به اندازه $NSize$ ذره، حول هر جواب اولیه، شکل می‌گیرد. در طی گام‌های الگوریتم، کیفیت و پراکندگی جواب‌ها بهبود می‌یابد و حداکثر به تعداد $SSize$ ذره از جواب‌های خوب و غیرتکراری انتخاب و در مجموعه‌ای با نام مجموعه مرجع - برای استفاده در مراحل بعدی الگوریتم - قرار داده می‌شود. در پایان بخش نخست، جواب‌های موجود در مجموعه مرجع به عنوان ذرات اولیه برای شروع بخش دوم در نظر گرفته می‌شوند. در صورتی که تعداد ذرات موجود در مجموعه مرجع کمتر از $SSize$ ذره باشد، ذرات باقیمانده مورد نیاز به صورت تصادفی ایجاد و افزوده می‌شوند. در بخش دوم نیز فرآیند بهبود جواب‌ها با مکانیزمی متفاوت ادامه می‌یابد و در هر تکرار الگوریتم جواب‌های خوب و غیرتکراری در مجموعه مرجع قرار داده می‌شوند. در پایان بخش دوم، جواب‌های موجود در مجموعه مرجع به عنوان ذرات اولیه برای شروع بخش نخست در نظر گرفته می‌شوند. در این بخش نیز، در صورتی که تعداد ذرات موجود در مجموعه مرجع کمتر از $SSize$ ذره باشد، ذرات باقیمانده مورد نیاز به صورت تصادفی تولید و اضافه می‌شوند. بخش اول و دوم، هر کدام به ترتیب پس از $PIS1$ و $PIS2$ مرتبه تکرار یا در صورتی که به ترتیب پس از طی $CR11$ و $CR12$ مرتبه تکرار در مقدار شایستگی بهترین جواب تغییری حاصل نشود، خاتمه می‌یابند. همچنین، مرحله اول الگوریتم - رویه تکرار پیاپی بخش اول و دوم - پس از طی PI مرتبه تکرار پایان

بررسی در این مقاله، از جمله مسایل با درجه پیچیدگی سخت است. در نتیجه در این مقاله، الگوریتم بهینه‌سازی گروه ذرات بهبود یافته جدیدی برای حل مسأله فوق توسعه داده شده است.

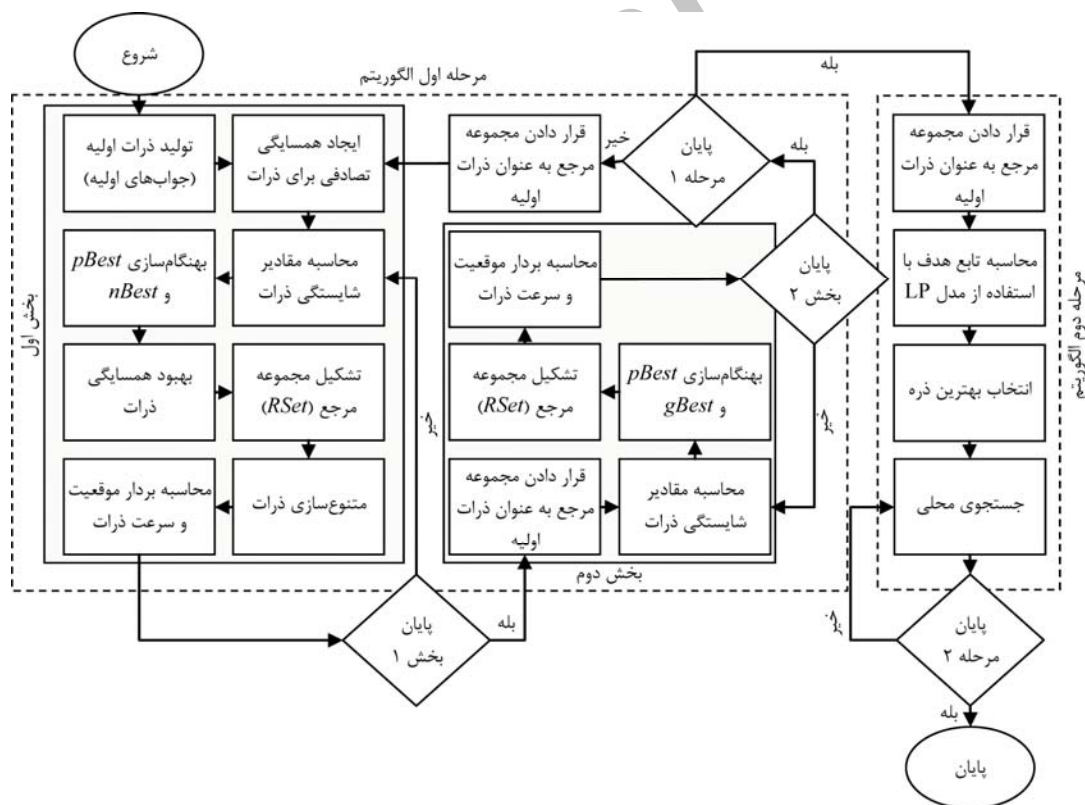
با توجه به معادلات (۱) الی (۸) مسأله مسیریابی - موجودی، یک مسأله برنامه‌ریزی عدد صحیح آمیخته است که در آن متغیرهای تصمیم به دو دسته متغیرهای پیوسته (I_{pjt}, W_{pjt}) و (P_{pt}) و متغیرهای صفر و یک $(X_{jt}$ و $Z_t)$ قابل تفکیک هستند. از آنجایی که دشواری حل مسأله مذکور ناشی از متغیرهای صفر و یک است؛ بنابراین، با تفکیک روی متغیرهای تصمیم، می‌توان راه‌حلی دو بخشی برای حل مسأله توسعه داد. در راه‌حل پیشنهادی ابتدا با استفاده از الگوریتم بهینه‌سازی گروه ذرات بهبود یافته، مقادیر متغیرهای صفر و یک $(X_{jt}$ و $Z_t)$ تعیین می‌شود، سپس با توجه به مشخص بودن مقادیر متغیرهای صفر و یک، مسأله مسیریابی - موجودی، به یک مسأله برنامه‌ریزی خطی برای تعیین مقادیر متغیرهای پیوسته (I_{pjt}, W_{pjt}) با هدف کمینه‌سازی مجموع هزینه‌ها تبدیل می‌شود که به سادگی می‌توان مسأله برنامه‌ریزی خطی حاصل را با روش‌های موجود برای حل مسایل برنامه‌ریزی خطی حل کرد و سرانجام، به جواب نهایی مسأله اصلی دست یافت.

ساختار کلی الگوریتم پیشنهادی در شکل ۲ ارائه شده است. همانگونه که در شکل نشان داده شده است، ساختار پیشنهادی مبتنی بر الگوریتم بهینه‌سازی گروه ذرات، طراحی شده است. مطابق با شکل ۲، الگوریتم پیشنهادی از دو مرحله تشکیل شده است که مرحله اول خود متشکل از دو بخش

هرچند در تمام مراحل الگوریتم پیشنهادی می‌توان مقادیر متغیرهای تصمیم پیوسته (I_{pjt}, w_{pjt}) و (P_{pt}) را با استفاده از حل یک مدل برنامه‌ریزی خطی بدست آورد؛ با وجود این، به دلیل افزایش سرعت الگوریتم، در مرحله اول، روشی تقریبی برای تعیین مقادیر متغیرهای تصمیم پیوسته توسعه و استفاده می‌شود.

در مرحله اول الگوریتم، بردارهای سرعت و موقعیت هریک از ذرات در هر تکرار الگوریتم، طبق روابط (۹) الی (۱۲) محاسبه می‌شوند. در بخش دوم از مرحله نخست الگوریتم، در رابطه (۹) مقدار $nBest_i$ با مقدار بهترین ذره در میان تمام ذرات $(gBest)$ جایگزین می‌شود.

می‌پذیرد. در پایان مرحله نخست الگوریتم، مجموعه مرجع تشکیل شده در طی این مرحله، به عنوان جواب‌های آغازین مرحله دوم الگوریتم، در نظر گرفته و الگوریتم وارد مرحله دوم می‌شود. در مرحله دوم، ابتدا با استفاده از مدل برنامه‌ریزی خطی، مقادیر بهینه متغیرهای تصمیم پیوسته با توجه به مقادیر متغیرهای صفر و یک $(X_{jt}$ و $Z_t)$ تعیین و از بین جواب‌ها، بهترین جواب انتخاب می‌شود. پس از آن با استفاده از رویه جستجوی محلی کیفیت جواب منتخب بهبود می‌یابد. مرحله دوم الگوریتم، بعد از $P2$ مرتبه تکرار یا در صورتی که پس از $CR2$ مرتبه تکرار تغییری در مقدار بهترین ذره حاصل نگردد، خاتمه می‌یابد.



شکل ۲. ساختار الگوریتم بهینه‌سازی گروه ذرات پیشنهادی

۳-۱- نحوه نمایش ذرات

یکی از مهمترین تصمیماتی که در زمان طراحی یک الگوریتم فراابتکاری می‌بایستی اتخاذ شود، نحوه نمایش جواب‌ها^۱ و نیز چگونگی برقراری ارتباطی مؤثر، یکتا و قابل شناسایی میان جواب‌ها و فضای جستجوی مسأله مورد بررسی است. در این مقاله برای نمایش ذرات، با توسعه روش عزیز و معین (۲۰۰۷)، از شیوه "نمایش زمان ارسال" برای نمایش بردار موقعیت ذرات استفاده شده است. در این شیوه نمایش از مقادیر صفر و یک برای نمایش زمان ارسال محصولات به خرده‌فروشان مختلف و زمان تولید استفاده شده است (مقادیر متغیرهای Z_i و X_{ji}). هر ۱ نشاندهنده تولید ($Z_i=1$) یا ارسال محصول به خرده‌فروش در آن دوره ($X_{ji}=1$) و مقدار صفر بیانگر عدم تولید ($Z_i=0$) یا ارسال محصول در آن دوره ($X_{ji}=0$) است. نمایی کلی از بردار موقعیت ذرات در نحوه نمایش زمان ارسال برای حل مسأله مسیریابی - موجودی با ارسال مستقیم برای حالتی با ۶ دوره برنامه‌ریزی و ۳ خرده‌فروش، در شکل ۳ ارائه شده است. موجودی خرده‌فروشان و نیز تولیدکننده در ابتدای دوره برنامه‌ریزی صفر فرض شده است، بنابراین، در نخستین دوره برنامه‌ریزی تمام مقادیر مربوط به محصولات و خرده‌فروشان برابر با ۱ است.

Z_i	۱	۰	۰	۱	۰	۰
X_{ji}	۱	۰	۱	۰	۱	۱
	۱	۰	۱	۱	۰	۱

شکل ۳. نمونه‌ای از نحوه نمایش ذرات

۳-۲- تولید جواب‌های اولیه

به طور کلی کیفیت جواب‌های آغازین بر عملکرد الگوریتم‌های فراابتکاری تأثیر بسزایی دارد. بنابراین، طراحی روشی مؤثر برای تولید جواب‌های اولیه از اهمیت بالایی برخوردار است. از این رو، در الگوریتم پیشنهادی، برای تولید ذرات اولیه، روشی شبه تصادفی طراحی شده است که با به‌کارگیری آن ذرات بین دو حالت کمترین و بیشترین تعداد دفعات ارسال ایجاد شوند. گام‌های رویه پیشنهادی برای تولید ذرات آغازین در زیر تشریح شده است:

- ۱- تمام مؤلفه‌های ذره نخست را برابر ۱ قرار دهید. این ذره، ذره‌ای با بیشترین تعداد دفعات ارسال و همواره شدنی است. به عبارت دیگر، ذره نخست بیانگر حالتی است که در آن، در تمام دوره‌ها تولید و در تمام دوره‌ها و برای تمام خرده‌فروشان ارسال داشته باشیم.
- ۲- با استفاده از رویه ارائه شده در شکل ۴، ذره دوم را ایجاد کنید. ذره دوم، ذره‌ای با کمترین تعداد دفعات ارسال است. برای تولید ذره دوم، محدودیت انباشت که با استفاده از رابطه (۶) بیان می‌شود، آزاد شده است.

```

Particle2 ← 0
RC ← [ $P_{max}$ ,  $Q$ ] - [ $\sum_j \sum_p k_p \cdot d_{pj1}$ ,  $\sum_p a_p \cdot d_{pj1}$ ]
for  $j=0$  to  $N$  do
    Particle2 $_{j1}$  ← 1
Next  $j$ 
for  $t=2$  to  $T$  do
    If  $RC_0 < \sum_j \sum_p k_p \cdot d_{pj1}$  then Particle2 $_{0t}$  ← 1
     $RC_0$  ←  $RC_0 - \sum_j \sum_p k_p \cdot d_{pj(t-1)} + P_{max} * \text{Particle2}_{0t}$ 
    for  $j=1..N$ 
        If  $RC_j < \sum_p a_p \cdot d_{pj1}$  then Particle2 $_{jt}$  ← 1
         $RC_j$  ←  $RC_j - \sum_p a_p \cdot d_{pj1} + Q * \text{Particle2}_{jt}$ 
    Next  $j$ 
Next  $t$ 
    
```

شکل ۴. شبه کد تولید ذره با کمترین تعداد دفعات تولید و

ارسال

می توان مقادیر مختلفی برای متغیرهای I_{pjt} , w_{pjt} و P_{pt} بدست آورد که از میان این مقادیر، I_{pjt} , w_{pjt} و P_{pt} را باید به گونه ای تعیین کرد که ضمن برقراری محدودیت های مسأله، کمترین هزینه ها را نیز دربرداشته باشد. در الگوریتم پیشنهادی برای یافتن مقادیر بهینه مذکور، از یک مدل برنامه ریزی خطی استفاده می شود. بدین منظور، با توجه به روابط (۲) و (۳) و با توجه به اینکه $I_{p00} = I_{p0} = 0$ خواهیم داشت:

$$I_{pjt} = \sum_{l=1}^t (w_{pjl} - d_{pjl}) \quad \forall p, t, j | j \neq 0 \quad (13)$$

$$I_{p0t} = \sum_{l=1}^t (P_{pl} - \sum_j w_{pjl}) \quad \forall p, t \quad (14)$$

با جایگذاری از روابط (۱۳) و (۱۴) در رابطه (۱) و با مشخص بودن زمان های تولید و ارسال، پس از ساده سازی خواهیم داشت:

$$\begin{aligned} \min Z &= \text{constant value} \\ &+ \sum_{p, j | j \neq 0, t} (T - t + 1)(h_{pj} - h_{p0})w_{pjt} \quad (15) \\ &+ \sum_{p, t} (T - t + 1)h_{p0} \cdot P_{pt} \end{aligned}$$

همچنین با جایگذاری از روابط (۱۳) و (۱۴) در رابطه (۶) خواهیم داشت:

$$\sum_p \sum_{l=1}^t a_p (P_{pl} - \sum_j w_{pjl}) \leq \text{Imax}_0 \quad \forall t \quad (16)$$

$$\sum_p \sum_{l=1}^t a_p \cdot w_{pjl} \leq \text{Imax}_j + \sum_p \sum_{l=1}^t a_p \cdot d_{pjl} \quad (17) \quad \forall t, j | j \neq 0$$

همچنین با توجه به روابط (۱۳)، (۱۴) و چون داریم $I_{pjt} \geq 0$ ، می توان محدودیت های (۲) و (۳) را با محدودیت های (۱۸) و (۱۹) جایگزین کرد:

$$\sum_{l=1}^t w_{pjl} \geq \sum_{l=1}^t d_{pjl} \quad \forall p, t, j | j \neq 0 \quad (18)$$

$$\sum_{l=1}^t (P_{pl} - \sum_j w_{pjl}) \geq 0 \quad \forall p, t \quad (19)$$

۳- ذره اول و دوم را به عنوان والد اول و دوم قرار دهید. پس از آن، با استفاده از عملگر تقاطع پراکنده^۱ سایر ذرات موردنیاز را ایجاد کنید. برای انجام عملگر تقاطع پراکنده، ابتدا یک آرایه تصادفی صفر و یک با اندازه برابر با اندازه ذرات ایجاد می-گردد. مؤلفه i ام ذره فرزند (ذره جدید) اگر مؤلفه i ام آرایه تصادفی ۱ بود، برابر با مؤلفه i ام والد ۱ (ذره اول) و در غیر این صورت برابر با مؤلفه i ام والد ۲ (ذره دوم) خواهد بود. نحوه انجام عملگر تقاطع پراکنده در شکل ۵ ارائه شده است.

والد (ذره) اول	۱	۱	۱	۱	۱	۱	۱
والد (ذره) دوم	۱	۰	۰	۱	۰	۰	۱
آرایه تصادفی	۱	۰	۱	۱	۱	۰	۱
فرزند (ذره جدید)	۱	۰	۱	۱	۱	۰	۱

شکل ۵. نحوه انجام عملگر تقاطع پراکنده

۳-۳. محاسبه مقادیر شایستگی

به طور معمول، در الگوریتم های فراابتکاری هر جواب در هر مرحله با استفاده از تابع شایستگی ارزیابی و مقدار شایستگی آن محاسبه می شود. در این مقاله نیز مانند سایر مسایل تک هدفه از تابع هدف مسأله که توسط رابطه (۱) بیان شده است، به عنوان تابع شایستگی ذرات استفاده می شود. در ساختار الگوریتم پیشنهادی، موقعیت هر ذره، تنها بیانگر متغیرهای Z_t و X_{jt} است. بنابراین، برای محاسبه تابع هدف لازم است، w_{pjt} ، I_{pjt} و P_{pt} در پی آن مقدار تابع هدف بر اساس Z_t و X_{jt} محاسبه شود. به ازای مقادیر ثابتی از متغیرهای Z_t و X_{jt}

1 Scattered crossover

کرد - زیرا ممکن است با حل مدل برنامه‌ریزی خطی مقدار بهینه برخی از مقادیر تولید یا ارسال صفر بدست آید که در این صورت زمان ارسال متناظر با آن نیز صفر خواهد شد.

محاسبه مقدار شایستگی ذرات با استفاده از حل مدل برنامه‌ریزی خطی تا حدی به زمان محاسباتی بالایی نیاز دارد، بنابراین برای افزایش سرعت الگوریتم، فقط در مرحله دوم الگوریتم از این روش استفاده می‌شود و برای محاسبه مقدار شایستگی در مرحله نخست الگوریتم، روشی تقریبی که در ادامه تشریح خواهد شد، توسعه داده شده است:

۱- با استفاده از رابطه (۲۸) مقدار اولیه w_{pjt} را محاسبه کنید:

$$w_{pjt} = \sum_{k=1}^{r-1} d_{pjk} \quad \forall p, j | j \neq 0 \quad (28)$$

که در رابطه مذکور، r نشانگر دوره بعدی است که ارسال بعدی در آن صورت می‌گیرد.

۲- با استفاده از روابط (۲۹) تا (۳۱) مقدار بدست آمده برای w_{pjt} را تعدیل کنید به گونه‌ای که تمام مقادیر w_{pjt} شذنی شود ($w_{pjt} \leq Q$):

$$\Delta_{jt} = X_{jt} \cdot Q - \sum_p a_p \cdot w_{pjt} \quad \forall t, j | j \neq 0 \quad (29)$$

$$\Delta_{j(t-1)} = \Delta_{j(t-1)} + \min(0, \Delta_{jt}) \quad (30)$$

$$\forall t | t > 1, j | j \neq 0$$

$$w_{pjt} = w_{pjt} \cdot (Q - \max(0, \Delta_{jt})) / \sum_p a_p \cdot w_{pjt} \quad (31)$$

$$\forall p, t, j | j \neq 0$$

۳- مقدار اولیه P_{pt} را با استفاده از رابطه (۳۲) محاسبه کنید:

$$P_{pt} = \sum_{k=1}^{r-1} w_{pjk} \quad \forall p, j | j \neq 0 \quad (32)$$

که در رابطه مذکور، r نشانگر دوره بعدی است که تولید بعدی در آن صورت می‌گیرد.

بنابراین از روابط فوق و با توجه به معادلات (۱) الی (۸)، مدل ساده شده به صورت زیر است:

$$\begin{aligned} \min Z &= \text{constant value} \\ &+ \sum_{p, j | j \neq 0, t} (T-t+1)(h_{pj} - h_{p0})w_{pjt} \quad (20) \\ &+ \sum_{p, t} (T-t+1)h_{p0} \cdot P_{pt} \end{aligned}$$

$$\sum_p k_p \cdot P_{pt} \leq P_{\max} \quad \forall t \in \{t | Z_t = 1\} \quad (21)$$

$$\sum_p a_p \cdot w_{pjt} \leq K_j \quad \forall j \in \{j | X_{jt} = 1\} \quad (22)$$

$$\sum_p a_p \cdot w_{pjt} \leq Q \quad (23)$$

$$\forall (j, t) \in \{(j, t) | X_{jt} = 1, t > 1\}$$

$$\sum_{l=1}^t w_{pjl} \geq \sum_{l=1}^t d_{pjl} \quad \forall p, j | j \neq 0, t | t > 1 \quad (24)$$

$$\sum_p \sum_{l=1}^t a_p \cdot w_{pjl} \leq \text{Imax}_j + \sum_p \sum_{l=1}^t a_p \cdot d_{pjl} \quad (25)$$

$$\forall j | j \neq 0, t | t > 1$$

$$w_{pjt} \geq d_{pjt}, w_{pjt} \geq 0, P_{pt} \geq 0 \quad (26)$$

$$\forall p, \forall (j, t) \in \{(j, t) | X_{jt} = 1\}, \forall t \in \{t | Z_t = 1\}$$

$$w_{pjt} = P_{pt} = 0 \quad (27)$$

$$\forall p, \forall (j, t) \in \{(j, t) | X_{jt} = 0\}, \forall t \in \{t | Z_t = 0\}$$

و معادلات (۱۶) و (۱۹).

کـــــه در رابطه (۲۲)،

$$K_j = \min(Q, \text{Imax}_j + \sum_p a_p \cdot d_{pjl})$$

مدل ساده شده فوق به تعداد $P(N+I)T$ متغیر و حداکثر $N(P+I)$ محدودیت کمتر از مدل اصلی دارد. حال در هر مرحله از الگوریتم، با مشخص بودن زمان‌های تولید و ارسال (X_{jt} و Z_t) و با استفاده از مدل مذکور، می‌توان متغیرهای w_{pjt} و P_{pt} و نیز مقادیر شایستگی هر یک از ذرات را محاسبه و با توجه به مقادیر حاصل از حل مدل برنامه‌ریزی خطی برای متغیرهای w_{pjt} و P_{pt} ، در صورت نیاز، اصلاح لازم در زمان‌های تولید و ارسال را نیز اعمال

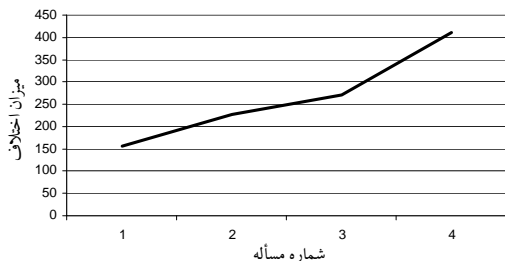
رو، در مرحله دوم الگوریتم، در صورتی که ذره‌ای نشدنی باشد نیز، از رویه تقریبی فوق برای محاسبه مقدار شایستگی آن ذره استفاده می‌شود.

تفاوت سرعت الگوریتم پیشنهادی در دو حالت محاسبه مقدار شایستگی ذرات با استفاده از حل مدل برنامه‌ریزی خطی در هر دو مرحله (LP-IPSO) و محاسبه مقدار شایستگی ذرات با استفاده از رویه تقریبی فوق در مرحله نخست و استفاده از مدل برنامه‌ریزی خطی فقط در مرحله دوم الگوریتم (IPSO) را می‌توان با استفاده از چند مثال بررسی کرد. میانگین نتایج حاصل از ۱۰ بار اجرای الگوریتم پیشنهادی در دو حالت مذکور برای حل چند مسأله نمونه (مسایل نمونه در بخش ۴ تشریح خواهند شد)، در جدول ۱ ارائه شده است.

جدول ۱. نتایج حاصل از حل مسایل نمونه با استفاده

از LP-IPSO و IPSO

شماره مسأله	تابع هدف		نشدنی بودن		زمان (ثانیه)	
	LP-IPSO	IPSO	LP-IPSO	IPSO	LP-IPSO	IPSO
۱	۲۳۵۸	۲۳۵۸	۰	۰	۱۶۰.۴	۴.۵
۲	۳۴۸۲	۳۴۸۷	۰	۰	۲۳۳	۵.۸
۳	۳۴۹۶.۳	۳۵۳۳.۶	۰	۰	۲۷۸.۶	۷.۱
۴	۵۰۴۸	۵۱۲۷.۶	۰	۰	۴۱۸.۴	۸.۷



شکل ۶. میزان اختلاف زمان محاسباتی موردنیاز برای

حل مسایل نمونه با استفاده از LP-IPSO و IPSO

۴- با استفاده از روابط (۳۳) تا (۳۵) مقدار بدست آمده برای P_{pt} را تعدیل کنید به گونه‌ای که تمام مقادیر P_{pt} شدنی شود ($P_{pt} \leq P_{max}$):

$$\Delta_{0t} = Z_t \cdot P_{max} - \sum_p k_p \cdot P_{pt} \quad \forall t \quad (33)$$

$$\Delta_{0(t-1)} = \Delta_{0(t-1)} + \min(0, \Delta_{0t}) \quad \forall t | t > 1 \quad (34)$$

$$P_{pt} = P_{pt} \cdot (P_{max} - \max(0, \Delta_{0t})) / \sum_p k_p \cdot P_{pt} \quad \forall p, t \quad (35)$$

۵- با توجه به روابط (۳۶) و (۳۷) مقدار I_{p0t} و I_{pjt} را محاسبه کنید:

$$I_{p0t} = I_{p0(t-1)} + P_{pt} - \sum_{j|j \neq 0} w_{pjt} \quad \forall p, t \quad (36)$$

$$I_{pjt} = I_{pj(t-1)} + w_{pjt} - d_{pjt} \quad \forall p, j, t \quad (37)$$

۶- با به‌کارگیری روابط (۳۸) تا (۴۰) میزان نشدنی بودن جواب‌های حاصل را بدست آورید:

$$P(X) = \sum_{k=1}^2 \delta_k \cdot P_k(X) \quad (38)$$

$$P_1(X) = \sum_{p,j,t} \max(0, -I_{pjt}) \quad (39)$$

$$P_2(X) = \sum_{j,t} \max(0, \sum_p a_p \cdot I_{pjt} - I_{max_j}) \quad (40)$$

که در روابط بالا، δ_k ضریب جریمه برای تخطی از محدودیت k ام است.

۷- با استفاده از رابطه (۴۱) مقدار شایستگی ذره را محاسبه کنید:

$$fitness(X) = Z(X) + P(X) \quad (41)$$

که در رابطه فوق، $Z(X)$ بیانگر تابع هدف حاصل از رابطه (۱) است.

در پایان رویه تقریبی نیز، مانند حل مدل برنامه‌ریزی خطی، با توجه به مقادیر حاصل برای متغیرهای w_{pjt} و P_{pt} ، در صورت نیاز اصلاح لازم در زمان‌های تولید و ارسال اعمال می‌شود. در مرحله دوم الگوریتم، در صورتی که ذره‌ای نشدنی باشد، حل مدل برنامه‌ریزی خطی امکانپذیر نیست. از این

۳-۴. ایجاد همسایگی تصادفی ذرات

در مرحله نخست الگوریتم، به منظور افزایش پراکندگی ذرات و جستجوی مناطق بیشتری از محدوده جواب، با بهره‌گیری از عملگر جهش، حول هر ذره همسایگی تصادفی تشکیل می‌شود. بدین منظور:

۱- تعداد $NSize-1$ کپی از ذره ایجاد می‌شود.
 ۲- به تعداد $NSize-1$ آرایه تصادفی صفر و یک، با تعداد مؤلفه‌های برابر با مؤلفه‌های یک ذره، ایجاد می‌گردد. $HDRate$ درصد از مؤلفه‌های آرایه تولیدشده، به صورت تصادفی برابر ۱ قرار داده می‌شود.

۳- مطابق با مؤلفه‌های با مقدار ۱ در آرایه تصادفی، مؤلفه متناظر در ذره کپی معکوس می‌گردد؛ به عبارت دیگر، مقدار ۰ به ۱ و بالعکس مقدار ۱ به ۰ تبدیل می‌شود.

نمایی کلی از نحوه انجام عملگر جهش در شکل ۷ ارائه شده است.

$HDRate$ درصد از مؤلفه‌های آرایه تصادفی برابر با ۱ است

آرایه تصادفی	۰	۱	۰	۱	۰	۰	۱
ذره کپی شده	۱	۰	۱	۱	۱	۰	۱
ذره جدید	۱	۱	۱	۰	۱	۰	۰

شکل ۷. نحوه انجام عملگر جهش

۳-۵. بهبود همسایگی ذرات

پس از ایجاد همسایگی ذرات، طی گام‌های الگوریتم، جواب‌های موجود در هر همسایگی مستقل از ذرات سایر همسایگی‌ها بهبود می‌یابند. اما به منظور افزایش میزان بهبود در کیفیت جواب‌ها طی گام‌های الگوریتم، لازم است میان همسایگی‌های

همانگونه که در جدول ۱ و شکل ۶ ارائه شده است، هرچند LP-IPSO به مقادیر تابع هدف بهتری نسبت به IPSO دست یافته است، اما به زمان محاسباتی به مراتب بیشتری نیاز دارد. با افزایش ابعاد مسأله، تفاوت سرعت الگوریتم در دو حالت افزایش می‌یابد.

تعیین مناسب ضریب جریمه از اهمیت بسزایی برخوردار است. زیرا، اگر ضریب جریمه بسیار کوچک انتخاب شود، نیروی کافی برای جلوگیری از تخطی از محدودیت‌ها وارد نمی‌شود. از سوی دیگر، اگر ضریب جریمه بسیار بزرگ انتخاب شود، تمام جواب‌های نشدنی حتی در گام‌های ابتدایی الگوریتم حذف می‌شوند و توانایی جستجوی الگوریتم کاهش می‌یابد (انگلبرکت، ۲۰۰۵). از این رو، در ساختار الگوریتم پیشنهادی، به منظور حفظ توانایی جستجوی الگوریتم در گام‌های ابتدایی و تاکید بر حذف جواب‌های نشدنی در گام‌های نهایی، طی گام‌های مرحله نخست الگوریتم، با استفاده از رابطه (۴۲) ضریب جریمه با روندی صعودی، افزایش می‌یابد.

$$\delta_{k(t+1)} = \begin{cases} \delta_{kt} & \text{if } \text{infeasibility}(X) < \varepsilon \\ \delta_{kt} + \Delta & \text{otherwise} \end{cases} \quad (42)$$

در رابطه بالا، δ_{kt} ضریب جریمه تخطی از محدودیت k ام در تکرار t ام، Δ میزان افزایش در ضریب جریمه و $\text{infeasibility}(X)$ میزان نشدنی بودن بهترین ذره است. ضریب جریمه در طی گام‌های مرحله دوم الگوریتم ثابت باقی می‌ماند.

۷-۳. متنوع‌سازی ذرات

در ساختار الگوریتم پیشنهادی، به منظور جلوگیری از همگرایی زودهنگام و ایجاد جواب‌های اولیه با پراکندگی و کیفیت خوب برای مراحل بعدی الگوریتم، مکانیزم متنوع‌سازی ذرات انجام می‌گیرد. به همین منظور، در بخش نخست مرحله اول الگوریتم، بهترین ذره در میان مجموعه مرجع انتخاب و به تعداد $MRate$ درصد از ذرات گروه، از ذره منتخب کپی ایجاد می‌شود. ذرات کپی با به‌کارگیری عملگر جهش که در بخش ۳-۴ تشریح شد، تغییر می‌یابند و جایگزین بدترین ذرات گروه می‌شوند. با استفاده از این روش، با اعمال تغییر تصادفی محدود در بهترین ذره گروه، با افزایش مؤلفه‌های خوب در میان جواب‌ها، فرآیند جستجو بهبود می‌یابد.

۸-۳. جستجوی محلی

در مرحله دوم از ساختار الگوریتم پیشنهادی، پس از انتخاب بهترین ذره، کیفیت ذره منتخب با استفاده از جستجوی محلی بهبود می‌یابد. برای این هدف، یکی از مؤلفه‌های ذره به صورت تصادفی انتخاب و مقدار آن مؤلفه معکوس می‌شود؛ به عبارت دیگر، اگر مؤلفه برابر ۱ باشد به ۰ و بالعکس اگر ۰ باشد به ۱ تبدیل می‌شود. بدین ترتیب ذره جدیدی تنها با یک مؤلفه متفاوت نسبت به ذره اولیه بدست می‌آید. با استفاده از مدل برنامه‌ریزی خطی که در بخش ۳-۳ ارائه شد، مقدار شایستگی ذره جدید محاسبه می‌گردد. اگر مقدار شایستگی ذره جدید از ذره اولیه بهتر باشد، ذره جدید جایگزین ذره اولیه می‌شود و در غیر اینصورت ذره جدید حذف و ذره اولیه بدون تغییر باقی می‌ماند. رویه مذکور حداکثر

مختلف ذرات، به گونه‌ای تبادل اطلاعات صورت گیرد. بدین منظور، پس از بهنگام‌سازی مقادیر $nBest$ در هر تکرار الگوریتم، با به‌کارگیری روش انتخاب مسابقه^۱ به تعداد $2SSize$ ذره از میان ذرات $nBest$ به عنوان والدین^۲ انتخاب می‌شوند. پس از آن با بهره‌گیری از عملگر تقاطع پراکنده که در بخش ۳-۲ تشریح شد، $SSize$ ذره جدید (فرزند^۳) ایجاد می‌شود. ذرات جدید، جایگزین بدترین ذره در هر همسایگی می‌شوند.

۶-۳. تشکیل و بهنگام‌سازی مجموعه مرجع ($RSet$)

در ساختار الگوریتم پیشنهادی، در مرحله نخست الگوریتم، به منظور حفظ جواب‌های غیرتکراری خوب در طی گام‌های الگوریتم و همچنین به منظور فراهم‌سازی جواب‌های اولیه با تنوع و کیفیت بالا برای گام‌های آتی الگوریتم، مجموعه مرجع تشکیل می‌گردد. بدین منظور از میان تمامی ذرات غیرتکراری موجود در گروه ذرات، حداکثر به تعداد $SSize$ ذره با بهترین مقادیر شایستگی انتخاب و در مجموعه مرجع قرار داده می‌شوند. در تکرارهای بعدی الگوریتم، تنها ذراتی در مجموعه مرجع قرار می‌گیرند که تکراری نبوده و به مقادیر شایستگی بهتری دست یافته باشند. رویه تشکیل مجموعه مرجع در دو بخش اول و دوم از مرحله نخست الگوریتم پیشنهادی، یکسان است.

1 Tournament selection
2 Parent
3 Child

۴-۱- چگونگی ایجاد مسایل نمونه

مسایل نمونه در دو گروه با ابعاد کوچک و بزرگ به صورت تصادفی ایجاد شده‌اند. ابعاد مسایل، تعداد دوره‌های برنامه‌ریزی و تعداد محصولات در جدول ۲ ارائه شده است. همانگونه که در جدول مذکور نشان داده شده است، در تولید مسایل نمونه، هزینه حمل محصولات از مسایل معیار مسأله مسیریابی وسیله نقلیه با ظرفیت محدود^۱ استخراج شده است (مجموعه داده‌های مسیریابی وسیله نقلیه^۲). اطلاعات لازم برای تولید مسایل نمونه در جدول ۳ ارائه شده است. در این مقاله، همانند بودیا و پرینز (۲۰۰۹) ظرفیت تولید و وسایل حمل متناسب با تقاضا، هزینه ثابت راه‌اندازی تولید و همچنین ظرفیت انبار تولیدکننده متناسب با ظرفیت تولید در نظر گرفته شده است. ظرفیت انبار خرده‌فروشان نیز با ظرفیت وسایل حمل متناسب است. بدین منظور، ظرفیت وسایل حمل ۲ برابر $(\theta_1 = 2)$ میانگین تقاضای یک خرده‌فروش و ظرفیت تولید ۳/۵ برابر میانگین تقاضای روزانه تمام خرده‌فروشان $(\theta_2 = 3.5)$ تعیین شده است. هزینه ثابت راه‌اندازی تولید ۱/۵ برابر $(\theta_3 = 1.5)$ و ظرفیت انبار تولیدکننده برابر با $(\theta_4 = 1)$ ظرفیت تولید در نظر گرفته شده است. ظرفیت انبار تمام خرده‌فروشان یکسان و برابر با ظرفیت وسیله حمل $(\theta_5 = 1)$ تعیین شده است. در جدول ۳، d_p بیانگر میانگین تقاضای محصول p ام در یک دوره است. در تولید مسایل نمونه با ۵ محصول، ۵ سطح متفاوت تقاضا شامل تقاضای خیلی کم (d_{1ji}) ، کم (d_{2ji}) ،

پس از $P2$ مرتبه تکرار یا در صورتی که در بهترین مقدار شایستگی که ذره بدان دست یافته است، طی $CR2$ تکرار متوالی بهبودی حاصل نشود، الگوریتم متوقف می‌گردد.

۴- نتایج محاسباتی

با استفاده از مسایل نمونه تصادفی متعددی، عملکرد الگوریتم پیشنهادی در مقایسه با الگوریتم بهینه‌سازی گروه ذرات، الگوریتم ژنتیک و نرم‌افزار Lingo (تنها برای مسایل با ابعاد کوچک) ارزیابی می‌شود. با مقایسه عملکرد الگوریتم پیشنهادی با الگوریتم بهینه‌سازی گروه ذرات، می‌توان به میزان بهبود حاصل از ساختار جدید طراحی شده برای الگوریتم پیشنهادی نسبت به ساختار پایه‌ای الگوریتم بهینه‌سازی گروه ذرات پی برد. الگوریتم ژنتیک نیز، از جمله معروفترین و پرکاربردترین الگوریتم‌های فراابتکاری است که توسط پژوهش‌های متعددی در حل مسایل بهینه‌سازی مختلفی به‌کار گرفته شده است. به همین دلیل، برای ارزیابی عملکرد الگوریتم پیشنهادی، از دو الگوریتم فوق به عنوان الگوریتم‌های معیار استفاده شده است.

الگوریتم‌های پیشنهادی و بهینه‌سازی گروه ذرات، در محیط Matlab 7.5 برنامه‌نویسی و الگوریتم ژنتیک با استفاده از جعبه ابزار نرم‌افزار Matlab 7.5 پیاده‌سازی شده است. تمامی الگوریتم‌های مذکور در کامپیوتری با مشخصات پردازنده AMD، ۶۴ بیتی، ۳ گیگاهرتز، ویندوز XP و ۲ گیگابایت رم، اجرا شده‌اند.

1 Capacitated vehicle routing problem (CVRP)

2 Vehicle routing data sets

جدول ۳، ایجاد می‌شود. برای مسایل با ۳ محصول، ۳ سطح تقاضای خیلی کم (d_{1jt})، متوسط (d_{3jt}) و خیلی زیاد (d_{5jt}) لحاظ شده است.

متوسط (d_{3jt})، زیاد (d_{4jt}) و خیلی زیاد (d_{5jt}) مدنظر قرار گرفته است که مقدار تقاضای هر محصول در هر دوره به صورت یکنواخت در بازه ارائه شده در

جدول ۲. چگونگی ایجاد مسایل نمونه

شماره مسأله	بعد مسأله ($T \times N \times P$)	نام مسأله	شماره مسأله	بعد مسأله ($T \times N \times P$)	نام مسأله
۱	کوچک ($10 \times 1 \times 3$)	P-n16-k8	۱۳	بزرگ ($10 \times 101 \times 3$)	M-n101-k10
۲	کوچک ($10 \times 1 \times 5$)	P-n16-k8	۱۴	بزرگ ($10 \times 101 \times 5$)	M-n101-k10
۳	کوچک ($15 \times 1 \times 3$)	P-n16-k8	۱۵	بزرگ ($15 \times 101 \times 3$)	M-n101-k10
۴	کوچک ($15 \times 1 \times 5$)	P-n16-k8	۱۶	بزرگ ($15 \times 101 \times 5$)	M-n101-k10
۵	کوچک ($10 \times 5 \times 3$)	P-n16-k8	۱۷	بزرگ ($10 \times 121 \times 3$)	M-n121-k7
۶	کوچک ($10 \times 5 \times 5$)	P-n16-k8	۱۸	بزرگ ($10 \times 121 \times 5$)	M-n121-k7
۷	کوچک ($15 \times 5 \times 3$)	P-n16-k8	۱۹	بزرگ ($15 \times 121 \times 3$)	M-n121-k7
۸	کوچک ($15 \times 5 \times 5$)	P-n16-k8	۲۰	بزرگ ($15 \times 121 \times 5$)	M-n121-k7
۹	کوچک ($10 \times 10 \times 3$)	P-n16-k8	۲۱	بزرگ ($10 \times 151 \times 3$)	M-n151-k12
۱۰	کوچک ($10 \times 10 \times 5$)	P-n16-k8	۲۲	بزرگ ($10 \times 151 \times 5$)	M-n151-k12
۱۱	کوچک ($15 \times 10 \times 3$)	P-n16-k8	۲۳	بزرگ ($15 \times 151 \times 3$)	M-n151-k12
۱۲	کوچک ($15 \times 10 \times 5$)	P-n16-k8	۲۴	بزرگ ($15 \times 151 \times 5$)	M-n151-k12

جدول ۳. نحوه تولید پارامترهای مسایل نمونه

نام پارامتر	مقدار پارامتر	نام پارامتر	مقدار پارامتر
Q	$\theta_1 \cdot \sum_p a_p \cdot d_p$	d_{1jt}	$U[1,3]$
P_{max}	$\theta_2 \cdot N \cdot \sum_p k_p \cdot d_p$	d_{2jt}	$U[7,10]$
f_t	$\theta_3 \cdot P_{max}$	d_{3jt}	$U[15,20]$
$Imax_0$	$\theta_4 \cdot P_{max}$	d_{4jt}	$U[25,35]$
$Imax_j$	$\theta_5 \cdot Q$	d_{5jt}	$U[45,60]$
h_{pj}	1	a_p, k_p	1

۲-۴- مفروضات و پارامترهای الگوریتم‌ها

برای تنظیم الگوریتم‌ها، آزمایش‌های زیادی با مجموعه مقادیر مختلف پارامترها انجام گرفته است که در پایان با استفاده از مجموعه مقادیر زیر بهترین نتایج حاصل شده است:

الگوریتم ژنتیک: تعداد تکرارها، اندازه جمعیت، تعداد جواب نخبه^۱ و ضریب جریمه برای مسایل نمونه کوچک به ترتیب برابر با ۱۰، ۱۰۰، ۲۰۰ و ۱۰۰ و برای مسایل نمونه بزرگ به ترتیب برابر با

1 Elite count

۴-۳- نتایج عددی

هریک از الگوریتم‌ها برای حل هر مسأله نمونه، ۱۰ بار اجرا شده‌اند که با تحلیل نتایج حاصل، از دیدگاه ۳ معیار عملکردی^۱ به ارزیابی عملکرد الگوریتم پیشنهادی پرداخته شده است: (۱) میانگین مقادیر تابع هدف بدست آمده، (۲) میانگین زمان محاسباتی و (۳) میزان استواری^۲ نتایج.

هر اندازه تغییرات مقادیر تابع هدف در اجراهای مختلف الگوریتمی مشخص، کمتر باشد؛ میزان استواری الگوریتم بیشتر خواهد بود. معیار عملکردی میزان استواری نتایج با استفاده از رابطه (۴۳) محاسبه می‌گردد (انگلیبرکت، ۲۰۰۵):

$$robustness = [\bar{\theta} - \sigma_{\theta}, \bar{\theta} + \sigma_{\theta}] \quad (43)$$

در رابطه فوق، $\bar{\theta}$ میانگین و σ_{θ} واریانس مقادیر تابع هدف بدست آمده در اجراهای مختلف الگوریتم است. هر اندازه واریانس نتایج و به نوبه آن بازه ارائه شده در رابطه (۴۳) کوچکتر باشد، میزان استواری الگوریتم بیشتر است.

در جداول ۴ و ۵، میانگین مقادیر تابع هدف بدست آمده و زمان‌های محاسباتی صرف شده توسط سه الگوریتم برای حل هر مسأله، ارائه شده است. در برخی از موارد، در حل یک مسأله مشخص، پس از اختتام الگوریتم، جوابی نشدنی حاصل شده است. در این صورت، میزان تخطی از محدودیت‌ها در هر اجرای الگوریتم برای آن مسأله تعیین و ذخیره و در پایان میانگین میزان تخطی از محدودیت‌ها در تمام اجراهای الگوریتم برای آن مسأله محاسبه می‌شود. در جداول ۴ و ۵، میانگین مقدار تخطی از محدودیت‌ها

۱۵۰۰، ۳۵۰، ۵۰ و ۱۰۸۵ قرار داده شده است. سایر پارامترها همانند گزینه‌های پیش فرض جعبه‌ابزار Matlab در نظر گرفته شده است.

الگوریتم بهینه‌سازی گروه ذرات: تعداد تکرارها، تعداد ذرات، ضرایب یادگیری (c_1 و c_2)، V_{max} و ضریب جریمه برای مسایل نمونه کوچک به ترتیب برابر با ۲۰۰، ۱۰۰، ۲، ۲، ۶، ۱۰۰ و برای مسایل نمونه بزرگ به ترتیب برابر با ۱۵۰۰، ۳۵۰، ۲، ۲، ۶ و ۱۰۸۵ قرار داده شده است. ضریب اینرسی به صورت خطی از مقدار ۱.۴ به ۰.۹ در طول اجرای الگوریتم کاهش می‌یابد.

الگوریتم بهینه‌سازی گروه ذرات بهبودیافته: تعداد تکرارهای بخش اول ($PIS2$) و دوم ($PIS1$)، مرحله اول ($P1$) و دوم ($P2$)، حداکثر تکرار بدون بهبود در بهترین جواب برای بخش اول ($C11$) و دوم ($C12$) و مرحله دوم ($C2$)، تعداد ذرات ($SSize$)، اندازه همسایگی ذرات ($NSize$)، ضرایب یادگیری (c_1 و c_2)، V_{max} در بخش اول و دوم، $MRate$ ، $HDRate$ ضریب جریمه اولیه برای بخش اول (δ_{11}) و دوم (δ_{12}) و مرحله دوم (δ_2) و میزان افزایش در ضریب جریمه در هر گام (Δ) برای مسایل نمونه کوچک به ترتیب برابر است با ۵۰، ۵۰، ۱۰، ۲۵۰، ۲۵، ۲۵، ۵۰، ۲۰، ۱۰، ۲، ۲، ۳، ۶، ۰.۰۷، ۰.۱، ۱۰، ۱۰، ۷۵ و ۰.۱ و برای مسایل نمونه بزرگ به ترتیب برابر با ۷۵، ۷۵، ۱۰، ۳۵۰، ۳۵، ۳۵، ۷۵، ۳۰، ۱۰، ۲، ۲، ۶، ۶، ۰.۱، ۰.۱، ۱۰۰، ۱۰۰ و ۷۵۰ و ۰.۵ قرار داده شده است. ضریب اینرسی به صورت خطی از مقدار ۱.۴ به ۰.۹ در طول اجرای الگوریتم کاهش می‌یابد.

1 Performance measure

2 Robustness

نزدیک به بهینه و در مسأله نمونه ۱ به مقدار بهینه دست یابد. همچنین، الگوریتم پیشنهادی در تمام مسایل نمونه به جوابی شدنی رسیده است، در حالی که الگوریتم بهینه‌سازی گروه ذرات در برخی از موارد به نتایج نشدنی منتج شده است.

در حل یک مسأله، در ستون نشدنی بودن ارائه شده است. با توجه به جداول مذکور، الگوریتم بهینه‌سازی گروه ذرات بهبودیافته پیشنهادی در تمام مسایل نمونه نسبت به الگوریتم‌های معیار، به نتایج بهتری دست یافته است. در مسایل نمونه با ابعاد کوچک، الگوریتم پیشنهادی توانسته است به مقادیری بسیار

جدول ۴. میانگین مقدار تابع هدف و زمان محاسباتی حل مسایل نمونه تصادفی با ابعاد کوچک

PSO			GA			IPSO			Lingo		شماره مسأله
زمان نشدنی بودن (ثانیه)	تابع هدف	زمان نشدنی بودن (ثانیه)	تابع هدف	زمان نشدنی بودن (ثانیه)	تابع هدف	زمان نشدنی بودن (ثانیه)	تابع هدف	زمان نشدنی بودن (ثانیه)	تابع هدف		
۰.۹	۰	۲۳۹۴.۸	۱.۲	۰	۲۳۹۰.۲	۴.۵	۰	۲۳۵۸	۱	۲۳۵۸	۱
۱.۳	۰	۳۴۹۷.۲	۱.۴	۰	۳۵۳۳.۹	۵.۸	۰	۳۴۸۷	۱	۳۴۸۲	۲
۱.۴	۰	۳۶۵۶.۹	۱.۴	۰	۳۶۳۷.۴	۷.۱	۰	۳۵۳۳.۶	۳	۳۴۸۱	۳
۱.۸	۰	۵۴۰۹.۵	۱.۹	۰	۵۳۰۸.۴	۸.۸	۰	۵۱۲۷.۶	۶	۵۰۴۸	۴
۳.۳	۰	۱۳۲۳۰.۲	۳.۱	۰	۱۳۳۰۱.۴	۱۴.۵	۰	۱۲۵۸۱.۴	۲۶	۱۲۵۵۴.۴	۵
۴.۹	۰	۱۸۷۲۳.۷	۴.۹	۰	۱۸۷۷۱.۹	۲۲.۱	۰	۱۸۲۳۸	۲۱	۱۸۱۵۰.۴	۶
۵	۰	۱۸۸۸۸.۷	۴.۸	۰	۱۸۹۹۰.۳	۲۳.۸	۰	۱۸۳۹۶	۱۰۸۰۰*	۱۶۴۳۸.۶	۷
۷.۶	۰.۱	۲۷۹۴۷.۴	۶.۹	۰	۲۸۵۸۴.۲	۳۹.۳	۰	۲۶۴۳۵.۵	۶۸۵۸	۲۶۱۸۳	۸
۶.۲	۰.۲	۲۶۸۰.۲	۵.۹	۰	۲۶۹۹۲.۸	۲۸.۴	۰	۲۵۵۶۵.۸	۵۱۰	۲۵۱۸۴.۶	۹
۹.۸	۰	۳۹۳۳۷.۸	۸.۹	۰	۳۹۲۲۸.۴	۴۳.۴	۰	۳۷۱۶۳.۷	۷۹۴۳	۳۶۴۱۱.۶	۱۰
۹.۳	۰.۴	۴۰۱۵۲.۸	۹	۰	۳۹۹۲۹.۲	۴۹.۱	۰	۳۷۵۵۱.۹	۱۰۸۰۰	۳۶۶۰۸	۱۱
۱۵.۶	۰.۳	۵۷۲۷۵.۵	۱۵	۰	۵۷۱۲۱.۹	۷۳.۷	۰	۵۳۱۴۱.۲	۱۰۸۰۰	۵۲۴۱۸	۱۲
۵.۶	۰.۱	۲۱۴۴۳	۵.۴	۰	۲۱۴۸۲.۵	۲۶.۷	۰	۲۰۲۹۸.۳	۳۹۸۰.۸	۱۹۸۵۹.۸	میانگین

- بهترین مقدار بدست آمده برای هر مسأله نمونه پررنگ شده است.

* نرم‌افزار به جواب بهینه نرسیده و متوقف شده است، بهترین جواب بدست آمده گزارش شده است.

جدول ۵. میانگین مقدار تابع هدف و زمان محاسباتی حل مسایل نمونه تصادفی با ابعاد بزرگ

شماره مسأله	PSO			GA			IPSO		
	زمان نشدنی بودن (ثانیه)	تابع هدف	زمان نشدنی بودن (ثانیه)	تابع هدف	زمان نشدنی بودن (ثانیه)	تابع هدف	زمان نشدنی بودن (ثانیه)	تابع هدف	
۱۳	۱۸۹۴	۰	۲۷۲۰۸۴	۱۷۶۰	۰	۲۷۳۶۸۲	۸۶۳	۰	۲۵۶۲۵۷
۱۴	۲۸۰۱	۰	۳۸۶۳۵۷	۲۷۰۶	۰	۳۸۵۱۷۶	۱۳۹۰	۰	۳۷۰۹۱۲
۱۵	۲۸۶۵	۰	۴۲۱۷۹۷	۲۶۶۸	۰	۴۰۶۵۱۶	۱۶۸۱	۰	۳۸۵۸۶۰
۱۶	۴۲۲۸	۰	۵۹۴۲۷۴	۴۱۳۶	۰	۵۷۸۴۴۴	۲۶۹۸	۰	۵۵۸۹۶۳
۱۷	۲۲۶۲	۰	۳۳۵۷۹۲	۲۱۱۴	۰	۳۲۶۶۲۱	۱۰۹۶	۰	۳۱۲۳۱۵
۱۸	۳۳۵۶	۰	۴۷۷۱۷۸	۳۲۵۳	۰	۴۶۲۵۸۲	۱۹۶۴	۰	۴۴۹۰۰۷
۱۹	۳۴۲۵	۱	۵۹۰۱۲۲	۳۲۱۰	۰	۴۹۸۸۰۱	۱۹۸۹	۰	۴۷۲۱۴۷
۲۰	۵۰۷۷	۲	۸۱۶۳۴۸	۴۹۲۱	۰	۶۸۵۵۳۷	۳۲۰۵	۰	۶۱۶۶۳۹
۲۱	۲۸۱۱	۰	۴۰۴۸۱۱	۲۶۳۶	۰	۳۸۰۰۹۸	۱۱۹۸	۰	۳۶۹۹۹۶
۲۲	۴۲۷۸	۰	۵۷۳۹۲۱	۴۰۴۱	۰	۵۴۷۷۳۲	۲۲۲۴	۰	۵۴۱۰۹۳
۲۳	۴۲۵۳	۳۵	۸۷۰۵۵۲	۴۰۰۵	۰	۵۷۵۱۰۹	۲۴۸۴	۰	۵۵۶۴۷۳
۲۴	۶۳۳۹	۵۱	۱۲۷۶۸۵۶	۶۱۲۶	۰	۸۳۰۵۰۵	۴۱۲۷	۰	۸۱۲۳۰۳
میانگین	۳۶۳۲	۷	۵۵۵۰۰۸	۳۴۶۵	۰	۴۹۵۹۰۰	۲۰۷۷	۰	۴۸۰۱۶۴

- بهترین مقدار بدست آمده برای هر مسأله نمونه پرننگ شده است.

۰/۰۱ ($\alpha = 0.01$) می توان یکسان نبودن عملکرد الگوریتم‌ها را نتیجه گرفت.

Source of variation	DF	SS	MS	F	P-value
Problem No.	11	9.62E+09	8.74E+08	1648.61	0
Algorithm	2	1.09E+07	5.43E+06	10.24	0.001
Error	22	1.17E+07	5.30E+05		
Total	35	9.64E+09			

شکل ۸. تحلیل واریانس دو طرفه بر روی نتایج حاصل از حل مسایل با ابعاد کوچک

Source of variation	DF	SS	MS	F	P-value
Problem No.	11	1.29E+12	1.17E+11	17.53	0
Algorithm	2	7.67E+10	3.84E+10	5.73	0.01
Error	22	1.47E+11	6.69E+09		
Total	35	1.51E+12			

شکل ۹. تحلیل واریانس دو طرفه بر روی نتایج حاصل از حل مسایل با ابعاد بزرگ

مقایسه میان زمان‌های محاسباتی صرف شده توسط هر سه الگوریتم در حل مسایل با ابعاد کوچک

برای آزمودن وجود تفاوتی معنی‌دار میان عملکرد الگوریتم پیشنهادی با الگوریتم‌های معیار از تحلیل واریانس دو طرفه^۱ بر روی مقادیر تابع هدف بدست آمده توسط سه الگوریتم در حل مسایل با ابعاد مختلف استفاده می‌کنیم. بدین منظور الگوریتم‌ها به عنوان تیمار^۲ و مسایل با ابعاد مختلف به عنوان بلوک^۳ در نظر گرفته شده‌اند. تحلیل واریانس با استفاده از نرم‌افزار Minitab 14 انجام شده است که نتایج تحلیل واریانس مسایل با ابعاد کوچک و بزرگ، به ترتیب در شکل‌های ۸ و ۹ ارائه شده است. با توجه به شکل‌های ۸ و ۹، در هر دو گروه مسایل نمونه با ابعاد کوچک و بزرگ در سطح معنی‌دار

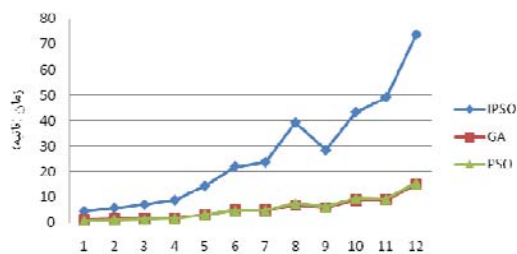
1 Two-way analysis of variance (ANOVA)
2 Treatment
3 Block

الگوریتم پیشنهادی در حل مسایل با ابعاد کوچک به زمان محاسباتی بیشتری نسبت به الگوریتم‌های معیار نیاز دارد. در حالی که در حل مسایل با ابعاد بزرگ زمان محاسباتی الگوریتم پیشنهادی در قیاس با الگوریتم‌های معیار به مراتب کمتر است. این امر را اینگونه می‌توان تفسیر کرد که در مسایل نمونه با ابعاد کوچک، هریک از الگوریتم‌ها برای دستیابی به جوابی مناسب، می‌بایست فضای جستجوی کوچکی را پوشش دهند. بنابراین، الگوریتم‌های معیار که تنها از یک بخش تشکیل شده‌اند در زمان محاسباتی اندکی قادرند به جوابی مناسب همگرا شوند. با وجود این، ساختار الگوریتم پیشنهادی از چند بخش تشکیل شده است و هر بخش به حداقل زمان محاسباتی نیاز دارد. به همین دلیل، الگوریتم پیشنهادی در حل مسایل با ابعاد کوچک، برای رسیدن به جوابی مناسب زمان محاسباتی بیشتری نسبت به دو الگوریتم معیار صرف می‌کند. اما در مسایل با ابعاد بزرگ، ساختار چند بخشی الگوریتم پیشنهادی موجب پوشش مؤثرتر و کاراتر فضای جستجو می‌شود. از این رو، الگوریتم پیشنهادی می‌تواند در زمان محاسباتی کمتری به نتایج مطلوب‌تری نسبت به دو الگوریتم معیار دست یابد.

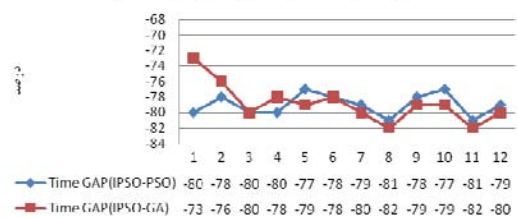
میزان استواری هر سه الگوریتم در حل مسایل کوچک و بزرگ، به ترتیب در جداول ۶ و ۷ ارائه شده است. مطابق با جداول مذکور، به طور میانگین الگوریتم پیشنهادی در مقایسه با دو الگوریتم معیار در هر دو گروه مسایل کوچک و بزرگ از تغییرپذیری کمتر و به نوبه آن از استواری بیشتری برخوردار است.

و بزرگ به ترتیب در شکل‌های ۱۰ و ۱۱ نمایش داده شده است.

الف) زمان محاسباتی

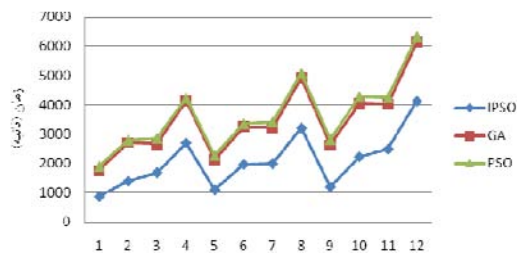


ب) درصد اختلاف زمان محاسباتی الگوریتم پیشنهادی و الگوریتم‌های معیار

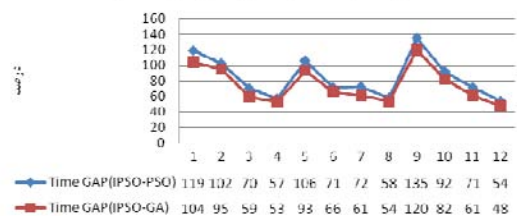


شکل ۱۰. مقایسه زمان محاسباتی الگوریتم پیشنهادی و الگوریتم‌های معیار در حل مسایل با ابعاد کوچک

الف) زمان محاسباتی



ب) درصد اختلاف زمان محاسباتی الگوریتم پیشنهادی و الگوریتم‌های معیار



شکل ۱۱. مقایسه زمان محاسباتی الگوریتم پیشنهادی و الگوریتم‌های معیار در حل مسایل با ابعاد بزرگ

خطی در مرحله دوم الگوریتم پیشنهادی برای محاسبه تابع هدف ذرات، به شدت بر سرعت الگوریتم می‌افزاید. بنابراین، می‌توان نتیجه گرفت که الگوریتم پیشنهادی با سرعت و ثبات بیشتری به نتایج بهتری دست می‌یابد.

۴-۴- تحلیل حساسیت

تعداد دفعات و زمان بهینه ارسال محصولات به خرده‌فروشان و تولید محصولات توسط تولیدکننده در پی آن ساختار مسایل نمونه، به دو عامل نسبت مجموع هزینه‌های راه‌اندازی و ارسال به هزینه‌های نگهداری و نسبت ظرفیت تولید، انبار یا وسایل حمل به تقاضا، وابسته است. با کاهش نسبت مجموع هزینه‌های راه‌اندازی و ارسال به هزینه‌های نگهداری و نیز با کاهش نسبت ظرفیت تولید، انبار یا وسایل

عملکرد بهتر الگوریتم پیشنهادی را می‌توان ناشی از ساختار جدید طراحی شده برای آن دانست. این مطلب از چهار بعد قابل بررسی است: (۱) نحوه تولید جواب‌های اولیه در الگوریتم پیشنهادی به گونه‌ای است که جواب‌هایی خوب برای مراحل بعدی فراهم می‌گردد، (۲) با تشکیل مجموعه مرجع در مرحله نخست الگوریتم، جواب‌های با کیفیت و منحصربفردی که در هر تکرار تولید شده‌اند، حفظ می‌شوند. بدین ترتیب جواب‌هایی خوب و متنوع برای گام‌های بعدی بدست می‌آید و جواب‌های نامناسب از چرخه الگوریتم حذف می‌شوند، (۳) با تشکیل همسایگی حول ذرات در بخش نخست الگوریتم، همسایگی هر ذره با دقت بیشتری جستجو و امکان بهبود در ذرات بررسی می‌شود، (۴) استفاده از رویه تقریبی در مرحله نخست و مدل برنامه‌ریزی

جدول ۶. میزان استواری الگوریتم‌ها در حل مسایل نمونه تصادفی با ابعاد کوچک

شماره مسأله	PSO استواری			GA استواری			IPSO استواری		
	طول بازه	کران پایین بازه	کران بالای بازه	طول بازه	کران پایین بازه	کران بالای بازه	طول بازه	کران پایین بازه	کران بالای بازه
۱	۱۰۴	۲۳۴۲۸	۲۴۴۶۸	۴۸	۲۳۶۶۲	۲۴۱۴۲	۰	۲۳۵۸	۲۳۵۸
۲	۴۴	۳۴۷۵۲	۳۵۱۹۲	۱۲۸	۳۴۶۹۹	۳۵۹۷۹	۰	۳۴۸۷	۳۴۸۷
۳	۲۵۲	۳۵۳۰۹	۳۷۸۲۹	۱۲۶	۳۵۷۴۴	۳۷۰۰۴	۱۱۰	۳۴۷۸۶	۳۵۸۸۶
۴	۲۱۴	۵۳۰۲۵	۵۵۱۶۵	۱۷۰	۵۲۲۳۴	۵۳۹۳۴	۲۰۶	۵۰۲۴۶	۵۲۳۰۶
۵	۵۴۴	۱۲۹۵۸۲	۱۳۵۰۲۲	۵۸۶	۱۳۰۰۸۴	۱۳۵۹۴۴	۳۸	۱۲۵۶۲۴	۱۲۶۰۰۴
۶	۶۶	۱۸۶۹۰۷	۱۸۷۵۶۷	۳۶۰	۱۸۵۹۱۹	۱۸۹۵۱۹	۱۶۰	۱۸۱۵۸	۱۸۳۱۸
۷	۲۴۴	۱۸۷۶۶۷	۱۹۰۱۰۷	۵۲۶	۱۸۷۲۷۳	۱۹۲۵۳۳	۲۷۰	۱۸۲۶۱	۱۸۵۳۱
۸	۱۴۳۴	۲۷۲۳۰۴	۲۸۶۶۴۴	۱۰۸۲	۲۸۰۴۳۲	۲۹۱۲۵۲	۹۳۶	۲۵۹۶۷۵	۲۶۹۰۳۵
۹	۱۴۵۴	۲۶۰۷۵	۲۷۵۲۹	۱۵۳۲	۲۶۲۲۶۸	۲۷۷۵۸۸	۴۰	۲۵۵۴۵۸	۲۵۵۵۵۸
۱۰	۱۶۸۸	۳۸۴۹۳۸	۴۰۱۸۱۸	۱۱۰۰	۳۸۶۷۸۴	۳۹۷۷۸۴	۵۶	۳۷۱۳۵۷	۳۷۱۹۱۷
۱۱	۱۴۸۲	۳۹۴۱۱۸	۴۰۸۹۳۸	۳۱۰۴	۳۸۳۷۷۲	۴۱۴۸۱۲	۱۵۷۲	۳۶۷۶۵۹	۳۸۳۳۷۹
۱۲	۱۶۰۸	۵۶۴۷۱۵	۵۸۰۷۹۵	۲۸۳۴	۵۵۷۰۴۹	۵۸۵۳۸۹	۱۹۷۸	۵۲۱۵۲۲	۵۴۱۳۰۲
میانگین	۷۶۱۲	۲۱۰۶۲۵	۲۱۸۲۳۶	۹۶۶۳	۲۰۹۹۹۳	۲۱۹۶۵۷	۴۴۷۲	۲۰۰۷۴۷	۲۰۵۲۱۹

جدول ۷. میزان استواری الگوریتم‌ها در حل مسایل نمونه تصادفی با ابعاد بزرگ

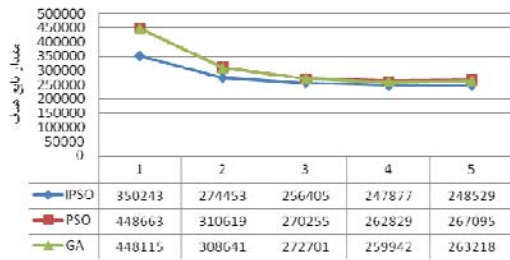
شماره مسأله	IPSO استواری			GA استواری			PSO استواری		
	کران بالای بازه	کران پایین بازه	طول بازه	کران بالای بازه	کران پایین بازه	طول بازه	کران بالای بازه	کران پایین بازه	طول بازه
۱۳	۲۵۶۶۶۸	۲۵۵۸۴۶	۸۲۲	۲۸۲۹۱۱	۲۶۴۴۵۳	۱۸۴۵۸	۲۷۷۳۷۹	۲۶۶۷۸۹	۱۰۵۹۰
۱۴	۳۷۱۲۹۶	۳۷۰۵۲۸	۷۶۸	۴۰۱۰۷۶	۳۶۹۲۷۶	۳۱۸۰۰	۳۹۰۸۵۳	۳۸۱۸۶۱	۸۹۹۲
۱۵	۳۹۱۴۲۵	۳۸۰۲۹۵	۱۱۱۳۰	۴۲۱۷۳۰	۳۹۱۳۰۲	۳۰۴۲۸	۴۳۱۶۱۹	۴۱۱۹۷۵	۱۹۶۴۴
۱۶	۵۵۹۴۰۲	۵۵۸۵۲۴	۸۷۸	۵۹۹۷۵۴	۵۵۷۱۳۴	۴۲۶۲۰	۶۰۳۳۶۲	۵۸۵۱۸۶	۱۸۱۷۶
۱۷	۳۱۳۵۱۴	۳۱۱۱۱۶	۲۳۹۸	۳۳۷۴۲۲	۳۱۵۸۲۰	۲۱۶۰۲	۳۴۲۲۴۲	۳۲۹۳۴۲	۱۲۹۰۰
۱۸	۴۵۰۲۲۶	۴۴۷۷۸۸	۲۴۳۸	۴۷۳۳۱۴	۴۵۱۸۵۰	۲۱۴۶۴	۴۹۴۱۹۱	۴۶۰۱۶۵	۳۴۰۲۶
۱۹	۴۷۳۲۲۰	۴۷۱۰۷۴	۲۱۴۶	۵۱۸۵۲۰	۴۷۹۰۸۲	۳۹۴۳۸	۶۵۹۳۷۷	۵۲۰۸۶۷	۱۳۸۵۱۰
۲۰	۶۷۸۹۲۹	۶۷۴۳۴۹	۴۵۸۰	۷۰۶۸۷۳	۶۶۴۲۰۱	۴۲۶۷۲	۹۳۱۰۶۴	۷۰۱۶۳۲	۲۲۹۴۳۲
۲۱	۳۷۰۵۰۹	۳۶۹۴۸۳	۱۰۲۶	۳۸۶۸۳۰	۳۷۳۳۶۶	۱۳۴۶۴	۴۲۳۴۰۸	۳۸۶۲۱۴	۳۷۱۹۴
۲۲	۵۴۱۵۸۷	۵۴۰۵۹۹	۹۸۸	۵۵۴۹۱۹	۵۴۰۵۴۵	۱۴۳۷۴	۵۹۸۲۴۵	۵۴۹۵۹۷	۴۸۶۴۸
۲۳	۵۶۴۷۶۰	۵۴۸۱۸۶	۱۶۵۷۴	۵۹۰۴۸۸	۵۵۹۷۳۰	۳۰۷۵۸	۹۴۴۵۳۹	۷۹۶۵۶۵	۱۴۷۹۷۴
۲۴	۸۱۲۸۹۳	۸۱۱۷۱۳	۱۱۸۰	۸۴۰۳۳۰	۸۲۰۶۸۰	۱۹۶۵۰	۱۳۵۳۲۸۸	۱۲۰۰۴۲۴	۱۵۲۸۶۴
میانگین	۴۸۲۰۳۶	۴۷۸۲۹۲	۳۷۴۴	۵۰۹۵۱۴	۴۸۲۲۸۷	۲۷۲۲۷	۶۲۰۷۹۷	۵۴۹۲۱۸	۷۱۵۷۹

جدول ۸. مقادیر پارامترهای مسأله در حالات مختلف برای تحلیل حساسیت

حالت پارامتر	۱	۲	۳	۴	۵
θ_1	۱.۵	۱.۷۵	۲	۲.۲۵	۲.۵
θ_2	۱.۵	۲.۵	۳.۵	۴.۵	۵.۵
θ_3	۰.۵	۱	۱.۵	۲	۲.۵
θ_4	۰.۵	۰.۷۵	۱	۱.۲۵	۱.۵
θ_5	۰.۵	۰.۷۵	۱	۱.۲۵	۱.۵

حمل به تقاضا، ارسال و تولید محصولات در فواصل زمانی کوتاه‌تر رخ می‌دهد. بنابراین، برای ارزیابی جامع‌تر عملکرد الگوریتم پیشنهادی، نسبت به مقادیر پارامترهای Q ، P_{max} ، f_i ، $Imax_0$ و $Imax_j$ تحلیل حساسیت صورت می‌گیرد. بدین منظور، با در نظر گرفتن $N=101$ ، $T=10$ و $P=3$ ، مطابق با جدول ۸، مقدار متفاوت به ترتیب برای پارامترهای θ_1 ، θ_2 ، θ_3 ، θ_4 و θ_5 تعیین می‌شود.

الف) تحلیل میانگین مقادیر تابع هدف



ب) درصد اختلاف میان الگوریتم معیار و الگوریتم پیشنهادی

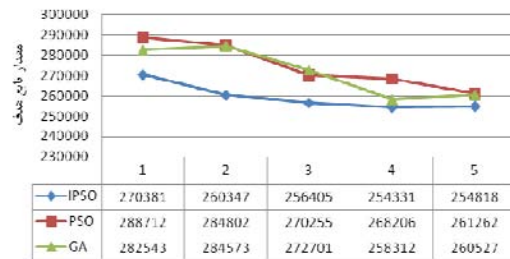


شکل ۱۳. تحلیل حساسیت نسبت به پارامتر P_{max}

هریک از الگوریتم‌ها برای حل هر حالت، ۱۰ بار اجرا شده‌اند که میانگین مقادیر تابع هدف بدست آمده و نیز درصد اختلاف میان میانگین مقادیر تابع هدف بدست آمده توسط هر یک از الگوریتم‌های معیار با الگوریتم پیشنهادی، در نمودارهای ۱۲ الی ۱۶ نشان داده شده است. این نکته قابل ذکر است که هر سه الگوریتم در تمامی حالت‌ها به جوابی شدنی دست یافته‌اند.

با توجه به نمودارهای مذکور، الگوریتم پیشنهادی تنها در یک مورد نسبت به الگوریتم ژنتیک از عملکرد اندکی ضعیف‌تر برخوردار است (در تحلیل حساسیت نسبت به پارامتر $Imax_0$ در حالت نخست) و در سایر موارد نسبت به دو الگوریتم معیار از عملکرد مناسب‌تری برخوردار است. بنابراین، در کل می‌توان عملکرد بهتر الگوریتم پیشنهادی را در ساختارهای مختلف مسایل نمونه نتیجه گرفت.

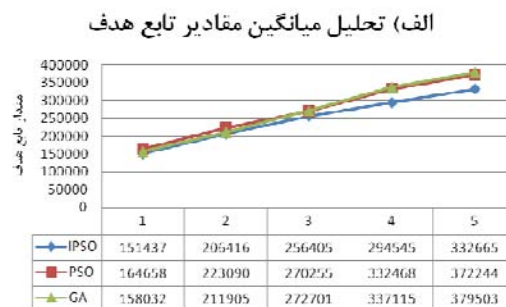
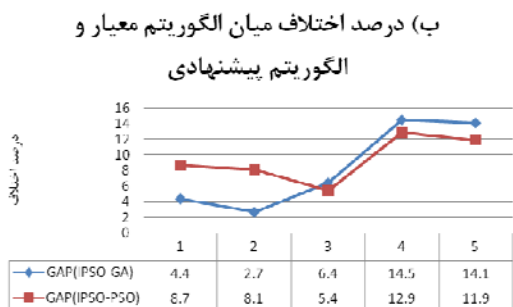
الف) تحلیل میانگین مقادیر تابع هدف



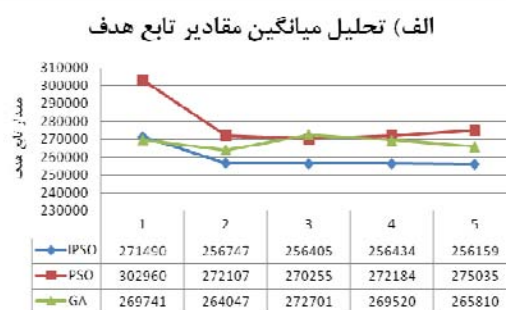
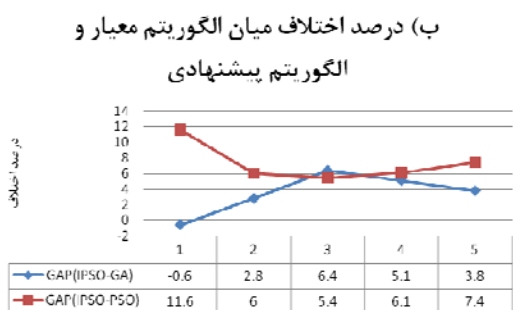
ب) درصد اختلاف میان الگوریتم معیار و الگوریتم پیشنهادی



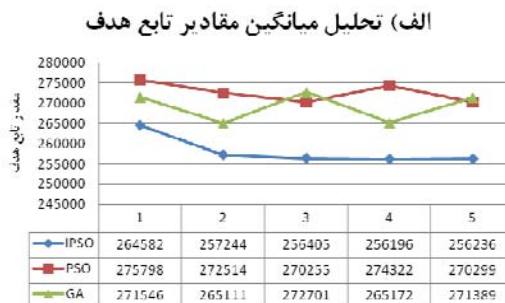
شکل ۱۲. تحلیل حساسیت نسبت به پارامتر Q



شکل ۱۴. تحلیل حساسیت نسبت به پارامتر f_i



شکل ۱۵. تحلیل حساسیت نسبت به پارامتر $Imax_0$



شکل ۱۶. تحلیل حساسیت نسبت به پارامتر $Imax_r$

۵- نتیجه گیری

فرض شده است که چند محصول در یک کارخانه ساخته و با استفاده از ناوگانی از وسایل حمل همسان با ظرفیت محدود برای مجموعه‌ای از خرده‌فروشان با شیوه توزیع مستقیم، ارسال می‌شود. همچنین، هر خرده‌فروش در هر دوره تنها باید توسط یک وسیله حمل و یکبار خدمت‌دهی شود. تولیدکننده و

این مقاله به بررسی مسأله مسیریابی-موجودی در حالت چند دوره‌ای چند محصولی و با هدف کمینه‌سازی مجموع هزینه‌های سیستم شامل هزینه‌های راه‌اندازی، توزیع و نگهداری موجودی پرداخته است. در مسأله بررسی شده در این مقاله،

- Research*, 101(3), 509-518.
- Bitran, G. R., & Yanasse, H. H. (1981). *Computational Complexity of the Capacitated Lot Size Problem*. Cambridge, MA: Massachusetts Institute of Technology.
- Boudia, M., Louly, M. A. O., & Prins, C. (2007). A Reactive Grasp and Path Relinking for a Combined Production-Distribution Problem. *Computers & Operations Research*, 34(11), 3402-3419.
- Boudia, M., & Prins, C. (2009). A Memetic Algorithm with Dynamic Population Management for an Integrated Production-Distribution Problem. *European Journal of Operational Research*, 195(3), 703-715.
- Campbell, A. M., & Hardin, J. R. (2005). Vehicle Minimization for Periodic Deliveries. *European Journal of Operational Research*, 165(3), 668-684.
- Cheng, L., & Duran, M. A. (2004). Logistics for World-Wide Crude Oil Transportation Using Discrete Event Simulation and Optimal Control. *Computers & Chemical Engineering*, 28(6-7), 897-911.
- Cordeau, J.-F., Laporte, G., Savelsbergh, M. W. P., & Vigo, D. (2007). Vehicle Routing. In C. Barnhart & G. Laporte (Eds.), *Handbook in Operations Research and Management Science* (Vol. 14, Transportation, (367-428).
- Engelbrecht, A. P. (2005). *Fundamentals of Computational Swarm Intelligence*. West Sussex, England: John Wiley & Sons, Ltd.
- Esparcia-Alcazar, A. I., Cardos, M., & Merelo, J. J. (2007). *Configuring an Evolutionary Tool for the Inventory and Transportation Problem*. Paper presented at the GECCO'07, London, England, United Kingdom.
- Hu, X., Shi, Y., & Eberhart, R. (2004). *Recent Advances in Particle Swarm*. Paper presented at the Congress on Evolutionary Computation, CEC2004.
- Jiao, B., Lian, Z., & Gu, X. (2006). A Dynamic Inertia Weight Particle Swarm Optimization Algorithm. *Chaos, Solitons and Fractals*, 37(3), 698-705.
- Kennedy, J., & Eberhart, R. C. (1995). *Particle Swarm Optimization*. Paper presented at the IEEE Conference on Neural Networks, Piscataway, NJ.
- خرده‌فروشان به نگهداری موجودی با ظرفیت محدود می‌پردازند، ظرفیت تولید محدود است و کمبود مجاز نیست. به دلیل پیچیدگی محاسباتی بالای مسأله، استفاده از روش‌های حل دقیق به ویژه برای مسایل با ابعاد بزرگ در زمان محاسباتی معقول امکان‌پذیر نیست. بنابراین، در این مقاله الگوریتم حل تقریبی کارایی مبتنی بر الگوریتم بهینه‌سازی گروه ذرات توسعه داده شده است. کارایی الگوریتم پیشنهادی با استفاده از مسایل نمونه متعددی که به صورت تصادفی ایجاد شده‌اند با الگوریتم‌های ژنتیک و بهینه‌سازی گروه ذرات مقایسه شده است. نتایج عددی بیانگر عملکرد بهتر الگوریتم پیشنهادی است.
- در تحقیقات آتی، توسعه مسأله مذکور برای سایر شرایطی که می‌تواند در صنعت وجود داشته باشد، بسیار مفید خواهد بود. در این راستا پیشنهاد می‌گردد مسأله برای حالتی که کمبود مجاز باشد یا امکان ارسال چندین وسیله حمل در هر دوره زمانی برای یک خرده‌فروش وجود داشته باشد، توسعه داده شود.

منابع

- Andersson, H., Hoff, A., Christiansen, M., Hasle, G., & Løkketangen, A. (2010). Industrial Aspects and Literature Survey: Combined Inventory Management and Routing. *Computers & Operations Research*, 37(9), 1515-1536.
- Aziz, N. A. B., & Moin, N. H. (2007). *Genetic Algorithm Based Approach for the Multi Product Multi Period Inventory Routing Problem*. Paper presented at the Proceedings of the 2007 IEEE IEEM, Singapur.
- Barnes-Schuster, D., & Bassok, Y. (1997). Direct Shipping and the Dynamic Single-Depot/Multi-Retailer Inventory System. *European Journal of Operational*

- Kleywegt, A. J., Nori, V. S., & Savelsbergh, M. W. P. (2002). The Stochastic Inventory Routing Problem with Direct Deliveries. *Transportation Science*, 36(1), 94-118.
- Lenstra, J. K., & Rinnooy, K. A. H. G. (1981). Complexity of vehicle routing and scheduling problems. *Networks*, 11, 221-227.
- Li, J., Chen, H., & Chu, F. (2010). Performance Evaluation of Distribution Strategies for the Inventory Routing Problem. *European Journal of Operational Research*, 202 (2), 412-419.
- Li, J.-A., Wu, Y., Lai, K. K., & Liu, K. (2008). Replenishment Routing Problems between a Single Supplier and Multiple Retailers with Direct Delivery. *European Journal of Operational Research*, 190(2), 412-420.
- Moin, N. H., Salhi, S., & Aziz, N. A. B. (2010). An Efficient Hybrid Genetic Algorithm for the Multi-Product Multi-Period Inventory Routing Problem. *International Journal of Production Economics*, In Press, Corrected Proof.
- Rusdiansyah, A., & Tsao, D.-b. (2005). An Integrated Model of the Periodic Delivery Problems for Vending-Machine Supply Chains. *Journal of Food Engineering*, 70(3), 421-434.
- Vehicle routing data sets. from: <http://www.coin-or.org/SYMPHONY/branchandcut/VRP/data>.
- Zhao, Q.-H., Chen, S., & Zang, C.-X. (2008). Model and Algorithm for Inventory/Routing Decision in a Three-Echelon Logistics System. *European Journal of Operational Research*, 191(3), 623-635.
- Zhao, Q.-H., Wang, S.-Y., & Lai, K. K. (2007). A Partition Approach to the Inventory/Routing Problem. *European Journal of Operational Research*, 177(2), 786-802.