مجله تحقیق در عملیات در کاربردهای آن، سال پانزدهم، شماره چهارم (پیاپی ۵۹)، زمستان ۹۷، ص ص۳۵-۱۵
شاپا ۷۲۸۶-۲۲۵۱

حل مساله حمل و نقل با هزینه ثابت تحت شرایط فازی با استفاده از الگوریتم‌های فراابتکاری به‌همراه یک روش جدید نمایش جواب

آتنا شعبانی[۱]، رضا توکلی مقدم[۲*]، مصطفی حاجی آقایی کشتلی[۳]
۱-دانش آموخته کارشناسی ارشد مهندسی صنایع، دانشگاه آزاد اسلامی، واحد علوم و تحقیقات، تهران
۲- استاد دانشکده مهندسی صنایع، پردیس دانشکده های فنی، دانشگاه تهران، تهران
۳- استادیار گروه مهندسی صنایع، دانشگاه علم و فناوری مازندران، بهشهر

رسید مقاله: ۱۳ شهریور ۱۳۹۵
پذیرش مقاله: ۷ تیر ۱۳۹۷

چکیده

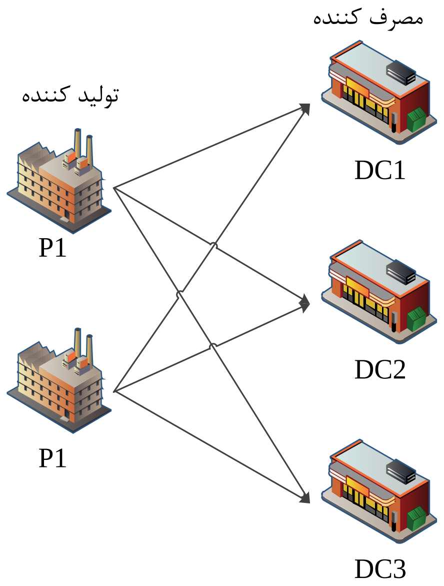
یکی از پراهمیت‌ترین مسایل در زنجیره‌تامین، مساله‌ی حمل و نقل است و یکی از شاخه‌های مساله‌ی حمل و نقل، مساله‌ی حمل و نقل هزینه ثابت است. هدف این مقاله، توسعه‌ی یک روش مفید و کارامد برای حل این مساله است. برای حل این مساله با داده‌های فازی، از الگوریتم‌های مختلفی از قبیل الگوریتم ژنتیک، شبیه‌سازی شده تبرید و کرم شب‌تاب، استفاده می‌شود و در ادامه یک روش جدید نمایش جواب در الگوریتم پیشنهادی ارایه می‌شود. با به کارگیری طراحی آزمایش تاگوچی پارامترهای موجود در الگوریتم‌های پیشنهادی تنظیم شده و بهترین حالت برای هر یک از پارامترها مشخص می-گردد و سپس کارایی الگوریتم‌ها مورد بررسی قرار می‌گیرد. در نهایت جهت انجام مقایسات بین سه الگوریتم پیشنهادی، چند مساله آزمایشی در دو مقیاس کوچک و بزرگ تولید می‌شود. نتایج نشان داده می‌شود که الگوریتم ژنتیک علاوه بر اینکه مقدار تابع هدف کم‌تری دارد، به مراتب زمان محاسباتی کم‌تری را برای رسیدن به جواب نزدیک به بهینه صرف می‌کند.

کلمات کلیدی: مساله حمل و نقل با هزینه ثابت، الگوریتم‌های فراابتکاری، محیط فازی، طراحی آزمایش تاگوچی

۱ مقدمه

مساله حمل و نقل، مساله انتقال کالا از چندین مبدا یا تولید کننده به چندین مقصد یا مصرف‌کننده با حداقل هزینه است. در این نوع خاص از مسایل برنامه‌ریزی خطی، کالاها تنها اجازه دارند که مستقیما از مبدا به مقصد جابه‌جا شوند. در این مسایل ظرفیت ایستگاه‌های مبدا محدود است و تقاضای ایستگاه‌های مقصد و هزینه انتقال هر واحد کالا از مبدا خاص به مقصد خاص مشخص می‌باشد. از مدل حمل و نقل علاوه بر مسایل توزیع کالا در مسایل مربوط به تعیین مکان و برنامه‌ریزی تولید نیز استفاده می‌گردد.

* عهده‌دار مکاتبات
آدرس الکترونیکی: tavakoli@ut.ac.ir



شکل ۱. شبکه زنجیره‌تامین یک مرحله‌ای

طی چند دهه انواع مختلفی از مدل‌های قطعی و غیر قطعی برای مسایل حمل و نقل[1] (TP) پیشنهاد داده شد. فرضیه پایه در هر مساله حمل و نقل این است که هزینه، ارتباط مستقیمی با تعداد واحدهای حمل و نقل دارد، در حالی که در اغلب کاربردهای جهان واقعی، مخصوصاً در تولید- توزیع، یک هزینه ثابت برای توسعه امکانات یا تحقق تقاضای مشتری از هر منبع مورد توجه قرار گرفته است. بسیاری از مسایل کاربردی حمل و نقل و مراکز توزیع با هزینه ثابت در لجستیک، می‌تواند به صورت مسایل حمل و نقل هزینه ثابت[2] (FCTP) فرموله شود.

مساله FCTP توسعه‌ای از مساله عمومی حمل و نقل است که در آن تعدادی از یک محصول برای ارضای تقاضا به محل های تقاضا حمل می‌شود در حالی که هزینه ثابتی علاوه بر هزینه متغیر قبلی اعمال می‌شود.

در مساله FCTP اگر در مسیری کالایی حمل نشود هزینه صفر است؛ ولی در صورت حمل کالا دو نوع هزینه وجود دارد، برای هر مسیر حمل و نقل در صورت استفاده از آن مسیر، هزینه ثابتی مستقل از میزان محصولات حمل شونده وجود دارد و همچنین هزینه متغیری متناسب با میزان محصولات حمل شونده در نظر گرفته می‌شود. هدف پیدا کردن ترکیبی از مسیرهایی است که کل هزینه های ثابت و متغیر را به حداقل می‌رساند؛ در حالی که عرضه مورد نیاز هر منبع و تقاضای مورد نیاز هر مقصد را ارضا می‌کند. حل FCTP به علت هزینه‌های ثابت که منجر به ناپیوستگی در تابع هدف می‌شود به مراتب مشکل‌تر از TP است؛ به طوری که نمی‌توان آن را توسط الگوریتم‌های حمل نقل و به شیوه‌ی مستقیم حل کرد. به‌دلیل استفاده از داده‌های نزدیک‌تر به مسایل دنیای واقعی در مسایل تولید شده، در این مقاله، فرض بر این است که در ابتدا داده‌ها (هزینه‌های تابع هدف) به صورت فازی تولید می‌شود و پس از تبدیل آن‌ها به عدد قطعی، از آن‌ها در مثال‌های ایجاد شده استفاده می‌گردد.

[1] Transportation Problem
[2] Fixed Charge Transportation Problem

۲ مرور ادبیات و پیشینه موضوع

برای اولین بار، بالینسکی [۱] مساله حمل و نقل با هزینه ثابت را فرمول‌بندی کرده و برای حل آن روش تقریبی پیشنهاد کرد. حل مسایل FCTP، از دسته‌ی غیر چندجمله ای سخت می‌باشد و زمان محاسباتی حل نسبت به سایز مساله به‌صورت نمایی افزایش می‌یابد. همچنین طی دو دهه اخیر، چندین روش، برای نمایش مسایل حمل و نقل به وسیله محققان پیشنهاد شده است. پنج روش شناخته شده و پرکاربرد عبارتند از: بر اساس ماتریس[۱] درخت حمل و نقل مستقیم[۲] بر اساس جواب‌های شدنی[۳] درخت پوشا و عدد پرافر[۴] بر اساس اولویت[۵]. کیم و پاردالوس [۲] با اقتباس از یک دیدگاه اقتصادی، یک روش برای حل مساله حمل و نقل با هزینه ثابت، با جریان شبکه ظرفیت‌دار یا بدون ظرفیت پیشنهاد کردند. گِن و چانگ [۳] الگوریتم ژنتیکی بر پایه ی درخت پوشا[۶] برای حل این‌گونه مسایل با استفاده از روش کدگذاری عدد پرافر استفاده کردند. پرافر نامبر یکی از روش‌های کدگذاری برای درخت پوشا می‌باشد. این روش به عنوان یکی از روش‌های کارآمد در مسایل شبکه‌ای شناخته شده است که اولین بار توسط گِن و چانگ [۴] ارایه شده است. ادلاکها و کوالسکی [۵] یک الگوریتم ساده برای حل در مقیاس کوچک پیشنهاد دادند و اظهار کردند که این روش اغلب اوقات کاربرد بیش‌تری نسبت به الگوریتم‌های حل مساله‌ی حمل و نقل معمولی دارد. گن و همکاران [۶] مساله حمل و نقل را با هزینه ثابت با دو مرحله در نظر گرفتند و از الگوریتم ژنتیک با استفاده از رمزگذاری بر اساس اولویت استفاده کردند و روش جدید برای طراحی عملگرها با نام WMX[۷] ارایه دادند. یانگ و لیو [۷] مساله‌های هزینه ثابت حمل ونقل را با داده های فازی بررسی کردند که در آن هزینه‌های متغیر و ثابت، منابع و تقاضاها و ظرفیت‌های حمل همگی به صورت فازی مطرح شده‌اند. یک الگوریتم هوشمند ترکیبی بر اساس روش شبیه‌سازی و الگوریتم جستجوی ممنوعه برای حل آن پیشنهاد گردید. جو و همکاران [۸] برای حل این مسایل از رویه درخت پوشا در الگوریتم ژنتیک استفاده کردند و همچنین روشی را برای اصلاح و شدنی بودن کروموزم های تولید شده ارایه دادند. کوالسکی و لو [۹] مساله حمل و نقل هزینه ثابت مرحله‌ای را مطرح کردند که در آن هزینه ثابت به صورت تابع پله‌ای وابسته به ظرفیت مسیر داده شده است. آن‌ها یک الگوریتم محاسباتی ساده برای حل مسایل دو مرحله‌ای در ابعاد کوچک ارایه دادند. جاوهَر و بلاجی [۱۰] با در نظر گرفتن هزینه حمل و هزینه ثابت مربوط به هر مسیر توزیع یا انبار، یک الگوریتم ژنتیک برای حل پیشنهاد کردند. بلاجی و جاوهَر [۱۱] الگوریتم شبیه‌سازی تبرید را برای حل مساله پیشنهادی خود ارایه کردند.

حاجی آقائی کشتلی و همکاران [۱۲] از الگوریتم ژنتیک بر پایه درخت پوشا و پرافر نامبر برای حل مساله استفاده کرده و همچنین به تولید کروموزوم‌های شدنی بدون نیاز به اصلاح و تصحیح رویه‌ی تبدیل کروموزوم به درخت حمل و نقل که پژوهشگران قبلی جو و همکاران [۸] ارایه کرده بودند، پرداختند. ملاعلیزاده و حاجی

[1] Matrix-based Representation
[2] Direct Transportation Tree Representation
[3] Basic Feasible Solution Representation
[4] Prüfer Number
[5] Priority-based Representation
[6] Spanning Tree
[7] Weight Mapping Crossover

آقائی کشتلی [۱۳] مساله حمل و نقل را با هزینه ثابت به صورت دو مرحله‌ای در نظر گرفتند و دو الگوریتم فراابتکاری ایمنی مصنوعی[1] (AIA) و ژنتیک را بر اساس درخت پوشا و عدد پروفر ارایه دادند. همچنین برای تنظیم پارامترها و عملگرهای الگوریتم‌های پیشنهادی، از روش طراحی پارامتر تاگوچی استفاده کردند. لطفی و توکلی مقدم [۱۴] برای مساله خطی و غیر خطی حمل و نقل با هزینه ثابت، یک الگوریتم ژنتیک با استفاده از رمزگذاری بر اساس اولویت انجام داده اند که در آن عملگرهای جدید ارایه شده است. آن‌ها مسایل FCTP را با دو روش ژنتیک بر اساس اولویت و ژنتیک بر اساس درخت پوشا مقایسه کردند. ملاعلیزاده و همکاران [۱۵] مساله حمل و نقل را با هزینه ثابت در محیط فازی در نظر گرفتند و سه الگوریتم فراابتکاری جست و جوی همسایگی VNS[2]، تبرید شبیه سازی شده SA و جست و جوی همسایگی تلفیقی[3] ارایه دادند. اِل شِربینی و همکاران [۱۶] ترکیبی از الگوریتم های ازدحام ذرات و ایمنی مصنوعی را به نام HPSIL[4] برای حل به کار برده‌اند و همچنین در الگوریتم مذکور از یک ساختار ذره انعطاف‌پذیر، روش رمزگشایی و تخصیصی جدید به جای بهره‌مندی از عدد پروفر استفاده شده است. ملاعلیزاده و همکاران [۱۷] مساله‌ی حمل و نقل را با هزینه ثابت پله ای در نظر گرفتند و از دو الگوریتم ژنتیک و ممتیک که بر پایه درخت پوشا رمزگشایی شده بود استفاده کردند. آن ها با استفاده از نمونه‌های زیادی به مقایسه دو الگوریتم از نظر کیفیت جواب نهایی پرداختند. کریستف کوالسکی [۱۸] با توسعه یک الگوریتم ابتکاری به جواب بهینه سراسری برای حل مسایل حمل و نقل با هزینه ثابت در مقیاس کوچک رسید. در روش ارایه شده مسایل به مجموعه‌ای از مسایل کوچک‌تر تجزیه می‌شود و در زمان کم‌تری جواب بهینه ارایه می‌شود. شیرازی و همکاران [۱۹] مساله حمل و نقل را با هزینه ثابت در سه مرحله با هزینه متغیر، هزینه‌ی ثابت و هزینه ی کمبود در نظر گرفتند و این مساله را به صورت برنامه‌ریزی عدد صحیح مختلط فرموله و از تصمیم‌گیری چند معیاره برای حل آن استفاده کردند. خورانا و ادلکها [۲۰] الگوریتمی برای حل مساله حمل و نقل چند شاخصه با هزینه ثابت دو معیاره پیشنهاد دادند و همچنین روشی برای پیدا کردن موازنه بهینه در میان موازنه زمان–هزینه کارآمد ارایه کردند. الگوریتم پیشنهاد شده متغیرها و هزینه‌های ثابت را همزمان کم می‌کند تا یک جواب پایه بهینه شدنی به دست آید.

۳ مقدمات فازی

رتبه‌بندی زیرمجموعه‌های فازی در مساله حمل و نقل بسیار مهم است. از آن جایی‌که تابع هدف، کمینه کردن هزینه حمل و نقل فازی می‌باشد، ما از یک روش ساده و انعطاف‌پذیر رتبه بندی اعداد فازی با مقدار صحیح استفاده کردیم که توسط لیو و ونگ [۲۱] توسعه داده شده است. بر اساس این روش، مقدار انتگرال کل یک ترکیب محدب از مقدار انتگرال راست و چپ اطراف یک شاخص بهینه است، $\alpha \in [0,1]$. مقدار انتگرال چپ برای بازتاب کردن نقطه نظر خوش‌بینانه و مقدار انتگرال راست برای بازتاب کردن نقطه نظر بدبینانه برای مدیر

[1] Artificial Immune Algorithm
[2] Variable Neighborhood Search
[3] Hybrid Variable Neighborhood
[4] Hybrid Particle Swarm Algorithm with Artificial Immune Learning

استفاده می‌شود. به ترکیب محدب از مقدار انتگرال راست و چپ اطراف یک شاخص بهینه، مقدار انتگرال کل گفته می‌شود که برای رتبه‌بندی اعداد فازی استفاده می‌شود.

عدد فازی مثلثی: عدد فازی مثلثی، یک عدد فازی است که با سه عدد حقیقی به صورت $A=(a_1,a_2,a_3)$ و یک تابع عضویت $\mu_A(x)$ نمایش داده می‌شود.

$$\mu_A(x)\begin{cases} \dfrac{x-a_1}{a_2-a_1} & a_1 \le x \le a_2 \\ \dfrac{a_3-x}{a_3-a_2} & a_2 \le x \le a_3 \\ 0 & otherwise \end{cases} \tag{1}$$

تابع معکوس مشابه $\mu_A(x)$ به صورت زیر است:

$$g_A(y)^L = a_1 + (a_2 - a_1) \tag{2}$$

$$g_A(y)^R = a_3 - (a_3 - a_2)y = a_3 + (a_2 - a_3) \tag{3}$$

جایی که $y \in [0,1]$ است؛ بنابراین مقدار انتگرال چپ و راست از فرمول های زیر به دست می‌آید:

$$I(A)^L = \int_0^1 g_A(y)^L dy = \frac{1}{2}(a_1 + a_2) \tag{4}$$

$$I(A)^R = \int_0^1 g_A(y)^R dy = \frac{1}{2}(a_2 + a_3) \tag{5}$$

مقدار انتگرال کل اعداد فازی مثلثی $A=(a_1,a_2,a_3)$ از معادله زیر به دست می‌آید:

$$I_T^{\alpha}(A) = \alpha I(A)^R + (1-\alpha) I(A)^L = \frac{1}{2}(\alpha a_3 + a_2 + (1-\alpha)a_1) \tag{6}$$

به α درجه خوش‌بینی گفته می‌شود. $\alpha \in [0,1]$

٤ مدل ریاضی FCTP

مدل FCTP را می‌توان به عنوان یک مساله‌ی تخصیص در نظر گرفت که در آن m تامین کننده و n متقاضی وجود دارد. هر یک از آن m تامین‌کننده توانایی ارسال کالا به هریک از n مشتری را دارا می‌باشد. در حالی‌که هزینه‌ی c_{ij} را به ازای هر واحد کالای ارسالی از تامین‌کننده i به مشتری j خواهیم داشت و همچنین علاوه بر این یک هزینه ثابت به ازای فعال شدن مسیر i,j به اندازه f_{ij} در نظر گرفته می‌شود. هریک از تامین‌کننده‌ها به مقدار a_i عرضه و هریک از مشتریان به مقدار b_j تقاضا دارند. تابع هدف مساله، تعیین‌کننده این است که چه

مسیری بازگشایی شود و چه سایزی از کالا از آن مسیر منتقل شود که محدودیت‌ها ارضا شده و هزینه‌ی کل انتقالات کمینه شود. مدل‌سازی استاندارد مساله حمل و نقل هزینه ثابت، به صورت زیر می‌باشد:

$$Min \quad Z=\sum_{i=1}^{m}\sum_{j=1}^{n}(c_{ij}.x_{ij}+f_{ij}.y_{ij}) \tag{7}$$

$$s.t.$$

$$\sum_{j=1}^{n}x_{ij}\le a_i; i=1,2,...,m, \tag{8}$$

$$\sum_{i=1}^{m}x_{ij}\le b_j, j=1,2,...,n, \tag{9}$$

$$x_{ij}\ge 0, \forall i,j, \tag{10}$$

$$x_{ij}=\begin{cases}1 & x_{ij}>0\\ 0 & otherwise\end{cases}, \ \forall i,j. \tag{11}$$

که پارامترهای آن به شکل زیر معرفی می‌شود:

i = شاخص تامین کننده — a_i = میزان عرضه تامین کننده i

j = شاخص مشتری — b_j = میزان تقاضا مشتری j

m = تعداد تامین کننده — x_{ij} = مقدار کالای ارسالی از تامین کننده i به مشتری j

n = تعداد مشتری — c_{ij} = هزینه متغیر حمل و نقل از تامین کننده i به مشتری j

f_{ij} = هزینه ثابت حمل و نقل مربوط به مسیر i,j

مجموعه محدودیت اول (۸) نشان دهنده این مطلب است که مجموع جریان‌های خروجی از مبدا i باید کوچک‌تر یا برابر با ظرفیت آن باشد و همچنین مجموعه محدودیت دوم (۹) نشان‌دهنده این است که مجموع مقادیر دریافتی هر مشتری نمی‌تواند کم‌تر از میزان تقاضایش باشد. متغیر y_{ij}(۱۱) یک متغیر ۰ و۱ است که باتوجه به مقدار x_{ij} تعیین می‌شود. در واقع اگر مقدار x_{ij} بزرگ‌تر از صفر باشد؛ یعنی نیاز به بازگشایی مسیر ij خواهیم داشت و مقدار y_{ij} برابر یک می‌شود، در غیراین‌صورت مقدار آن برابر با صفر است. در واقع، y_{ij} متغیری می‌باشد که فعال شدن یا نشدن یک مسیر را نشان می‌دهد.

تابع هدف فازی پیشنهادی در این مقاله به صورت زیر می‌باشد:

$$Min \ Z=\sum_{i=1}^{m}\sum_{j=1}^{n}(\tilde{c}_{ij}.x_{ij}+f_{ij}.y_{ij}) \tag{12}$$

همان طور که مشاهده می‌شود، فقط داده‌های تابع هدف، اعم از ثابت و متغیر، فازی در نظر گرفته شده‌است.

۵ روش حل

با توجه به غیرچندجمله‌ای سخت بودن مساله که در تحقیقات قبلی به آن اشاره گردید، برای حل این مساله با داده‌های فازی، از الگوریتم‌های مختلفی از قبیل الگوریتم ژنتیک، شبیه‌سازی شده تبرید و کرم شب‌تاب، استفاده

می‌شود. اولین گام در حل مدل مساله، ارتباط دادن آن با ساختار الگوریتم‌ها فراابتکاری است؛ یعنی ساختن یک پل ارتباطی بین مساله‌ی اصلی و فضای حلی که در آن تکامل رخ می‌دهد. در عمل، باید یک روش برای نمایش کروموزوم های شدنی انتخاب شود. به طور کلاسیک، یک کروموزوم شدنی، در مسایل بهینه‌سازی ترکیبی (گسسته)، می‌تواند رشته ای از اعداد صحیح، در مسایل پیوسته برداری، رشته‌ای از اعداد حقیقی، در مسایل بولین، رشته‌ای از ارقام دودویی و حتی در صورت نیاز، ترکیبی از نمایش‌ها باشد؛ بنابراین انتخاب یک روش نمایش مناسب، یکی از مهم‌ترین قسمت‌های طراحی یک الگوریتم است. اکثر الگوریتم‌های تکاملی از رویه‌ی تصادفی برای تولید مجموعه جواب‌های اولیه استفاده می‌کنند.

۵-۱ نحوه نمایش جواب

در این مقاله کروموزوم دو بخشی است. بخش اول، مقدار کالای ارسالی به مشتریان است که به صورت ماتریس $I\times J$ با مقادیر بین صفر و یک می‌باشد که نحوه تفسیر مقدار به صورت زیر است:

$$\sum_{j=1}^{n} x_{ij} = a_i \tag{۱۳}$$

x_{ij}: سهم تامین‌کننده‌ی i در برآورده کردن تقاضای مشتری j

$$0 \le x_{ij} \le 1 \quad x_{ij} = \frac{x_{ij} \times a_i}{\sum_{j=1}^{n} x_{ij}} \tag{۱۴}$$

$$i\left\{\overbrace{\begin{bmatrix} ۰/۴۸ & \cdots & ۰/۷ \\ \vdots & \ddots & \vdots \\ ۰/۵ & \cdots & ۰/۳۵ \end{bmatrix}}^{j}\right.$$

بخش دوم مربوط به تصمیم‌گیری در مورد بازگشایی یک تامین‌کننده برای ارسال کالا به مشتریان می‌باشد که به صورت ماتریس باینری با ابعاد $I\times J$ است که در آن مقدار صفر نشان دهنده ی عدم بازگشایی یک تامین‌کننده و عدد یک نیز نشان دهنده‌ی بازگشایی یک تامین‌کننده برای ارسال کالا به مشتریان می‌باشد.

$$i\left\{\overbrace{\begin{bmatrix} ۱ & \cdots & ۰ \\ \vdots & \ddots & \vdots \\ ۱ & \cdots & ۱ \end{bmatrix}}^{j}\right.$$

۵-۲ الگوریتم ژنتیک (Genetic Algorithm)

براساس نقش ژنتیک در طبیعت و تکامل طبیعی موجودات زنده، جان هالند[۲۲] نوع خاصی از الگوریتم‌های تکاملی را در اوایل دهه‌ی ۷۰ میلادی به نام الگوریتم ژنتیک ارایه داد. الگوریتم ژنتیک، الگوی ریاضی است که

با استفاده از الگوهای عملیاتی داروینی در مورد تکثیر بقای نسل برتر و براساس فرآیند طبیعی ژنتیک، جمعیتی از کروموزوم‌ها را به جمعیت جدید تبدیل می‌کند. ساختار کلی یک الگوریتم ژنتیک را می‌توان چنین تصور کرد که پیش از هر چیز باید مکانیسمی برای تبدیل جواب هر مساله به یک کروموزوم تعریف شود. پس از آن مجموعه‌ای از کروموزوم‌ها، که در حقیقت مجموعه‌ای از جواب‌های مساله هستند، به عنوان جمعیت اولیه در نظر گرفته می‌شوند. بعد از تعریف جواب اولیه باید با به کارگیری عملیات ژنتیک اقدام به ایجاد کروموزوم‌های جدید موسوم به فرزند نمود. این عملیات به دو گونه اصلی تقاطع و جهش تقسیم بندی می‌شود. همچنین برای گزینش کروموزوم‌هایی که باید نقش والدین را بازی کنند، دو مفهوم عملگر تقاطع و عملگر جهش کاربرد فراوان دارند که این عملگر نیز تعریف می‌شود. بعد از ایجاد جمعیت فرزندان باید با استفاده از عمل ارزیابی اقدام به انتخاب بهترین کروموزوم‌ها کرد. فرآیند انتخاب مبتنی بر مقدار برازندگی هر رشته است. در حقیقت فرآیند ارزیابی مهم‌ترین بحث در فرآیند انتخاب است. بر این اساس، پس از تکرار چند نسل، بهترین نسل که همان پاسخ بهینه مساله است، ایجاد خواهد شد.

۵-۲-۱ تولید جواب اولیه

پس از تعریف مساله حمل و نقل با هزینه ثابت و تولید یک راه حل تصادفی، ارزیابی راه حل و شیوه تولید جواب های همسایه، به حل مساله با الگوریتم ژنتیک می‌پردازیم. ابتدا یک جواب اولیه شدنی با استفاده از نحوه نمایش ارایه شده در قسمت قبل به دست می‌آوریم، سپس برای تولید جواب‌های همسایگی پس از انتخاب کروموزوم‌های مناسب از عملگرهای تقاطع و جهش استفاده می‌کنیم.

۵-۲-۲ ارزیابی و مکانیسم انتخاب

در الگوریتم ژنتیک، روش‌های مختلفی برای انتخاب کروموزوم‌ها برای اعمال عملگرهای مختلف وجود دارد. در این مطالعه از انتخاب براساس نخبه‌گرایی استفاده شده است. در این روش، بعد از به‌دست آوردن مقدار شایستگی هر کدام از کروموزوم‌ها، با توجه به اینکه مساله مینیمم‌سازی است، کروموزومی که دارای کم‌ترین مقدار تابع شایستگی است، به عنوان رتبه اول در نظر گرفته می‌شود و بدین ترتیب، با توجه به تعداد کروموزوم‌های جمعیت، رتبه‌بندی براساس افزایش مقدار تابع شایستگی صورت می‌گیرد، سپس به تعداد دلخواه (m)، از رتبه اول تا رتبه m ام کروموزوم‌ها انتخاب می‌گردد.

۵-۲-۳ عملگرهای ژنتیک

عملگر، اصطلاحی برای سازوکارها یا پردازش‌های صورت گرفته در الگوریتم ژنتیک است. همان‌طور که در قسمت قبل گفته شد عملگر انتخاب، برای پالایش جمعیت جاری مورد استفاده واقع می‌شود. به این معنی که جواب های خوب تکثیر و جواب‌های ضعیف حذف می‌شود. عملگر تقاطع فرآیندی است که براساس آن دو

جواب والد از جمعیت پالایش شده در نظر گرفته می‌شود و از آن‌ها یک یا چند فرزند ایجاد می‌گردد. عملگر جهش نیز از گرفتار شدن الگوریتم در دام نقاط بهینه محلی جلوگیری می‌کند.

۵-۲-۳-۱ تقاطع

تقاطع یک نقطه: تقاطع یک نقطه بدین گونه است که ابتدا دو رشته را به صورت تصادفی انتخاب می‌کند. سپس محلی را برای عمل تقاطع به صورت تصادفی مشخص و سرانجام مقدار دو رشته را با توجه به محل تقاطع جا به جا می‌کند.

تقاطع دو نقطه: در این روش دو نقطه را به صورت تصادفی به عنوان نقطه برش، در طول دو کروموزومی که به عنوان والدین انتخاب شده‌اند در نظر گرفته و کروموزوم ها از آن دو نقطه به سه قسمت تقسیم می‌شوند، که در این تقسیم‌بندی، قسمت وسط ثابت مانده و قسمت‌های دو طرف از دو کروموزوم با هم تعویض می‌شوند.

۵-۲-۳-۲ جهش

عملگر جهش بر روی یک یا چند ژن کروموزوم که به‌طور تصادفی انتخاب شده تغییراتی ایجاد می‌کند و مقدار آن را تغییر می‌دهد. می‌توان استنباط کرد که مهم‌ترین وظیفه جهش اجتناب از همگرایی به بهینه محلی است و به تعبیری می‌توان جهش را مشابه شروع مجدد تصادفی الگوریتم دانست. بسیاری از مسایل بهینه‌سازی ترکیبی، به طور طبیعی، به صورت جایگشتی از n جز فرموله می‌شوند. روش‌های تغییر کروموزوم در این‌گونه مسایل به روش‌های تولید جواب همسایگی نیز معروف هستند. در این مطالعه از سه نوع جهش تعویض، معکوس‌سازی و تغییر مکان استفاده شده است.

۵-۳ الگوریتم شبیه‌سازی تبرید (Simulated Annealing Algorithm)

الگوریتم شبیه‌سازی تبرید (*SA*) را کرک پاتریک و همکاران [۲۳] معرفی کردند. ایده‌ی اولیه این الگوریتم، برای اولین بار توسط متروپلیس [۲۴] ارایه گردید که براساس فرآیند سرمایش یا تبرید مواد در علم ترمودینامیک آماری مطرح شد. در الگوریتم شبیه‌سازی تبرید، پاسخ‌های پیشنهادی برای مساله در دمای بالاتر قرار دارند و غالبا پاسخ های نامناسبی هستند، سپس متغیری که نقش دما را بر عهده دارد به مرور زمان کاهش (افزایش تکرارها) داده می‌شود تا به این ترتیب پاسخ‌های بهتری در دماهای پایین تشکیل شوند. کاهش دما در فرآیند شبیه‌سازی تبرید شبیه کاهش مقدار تابع هدف (در مسایل کمینه‌سازی) است که یک سری تغییرات بهبود دهنده آن را ایجاد می‌کند. برای آنکه اجازه داده شود دما به آهستگی کاهش یابد، باید تغییرات غیر بهبود دهنده تابع هدف نیز با احتمال معینی پذیرفته شود، به طوری که با افزایش تکرارها (کاهش دما) این احتمال نیز کاهش یابد. این امر سبب می‌شود الگوریتم در دام بهینه موضعی نیفتد؛ بنابراین، دما در مسایل بهینه‌سازی به عنوان یک پارامتر کنترلی عمل می‌کند. به همین منظور، الگوریتم شبیه‌سازی تبرید جزء روش‌های جستجوی موضعی (همسایگی) است که به

دلیل پذیرش پاسخ‌های غیر بهبود دهنده تابع هدف، بر خلاف سایر روش‌های جستجوی موضعی به نقطه‌ی شروع (پاسخ اولیه) وابسته نیست و می‌تواند از دام بهینه‌های موضعی تا حد زیادی رهایی یابد.

۵-۳-۱ نحوه پیاده‌سازی

پس از تعریف مساله حمل و نقل با هزینه ثابت و تولید یک راه حل تصادفی، ارزیابی راه حل و شیوه تولید جواب های همسایه به حل مساله با الگوریتم شبیه‌سازی تبرید می‌پردازیم. در بخش تعریف پارامترهای الگوریتم شبیه-سازی تبرید تعداد تکرار حلقه اصلی را برابر ۵۰۰، مقدار دمای اولیه نیز $T_{\circ}=۱۰۰$ و ضریب کاهش دما در هر تکرار $\alpha=۰/۹۹$ در نظر گرفتیم. سپس از تابع احتمال $p=e^{\frac{-\Delta E}{T}}$ برای پذیرش جواب‌های بدتر و خروج از بهینه محلی استفاده کردیم. سپس از الگو خطی $T_n=\alpha\times T_{n-۱}$ پارامتر دما را در هر تکرار کاهش دادیم. شرایط خاتمه‌ی متفاوتی می‌تواند برای یک الگوریتم شبیه‌سازی تبرید به کار رود که از جمله آن‌ها می‌توان به تعداد دفعات کاهش سطح دما، تعداد دفعات متوالی که در تابع هدف بهبودی مشاهده نشود، رسیدن به دمای پایانی مورد نظر و تعداد تکرار الگوریتم اشاره کرد. که ما برای خاتمه الگوریتم از تعداد دفعات تکرار برابر ۵۰۰استفاده کردیم.

۵-٤ الگوریتم کرم شب‌تاب (Firefly Algorithm)

الگوریتم کرم شب‌تاب یک مدل تکاملی است که بر گرفته از طبیعت و مبتنی بر الگوریتم‌های هوش جمعی می‌باشد. این الگوریتم اولین بار توسط آقای یانگ [۲۵] در دانشگاه کمبریج مطرح گردید.

۵-٤-۱ ساختار الگوریتم

برای ساده سازی الگوریتم کرم شب‌تاب سه فرض زیر در نظر گرفته شده است:

تمام کرم‌های شب‌تاب تک جنسی هستند؛ بنابراین هر کرم شب‌تاب مستقل از جنسیت سایر کرم‌های شب‌تاب می‌تواند به سوی آن ها جذب شود. میزان جذابیت هر کرم شب‌تاب (برای سایر کرم های شب‌تاب مورد استفاده در الگوریتم FA) با شدت نور دریافت شده از آن کرم شب‌تاب توسط هر یک از دیگر کرم‌های شب‌تاب گروه متناسب است. شدت نور هر یک از کرم‌های شب‌تاب توسط مقدار به دست آمده برای تابع هدف در آن نقطه تعیین می‌شود. نحوه تعریف تابع شدت نور باید به گونه‌ای باشد که میزان درخشندگی هر کرم شب‌تاب به طور یکنواخت با افزایش بهینگی آن افزایش یابد.

با استفاده از سه قانون فوق، الگوریتم FA برای حل مسایل بهینه‌سازی پیوسته به صورت زیر پیشنهاد می‌شود:

۱. مقداردهی اولیه به پارامترها. مقادیر مناسبی را به پارامترهای γ، MaxGeneration، α، $\beta_{\circ}$، m نسبت دهید که در آن

γ: ضریب جذب نور توسط محیط

MaxGeneration : حداکثر تعداد تکرارها (شرط خاتمه اجرای الگوریتم)

$\beta_{\circ}$: حداکثر ضریب جذابیت بین دو کرم شب‌تاب

α: ضریب بردار جابجایی تصادفی

m : تعداد کرم های شب‌تاب

است. متغیر t (شمارشگر تعداد تکرارهای انجام شده) را نیز برابر با یک قرار دهید.

۲. مقداردهی اولیه به کرم‌های شب‌تاب. جمعیت اولیه‌ای از کرم‌های شب‌تاب (یعنی مجموعه‌ای از بردارها به صورت $x_i, (i = ۱, ۲, ..., m)$ را به طور تصادفی در دامنه مساله تولید کنید.

۳. شدت نور iامین کرم شب‌تاب، I_i، را با استفاده از مقدار به دست آمده برای تابع هدف در این نقطه، $f(x_i)$ ، تعیین کنید.

۴. اگر t<MaxGeneration باشد آن‌گاه به قدم ۵ و در غیر این صورت به قدم ۱۱ بروید.

۵. به ازای $i : m = ۱$ مراحل ۶ و۷ را انجام دهید.

۶. به ازای $j : m = ۱$ مرحله ۷ را انجام دهید.

۷. اگر $I_j > I_i$ باشد آن‌گاه کرم شب‌تاب iام را به سوی کرم شب‌تاب j ام حرکت دهید. برای این منظور موقعیت کرم شب‌تاب iام را با استفاده از معادله زیر به‌روزرسانی کنید:

$$x_i \leftarrow x_i + \beta_{\circ} e^{-\gamma r_{ij}^{۲}} (x_j - x_i) + \alpha \in i \quad (۱۵)$$

که در آن r_{ij} فاصله اقلیدسی بین دو کرم شب‌تاب است. پس از محاسبه موقعیت جدید کرم شب‌تاب iام، شدت روشنایی آن را به روز رسانی کنید.

۸. موقعیت بهترین (یا در واقع، بهینه‌ترین) کرم شب‌تاب را به طور تصادفی تغییر دهید. برای این منظور فرض کنید x_{opt} بیانگر موقعیت بهینه‌ترین کرم شب‌تاب به دست آمده در مراحل ۶ و۷ باشد. برای جابه‌جایی این کرم شب‌تاب به طور تصادفی می‌توان x_{opt} را با یک بردار تصادفی هم طول جمع کرد به طوری که بردار حاصل از جمع این دو همچنان در دامنه تعریف مساله قرار داشته باشد.

۹. کرم های شب‌تاب را بر اساس مقادیر به دست آمده برای تابع هزینه مرتب‌سازی کنید و بهترین جواب به دست آمده را تعیین نمایید.

۱۰. به قدم ۴ بروید.

۱۱. بهترین جواب به دست آمده در طول تمام تکرارها را به عنوان جواب مساله بهینه‌سازی معرفی کنید.

۵-٤-۲ جاذبه

در الگوریتم کرم شب‌تاب دو مساله اساسی وجود دارد. یکی تفاوت در شدت نور و دیگری فرموله کردن میزان جاذبه است. برای سادگی می‌توان فرض کرد که جاذبه کرم شب‌تاب با درخشندگی‌اش مشخص می‌گردد که به تابع هدف بستگی دارد. از آنجا که جاذبه کرم شب‌تاب با شدت نور کرم شب‌تاب مجاور متناسب است، می‌توان جاذبه β از یک کرم شب‌تاب را بدین صورت تعریف نمود:

$$\beta = \beta_{\circ} e^{-\gamma r_{ij}} \tag{۱۶}$$

۵-٤-۳ فاصله و حرکت

فاصله بین هر دو کرم شب‌تاب i و j در نقطه x_i و x_j می‌تواند به صورت زیر نشان داده شود:

$$r = \left\| x_i - x_j \right\| = \sqrt{\sum_{k=1}^{n} (x_{i,k} - x_{j,k})^2} \tag{۱۷}$$

در عبارت فوق منظور از $x_{i,k}$ مولفه k ام از بردار موقعیت کرم شب‌تاب i ام، x_i است.

حرکت کرم شب‌تاب i و جذب آن به کرم شب‌تاب j که درخشان‌تر است به صورت زیر محاسبه می‌شود:

$$x_i \leftarrow x_i + \beta_{\circ} e^{-\gamma r_{ij}^2} (x_j - x_i) + \alpha \in_i \tag{۱۸}$$

دومین جمله در سمت راست رابطه (۱۸) بیانگر میزان کشش کرم شب‌تاب i ام به سوی کرم شب (پرنورتر) j ام و سومین جمله در سمت راست این معادله بیانگر میزان عدم قطعیت این کشش است.

٦ طراحی آزمایش

تاگوچی [۲۵] یک روش جدید برای طراحی آزمایش‌ها ارایه کرد. تاگوچی به عنوان اولین ارایه‌دهنده روش طراحی پارامتر، یک روش مهندسی برای طراحی محصول یا فرآیند ارایه نمود که هدف آن مینیمم‌کردن تغییرات و حساسیت عوامل اغتشاش بود. در یک طراحی پارامتر کارا هدف اول شناسایی و تنظیم فاکتورهای است که تغییرات متغیر پاسخ را به حداقل می‌رساند و هدف بعدی شناسایی فاکتورهای قابل کنترل و غیر قابل کنترل می باشد. هدف نهایی این روش پیدا کردن ترکیب بهینه مقدار فاکتورهای قابل کنترل می‌باشد. تاگوچی مجموعه‌ی ویژه‌ای از طرح‌های کلی برای آزمایش‌های فاکتوریلی ایجاد کرده است که اغلب کاربردها را پوشش می‌دهد. آرایه‌های متعامد، جزیی از این مجموعۀ طراحی‌ها می‌باشد. استفاده از این آرایه‌ها ما را در تعیین کم‌ترین تعداد آزمایش‌های مورد نیاز برای مجموعه‌ای از فاکتورها یاری می کند. طرح‌های آرایه های متعامد به کار رفته در این مطالعه بدین گونه است که از L۹ برای الگوریتم‌های ژنتیک و تبرید شبیه‌سازی شده وL۲۷ برای الگوریتم کرم شب‌تاب استفاده شد. آزمایش‌های مورد نیاز برای بررسی ترکیبات مختلف پارامترها و پاسخ‌های مربوطه در جداول مربوط به هر الگوریتم ارایه شده است. داده‌های تولید شده توسط نرم افزار MINITAB۱۶ تحلیل شده و نتایج به‌دست آمده از آن‌ها در جداول و شکل‌های زیر ذکر شده است.

جدول ۱. پارامترها و سطوح الگوریتم‌ها

پارامترها و سطوح الگوریتم FA		پارامترها و سطوح الگوریتم SA		پارامترها و سطوح الگوریتم GA	
FA Parameters	Levels	SA Parameters	Levels	GA Parameters	Levels
Max_{it}	۱۰۰, ۱۵۰, ۲۰۰	Max_{it}	۳۰۰, ۴۵۰, ۶۰۰	Max_{it}	۲۰۰, ۳۰۰, ۵۰۰
N_{pop}	۲۰, ۴۰, ۶۰	Max_{sl}	۲۰, ۳۰, ۵۰	N_{pop}	۳۰, ۵۰, ۷۰
α	۰/۰۴, ۰/۳, ۰/۲	T_0	۷۰, ۱۰۰, ۱۲۰	P_c	۰/۴, ۰/۵, ۰/۶
β	۱/۵, ۲, ۲/۵	α	۰/۹۳, ۰/۹۶, ۰/۹۹	P_m	۰/۴ , ۰/۳, ۰/۲
γ	۰/۸, ۱, ۱/۲				

برای تعیین اولویت پی درجه اهمیت هرکدام از فاکتورهای الگوریتم GA، سطوح پاسخ با توجه به شاخص‌های میانگین پاسخ‌ها و نسبت‌های *SN* بررسی شده فاکتورها رتبه‌بندی می‌شود. این جداول نشان‌دهنده‌ی مقدار متوسط پاسخ برای هر سطر از هر فاکتور می‌باشند. جداول ۳ و ۴ رتبه‌بندی پارامترها را نشان می‌دهند که در هر دوی آن-ها، اولویت‌بندی پارامترها، مشابه بوده و به ترتیب برابر با نرخ جهش، تعداد جمعیت اولیه، نرخ تقاطع و حداکثر تعداد الگوریتم می‌باشد.

جدول ۲. ترکیبات پیشنهادی روش تاگوچی و سطوح پاسخ به دست آمده برای الگوریتم GA

Max_{it}	N_{pop}	p_c	p_m	Time	Value
۱	۱	۱	۱	۲/۴۴	۲۵۷۳
۱	۲	۲	۲	۴/۳۹	۲۶۲۳/۶۴
۱	۳	۳	۳	۶/۶۶	۲۵۷۳
۲	۱	۲	۳	۵/۱۶	۲۵۷۳
۲	۲	۳	۱	۶/۱۲	۲۵۷۳
۲	۳	۱	۲	۷/۶۸	۲۶۰۲/۶۶
۳	۱	۳	۲	۷/۵۷	۲۵۷۳
۳	۲	۱	۳	۱۱/۱۲	۲۵۷۳
۳	۳	۲	۱	۱۲/۰۶	۲۵۷۳

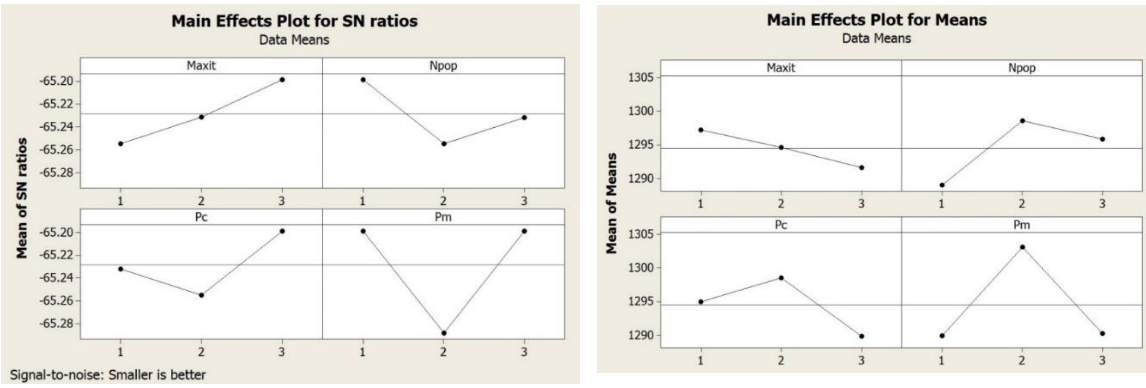
جدول ۳. پاسخ نسبت‌های SN برای الگوریتم GA

Level	Max_{it}	N_{pop}	P_c	P_m
۱	-۶۵/۲۳	-۶۵/۲	-۶۵/۲۳	-۶۵/۲
۲	-۶۵/۲۳	-۶۵/۲۵	-۶۵/۲۵	-۶۵/۲۹
۳	-۶۵/۲	-۶۵/۲۳	-۶۵/۲	-۶۵/۲
Delta	۰/۰۶	۰/۰۶	۰/۰۶	۰/۰۹
Rank	۴	۳	۲	۱

جدول ٤. پاسخ میانگین‌ها برای الگوریتم GA

Level	Max_{it}	N_{pop}	P_c	P_m
۱	۱۲۹۷	۱۲۸۹	۱۲۹۵	۱۲۹۰
۲	۱۲۹۵	۱۲۹۹	۱۲۹۹	۱۳۰۳
۳	۱۲۹۲	۱۲۹۶	۱۲۹۰	۱۲۹۰
Delta	۶	۱۰	۹	۱۳
Rank	۴	۳	۲	۱

برای یافتن سطوح بهینه هر یک از فاکتورها از تحلیل اثرات متقابل آن‌ها استفاده می‌شود. این اثرات در شکل ۲ نمایش داده شده است که با توجه به آن‌ها، پارامتر حداکثر تعداد تکرار الگوریتم $\boldsymbol{Max_{it}}$ در سطح اول خود یعنی ۲۰۰، تعداد جمعیت اولیه $\boldsymbol{N_{pop}}$ در سطح دوم ؛ یعنی ۵۰، نرخ تقاطع $\boldsymbol{P_c}$ در سطح دوم ، یعنی ۰/۵ و در نهایت نرخ جهش $\boldsymbol{P_m}$ در سطح دوم خود یعنی ۰/۳ پاسخ میانگین را کمینه و شاخص SN را حداکثر می‌کند؛ بنابراین، مقادیر پارامترهای کنترلی الگوریتم GA به دست آمده است. برای انجام طراحی آزمایش‌های چند عاملی برای فاکتورهای الگوریتم SA به‌طور مشابه الگوریتم GA عمل می‌کنیم. در الگوریتم SA با توجه به اینکه چهار فاکتور و سه سطح درنظر گرفته شد ترکیبات پیشنهادی روش تاگوچی و نتایج آن‌ها در جداول ۵ الی ۷ ارایه گردید.



شکل ۲. پاسخ میانگین‌ها و میانگین نسبت SN برای الگوریتم GA

جدول ۵. ترکیبات پیشنهادی روش تاگوچی وسطوح پاسخ به‌دست آمده برای الگوریتم SA

Max_{it}	$MaxSub_{it}$	T_0	Alpha	Time	Value
۱	۱	۱	۱	۵/۶۶	۲۶۳۵/۴۶
۱	۲	۲	۲	۶/۸	۲۶۷۸/۶۱
۱	۳	۳	۳	۹/۶۸	۲۶۴۵/۹۴
۲	۱	۲	۳	۷/۴۷	۲۶۵۵/۵۶
۲	۲	۳	۱	۹/۹۸	۲۵۷۹
۲	۳	۱	۲	۱۴/۳۹	۲۶۵۲/۴۹
۳	۱	۳	۲	۱۰/۰۶	۲۷۲۴/۶۲
۳	۲	۱	۳	۱۳/۰۸	۲۶۷۶/۹۵
۳	۳	۲	۱	۱۸/۹۵	۲۶۶۱/۰۹

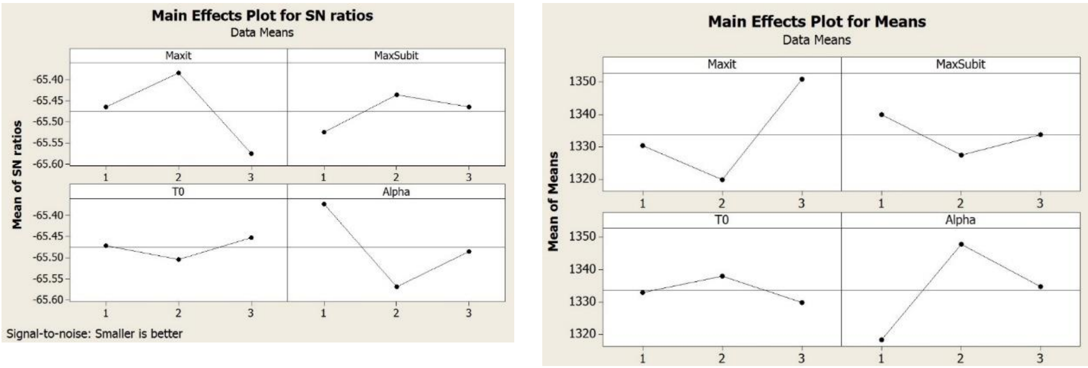
جدول ٦. پاسخ نسبت‌های SN برای الگوریتم SA

Level	Max_{it}	$MaxSub_{it}$	T_0	Alpha
۱	-۶۵/۴۷	-۶۵/۵۳	-۶۵/۴۷	-۶۵/۳۷
۲	-۶۵/۳۸	-۶۵/۴۴	-۶۵/۵	-۶۵/۵۷
۳	-۶۵/۵۸	-۶۵/۴۷	-۶۵/۴۵	-۶۵/۴۹
Delta	۰/۱۹	۰/۰۹	۰/۰۵	۰/۲
Rank	۲	۳	۴	۱

جدول ۷. پاسخ میانگین‌ها برای الگوریتم SA

Level	Max_{it}	$MaxSub_{it}$	T_0	Alpha
۱	۱۳۳۰	۱۳۴۰	۱۳۳۳	۱۳۱۸
۲	۱۳۲۰	۱۳۲۷	۱۳۳۸	۱۳۴۸
۳	۱۳۵۱	۱۳۳۴	۱۳۳۰	۱۳۳۵
Delta	۳۱	۱۲	۸	۲۹
Rank	۲	۳	۴	۱

مشابه آنچه در بخش الگوریتم GA گفته شد، جهت تعیین درجه اهمیت هرکدام از پارامترها، جواب‌ها نسبت به شاخص‌های میانگین پاسخ‌ها و نسبت‌های *SN* تحلیل و پارامترها رتبه‌بندی می‌شود. این شاخص در مقاله [۱۰] توضیح و اهمیت آن ذکر شده است. جداول ۶ و ۷ رتبه‌بندی پارامترها را نشان می‌دهند. با توجه به شکل ۳، ترکیب بهینه پارامترهای الگوریتم SA، به‌دست می‌آید که در آن‌ها، $\boldsymbol{Max_{it}}$ در سطح دوم با مقدار ۴۵۰، $\boldsymbol{MaxSub_{it}}$ در سطح دوم؛ یعنی ۳۰، T_0 در سطح دوم؛ یعنی ۱۰۰ و در نهایت α در سطح دوم با مقدار ۰/۹۶ پاسخ میانگین را حداقل و نسبت‌های SN را حداکثر می‌کنند.



شکل ۳. پاسخ میانگین‌ها و میانگین نسبت SN برای الگوریتم SA

برای انجام طراحی آزمایش‌های چند عاملی برای فاکتورهای الگوریتم FA به‌طور مشابه الگوریتم‌های قبلی عمل می‌کنیم. در الگوریتم FA با توجه به اینکه پنج فاکتور و سه سطح درنظر گرفته شد ترکیبات پیشنهادی روش تاگوچی و نتایج آن‌ها در جداول ۸ الی ۱۰ ارایه گردید.

جدول ۸. ترکیبات پیشنهادی روش تاگوچی وسطوح پاسخ به‌دست آمده برای الگوریتم FA

Max_{it}	N_{pop}	Alpha	Beta	Gama	Time	Value
۱	۱	۱	۱	۱	۱۶/۱۸	۲۸۹۹/۸۱
۱	۱	۱	۱	۲	۱۷/۵	۲۷۶۰/۱۲
۱	۱	۱	۱	۳	۱۳/۶۸	۲۹۳۶/۶۵
۱	۲	۲	۲	۱	۴۸/۵۴	۲۷۱۸/۵۸
۱	۲	۲	۲	۲	۴۸/۶۴	۲۶۶۴/۵۱
۱	۲	۲	۲	۳	۴۸/۷۹	۲۷۴۶/۰۶
۱	۳	۳	۳	۱	۱۰۳/۵۵	۲۷۶۷/۵۸
۱	۳	۳	۳	۲	۱۴۵/۲۱	۲۶۴۸
۱	۳	۳	۳	۳	۱۰۶/۶۹	۲۷۳۹/۳۱
۲	۱	۲	۳	۱	۲۳/۶۴	۲۷۰۸/۵
۲	۱	۲	۳	۲	۲۸/۰۶	۲۸۴۵/۵۷
۲	۱	۲	۳	۳	۲۰/۹۶	۲۸۱۳/۸۳
۲	۲	۳	۱	۱	۶۹/۸۵	۲۷۰۳/۷۳
۲	۲	۳	۱	۲	۷۳/۱	۲۶۶۱/۱۳
۲	۲	۳	۱	۳	۹۰/۹	۲۷۰۷/۴۲
۲	۳	۱	۲	۱	۲۲۱/۷۴	۲۶۴۶
۲	۳	۱	۲	۲	۱۸۲/۸۳	۲۶۱۲
۲	۳	۱	۲	۳	۲۲۴/۱۴	۲۶۱۲
۳	۱	۳	۲	۱	۳۴/۵۶	۲۷۹۹/۰۴
۳	۱	۳	۲	۲	۴۰/۸۴	۲۶۰۹
۳	۱	۳	۲	۳	۳۱/۸۸	۳۰۰۰/۵۸
۳	۲	۱	۳	۱	۱۴۴/۹۲	۲۶۷۶/۶
۳	۲	۱	۳	۲	۱۲۸/۳۴	۲۶۴۹

۳	۲	۱	۳	۳	۱۵۲/۹۹	۲۵۷۹
۳	۳	۲	۱	۱	۲۱۱/۲۷	۲۷۴۳
۳	۳	۲	۱	۲	۲۹۶/۱۹	۲۶۹۰
۳	۳	۲	۱	۳	۲۰۱/۱۴	۲۷۵۰/۱۸

داده‌های تولیدی توسط نرم افزار MINITAB ۱۶ تحلیل شده و جداول ۹و ۱۰ به‌دست آمده است.

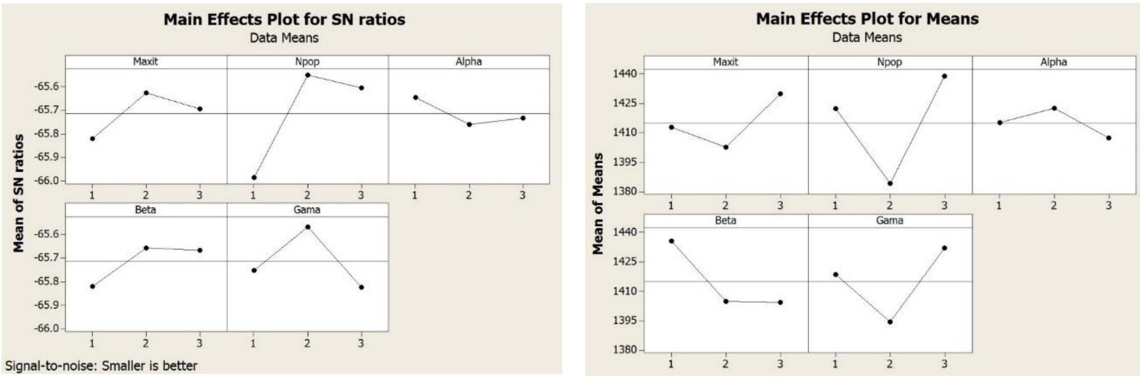
جدول ۹. پاسخ نسبت‌های SN برای الگوریتم FA

Level	Max_{it}	N_{pop}	Alpha	Beta	Gama
۱	-۶۵/۸۲	-۶۵/۹۹	-۶۵/۶۵	-۶۵/۸۲	-۶۵/۷۵
۲	-۶۵/۶۳	-۶۵/۵۵	-۶۵/۷۶	-۶۵/۶۶	-۶۵/۵۷
۳	-۶۵/۶۹	-۶۵/۶۱	-۶۵/۷۳	-۶۵/۶۷	-۶۵/۸۲
Delta	۰/۱۹	۰/۴۳	۰/۱۱	۰/۱۶	۰/۲۶
Rank	۳	۱	۵	۴	۲

جدول ۱۰. پاسخ میانگین‌ها برای الگوریتم FA

Level	Max_{it}	N_{pop}	Alpha	Beta	Gama
۱	۱۴۱۳	۱۴۲۲	۱۴۱۵	۱۴۳۶	۱۴۱۹
۲	۱۴۰۳	۱۳۸۴	۱۴۲۳	۱۴۰۵	۱۳۹۴
۳	۱۴۳۰	۱۴۳۹	۱۴۰۷	۱۴۰۵	۱۴۳۲
Delta	۲۷	۵۵	۱۵	۳۱	۳۸
Rank	۴	۱	۵	۳	۲

جهت تعیین درجه اهمیت هر کدام از پارامترها، جواب‌ها نسبت به شاخص‌های میانگین پاسخ‌ها و نسبت‌های SN تحلیل و پارامترها رتبه‌بندی می‌شود. جداول ۹ و ۱۰ رتبه‌بندی پارامترها را نشان می‌دهند. برای یافتن سطوح بهینه هر یک از فاکتورها از تحلیل اثرات متقابل آن‌ها استفاده می‌شود. این اثرات در شکل ۴ نمایش داده شده است. با توجه به این اشکال ترکیب بهینه پارامترهای کنترل‌کننده الگوریتم FA پیشنهادی به دست می‌آید که در آن‌ها، پارامترهای Max_{it} در سطح دوم؛ یعنی ۱۵۰، N_{pop} در سطح دوم با مقدار ۴۰، α در سطح دوم یعنی ۰.۳، β در سطح دوم؛ یعنی ۲ و در نهایت γ در سطح دوم با مقدار ۱ بهترین ترکیب می‌باشند.



شکل ٤. پاسخ میانگین‌ها و نسبت‌های SN برای الگوریتم FA

٦-١ ارزیابی الگوریتم‌ها

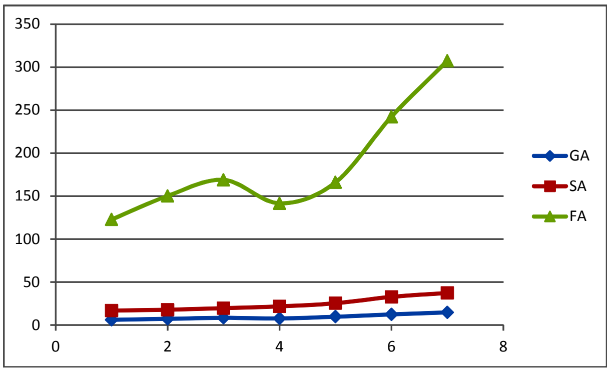
جهت انجام مقایسات بین سه الگوریتم پیشنهادی، ۱۵ مساله آزمایشی در دو مقیاس کوچک و بزرگ تولید و نتایج حاصل از حل مسایل توسط الگوریتم‌ها در قالب جداولی در ادامه آورده شده است.

٦-١-١ مقایسه الگوریتم‌ها برای مسایل با ابعاد کوچک

در جدول ۱۱ نتایج حاصل از اجرای الگوریتم‌ها برای مسایل با ابعاد کوچک گزارش شده است. با توجه به نتایج به‌دست آمده به راحتی می‌توان نتیجه گرفت که الگوریتم ژنتیک علاوه براینکه مقدار تابع هدف کم‌تری دارد، به مراتب زمان محاسباتی کم‌تری را برای رسیدن به جواب نزدیک به بهینه صرف می‌کند. شکل ۵ زمان محاسباتی الگوریتم‌ها را نشان می‌دهد.

جدول ۱۱. نتایج محاسباتی برای مساله با ابعاد کوچک

N	M	GA		SA		FA	
		Time	Cost	Time	Cost	Time	Cost
۱۰	۱۰	۶/۰۴	۷۴۸۵/۷۳	۱۶/۷۷	۸۳۴۱/۵۹	۱۲۲/۷۱	۹۲۲۵/۴۱
۱۰	۲۰	۷/۱۹	۱۶۰۴۱/۰۴	۱۷/۷	۱۷۱۷۹/۱	۱۵۰/۱۱	۱۷۴۲۵/۱۵
۱۰	۳۰	۸/۳۳	۲۸۶۵۰/۹	۱۹/۵۷	۳۸۸۰۶/۳۹	۱۶۸/۸۱	۲۸۰۸۸/۴۶
۱۵	۱۵	۷/۷۱	۱۲۴۲۰/۶۸	۲۱/۶۹	۱۳۶۴۴/۳۲	۱۴۱/۵۴	۱۴۷۹۹/۶۶
۱۵	۳۰	۹/۶۵	۲۶۴۴۵/۱۳	۲۵/۳۲	۲۸۰۹۷/۱۲	۱۶۵/۸۹	۲۸۰۲۹/۲
۲۰	۴۰	۱۲/۴۲	۴۰۱۵۷/۴۹	۳۲/۶۹	۴۲۴۴۶/۸۸	۲۴۲/۲۲	۴۳۷۱۵/۶۳
۲۰	۶۰	۱۴/۷۵	۶۸۵۹۷/۹۶	۳۷/۲۶	۱۷۸۶۱۱/۶۲	۳۰۷/۲۱	۲۱۳۵۸۶/۴۲



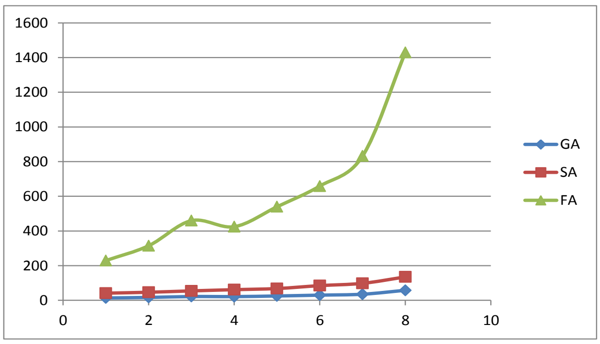
شکل ۵. نمودار زمان محاسباتی الگوریتم‌های پیشنهادی برای مساله با ابعاد کوچک

٦-١-٢ مقایسه الگوریتم‌ها برای مسایل با ابعاد بزرگ

در این نوع مسایل هم مشابه‌ی مسایل با ابعاد متوسط الگوریتم ژنتیک در همه‌ی مسایل مقدار تابع هدف بهتر و زمان محاسباتی کم‌تری را نسبت به دو الگوریتم دیگر به دست آورده است. شایان ذکر است همان طور که در جدول ۱۲ مشاهده می‌شود با افزایش تعداد تامین کننده‌ها و مشتریان، زمان محاسباتی روند صعودی را طی می کند. شکل ۶ زمان محاسباتی الگوریتم‌ها را نشان می‌دهد.

جدول ۱۲. نتایج محاسباتی برای مساله با ابعاد بزرگ

N	M	GA		SA		FA	
		Time	Cost	Time	Cost	Time	Cost
۳۰	۳۰	۱۳/۶۲	۳۳۷۰۲/۴۸	۴۱/۳۱	۳۵۶۷۹/۸۴	۲۲۹/۳۲	۳۷۵۰۳/۸۸
۳۰	۵۰	۱۶/۶۴	۶۰۶۱۱/۵۶	۴۶/۵۷	۶۱۵۳۶/۹۷	۳۱۴/۸۹	۶۳۵۸۱/۷۴
۳۰	۸۰	۲۱/۸۲	۱۰۳۶۳۹/۶۷	۵۴/۵۱	۱۴۸۹۵۲/۱	۴۵۹/۶۶	۱۳۶۶۵۰/۲۳
۴۰	۶۰	۲۱/۵۲	۷۹۴۱۸/۱۴	۶۱/۴۴	۸۳۴۷۶/۰۳	۴۲۴/۵۶	۸۶۴۵۷/۶۸
۴۰	۸۰	۲۵/۱۱	۱۰۹۴۰۲/۸۳	۶۷/۹۹	۱۱۳۳۰۴	۵۳۹/۴۹	۱۱۶۳۶۵/۴۴
۵۰	۸۰	۲۹/۹۴	۱۲۲۵۳۲/۷۲	۸۴/۸	۱۲۶۴۰۸/۹۸	۶۵۸/۹۳	۱۲۸۳۶۳/۵۴
۵۰	۱۰۰	۳۵/۲۲	۱۵۳۴۳۵/۷۶	۹۷/۹۹	۱۵۸۴۷۹/۸۳	۸۳۲/۰۵	۱۶۲۳۵۱/۷
۵۰	۲۰	۵۷/۸۳	۴۷۰۲۵۵۴/۰۴	۱۳۴۰۹	۴۸۲۹۶۳۵/۸۸	۱۴۲۹/۶۹	۴۷۲۹۲۳۳/۹۹



شکل ٦. نمودار زمان محاسباتی الگوریتم‌های پیشنهادی برای مساله با ابعاد بزرگ

۷ نتیجه‌گیری و پیشنهادهای آتی

مساله مورد بحث در این مطالعه، یک موضوع چالش برانگیز برای بسیاری از محققان از ابتدای معرفی آن بوده است. برای حل مدل ارایه شده با توجه به NP-*hard* بودن مدل، سه الگوریتم فراابتکاری ژنتیک، کرم شب‌تاب و تبرید شبیه‌سازی توسعه داده شد. پارامترهای موجود در الگوریتم‌های پیشنهای از طریق تاگوچی تنظیم شده است و سپس، با استفاده از این الگوریتم‌ها به حل مسایل حمل و نقل هزینه‌ی ثابت در ابعاد متفاوت پرداخته شد. کارایی الگوریتم‌های پیشنهادی نشان داده شد و در نهایت دو مساله در ابعاد کوچک و بزرگ با داده‌های فازی را با الگوریتم‌های پیشنهادی با پارامترهای تنظیم شده حل شد و نتیجه‌ای که از آن حاصل شد این بود که الگوریتم ژنتیک علاوه براینکه مقدار تابع هدف کم‌تری دارد، به مراتب زمان محاسباتی کم‌تری را برای رسیدن به جواب نزدیک به بهینه صرف می‌کند. تحقیقات نامحدودی می‌توان بر روی مسایل حمل و نقل با هزینه ثابت همچون در طراحی شبکه حمل و نقل انجام داد. از الگوریتم‌های فراابتکاری دیگری چون الگوریتم مورچگان، جستجوی ممنوع، الگوریتم کشتل و غیره برای حل مساله مورد نظر می‌توان استفاده کرد. نیز محدودیت‌های جدیدی به مساله اضافه شود. به عنوان مثال می‌توان مساله را با اضافه کردن محدودیت پنجره زمانی برای ارسال کالا به مشتریان حل کرد. همچنین می‌توان نوع حمل و نقل را نیز به مدل مساله اضافه کرد.

منابع

[1] Balinski, M.L. (1961). Fixed cost transportation problems. Naval Research Logistic Quarterly, 8(1), 41–54.

[2] Kim, D., Pardalos, P. M. (1999). A solution approach to the fixed charge network flow problem using a dynamic slope scaling procedure. Operations Research Letters, 24(4), 195-203.

[3] Gen, M., Cheng, R. (2000). Genetic algorithms and engineering optimization. NY: John Wiley and Sons.

[4] Gen, M., Cheng, R. (1997). Genetic algorithms and engineering design. NY: John Wiley and Sons.

[5] Adlakha, V., Kowalski, K. (2003). A simple heuristic for solving small fixed-charge transportation problems. Omega, 31(3), 205-211.

[6] Gen, M., Altiparmak, F., Lin, L. (2006). A genetic algorithm for two-stage transportation problem using priority-based encoding. OR spectrum, 28(3), 337-354.

[7] Yang, L., Liu, L. (2007). Fuzzy fixed charge solid transportation problem and algorithm. Applied Soft Computing, 7(3), 879-889.

[8] Jo, J. B., Li, Y., Gen, M. (2007). Nonlinear fixed charge transportation problem by spanning tree-based genetic algorithm. Computers and Industrial Engineering, 53(2), 290-298.

[9] Kowalski, K., Lev, B. (2008). On step fixed-charge transportation problem. Omega, 36(5), 913-917.

[10] Jawahar, N., Balaji, A. N. (2009). A genetic algorithm for the two-stage supply chain distribution problem associated with a fixed charge. European Journal of Operational Research, 194(2), 496-537.

[11] Balaji, A. N., Jawahar, N. (2010). A simulated annealing algorithm for a two-stage fixed charge distribution problem of a supply chain. International Journal of Operational Research, 7(2), 192-215.

[12] Hajiaghaei-Keshteli, M., Molla-Alizadeh-Zavardehi, S., Tavakkoli-Moghaddam, R. (2010). Addressing a nonlinear fixed-charge transportation problem using a spanning tree-based genetic algorithm. Computers and Industrial Engineering, 59(2), 259-271.

[13] Molla-Alizadeh-Zavardehi, S., Hajiaghaei-Keshteli, M., Tavakkoli-Moghaddam, R. (2011). Solving a capacitated fixed-charge transportation problem by artificial immune and genetic algorithms with a Prüfer number representation. Expert Systems with Applications, 38(8), 10462-10474.

[14] Lotfi, M. M., Tavakkoli-Moghaddam, R. (2013). A genetic algorithm using priority-based encoding with new operators for fixed charge transportation problems. Applied Soft Computing, 13(5), 2711-2726.

[15] Molla-Alizadeh-Zavardehi, S., Nezhad, S. S., Tavakkoli-Moghaddam, R., Yazdani, M. (2013). Solving a fuzzy fixed charge solid transportation problem by metaheuristics. Mathematical and Computer Modelling, 57(5), 1543-1558.

[16] El-Sherbiny, M. M., Alhamali, R. M. (2013). A hybrid particle swarm algorithm with artificial immune learning for solving the fixed charge transportation problem. Computers and Industrial Engineering, 64(2), 610-620.

[17] Molla-Alizadeh-Zavardehi, S., Sanei, M., Soltani, R., Mahmoodirad, A. (2013). Solving a Step Fixed Charge Transportation Problem by a Spanning Tree-Based Memetic Algorithm. International Journal of Mathematical Modelling and Computations, 4(2).

[18] Kowalski, K., Lev, B., Shen, W., Tu, Y. (2014). A fast and simple branching algorithm for solving small scale fixed-charge transportation problem. Operations Research Perspectives, 1(1), 1- 5.

[19] Shirazi, N., Seyyed Esfahani, M., Soleimani, H. (2015). Modeling and solving a three-stage fixed charge transportation problem considering stochastic demand and price. Journal of Industrial Engineering and Management Studies, 2(1), 27-40.

[20] Khurana, A., Adlakha, V. (2015). On multi-index fixed charge bi-criterion transportation problem. OPSEARCH, 52(4), 733-745.

[21] Liou, T. S., Wang, M. J. J. (1992). Ranking fuzzy numbers with integral value. Fuzzy sets and systems, 50(3), 247-255.

[22] Holland, J. H. (1975). Adaptation in natural and artificial systems. An introductory analysis with application to biology, control, and artificial intelligence. Ann Arbor, MI: University of Michigan Press.

[23] Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. Science, 220(4598), 671-680.

[24] Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., Teller, E. (1953). Equation of state calculations by fast computing machines. The journal of chemical physics, 21(6), 1087-1092.

[25] Yang, X. S. (2009). Firefly algorithms for multimodal optimization. In International Symposium on Stochastic Algorithms, pp. 169-178. Springer Berlin Heidelberg.

[26] Taguchi, G. (1986). Introduction to quality engineering: designing quality into products and processes, Published by Quality Resources, White Plains, NY.