

الگوریتم شبیه‌سازی تبرید مبتنی بر ابر برای طراحی شبکه زنجیره تامین حلقه بسته: نمایش جواب درخت پوشا

احسان یادگاری*، اکبر عالم تبریز**، مصطفی زندیه***

تاریخ دریافت: ۹۷/۷/۲۱ - تاریخ ارسال: ۹۷/۱۰/۵

چکیده

در طی دهه گذشته، به دلیل قوانین زیست محیطی و فضای رقابتی تدوین یک برنامه تاکتیکی موثر که از یک طرف قادر به برنامه‌ریزی کارا و یکپارچه تامین کالا برای مشتریان باشد و از طرف دیگر مسئولیت سازمان‌ها برای جمع‌آوری کالای معیوب را در نظر بگیرد، امری اجتناب‌ناپذیر به نظر می‌رسد. در این مقاله یک برنامه‌ریزی خطی عدد صحیح آمیخته در نظر گرفته شده است که در جهت رو به جلو مواد اولیه را از تامین‌کنندگان به کارخانه‌ها و در ادامه از طریق مراکز توزیع، محصول نهایی را به مشتریان تحویل می‌دهد. از طرفی به طور همزمان کالای بازمانده از مشتریان را جمع‌آوری کرده وارد چرخه بازسازی و یا انهدام ایمن می‌کند. مدل‌سازی ریاضی این مساله با هدف کمینه کردن هزینه‌های تاسیس تسهیلات در نقاط بالقوه و همچنین جریان بهینه مواد در لایه‌های شبکه انجام شده است. از آنجایی که مساله مورد بررسی از دسته مسائل NP-hard است، برای حل آن از الگوریتم فراابتکاری شبیه‌سازی تبرید مبتنی بر ابر برای اولین بار در پیشینه این حوزه استفاده شده است. همچنین برای نمایش جواب از روش درخت پوشا که نسبت به روش‌های دیگر در ادبیات موضوع از آرایه‌های کمتری استفاده می‌کند بهره‌جسته‌ایم. برای تحلیل دقت و سرعت الگوریتم مورد بررسی، عملکرد آن را با الگوریتم ژنتیک و الگوریتم شبیه‌سازی تبرید (که در ادبیات موضوع به کار گرفته شده بودند) مقایسه کرده‌ایم. نتایج نشان می‌دهند تابع هزینه در الگوریتم شبیه‌سازی تبرید مبتنی بر ابر نسبت به هر دو الگوریتم مورد بررسی در ادبیات پاسخ‌های دقیق‌تری را ارائه می‌دهد. همچنین از نظر معیار سرعت همگرایی، روش پیشنهادی نسبت به الگوریتم ژنتیک در وضعیت بهتری است اما نسبت به الگوریتم شبیه‌سازی تبرید تفاوت معنا داری ندارد.

واژگان کلیدی: شبیه‌سازی تبرید مبتنی بر ابر، درخت پوشا، طراحی شبکه زنجیره‌تامین، حلقه بسته، برنامه‌ریزی خطی عدد صحیح آمیخته

* دکتری مدیریت تولید و عملیات، گروه مدیریت صنعتی، دانشکده مدیریت و حسابداری، دانشگاه شهید بهشتی

** استاد دانشگاه، گروه مدیریت صنعتی، دانشکده مدیریت و حسابداری، دانشگاه شهید بهشتی. (نویسنده مسئول)

a-tabriz@sbu.ac.ir

*** دانشیار دانشگاه، گروه مدیریت صنعتی، دانشکده مدیریت و حسابداری، دانشگاه شهید بهشتی

مقدمه و پیشینه

توسعه روز افزون فضای رقابتی و جهانی شدن بازار محصولات موجب شده است که سازمان‌ها در جهت بقای خود، تلاش چشمگیری را در راستای تأمین^۱ و تدارک^۲، تولید^۳ و توزیع کالا^۴ شرکت خود بکار گیرند تا توان پاسخ‌گویی به نیازهای متنوع مشتریان را در حداقل زمان و با صرف حداقل هزینه داشته باشند. در سالیان اخیر و با تشدید فضای رقابتی این موضوع که به عنوان یک عنصر مؤثر در حیات اقتصادی و صنعتی شناخته شده است، بیش از پیش به عنوان یک مسأله بسیار مهم مورد توجه قرار گرفته است.

مسئله طراحی شبکه زنجیره تأمین^۵ شامل تصمیمات استراتژیکی می‌شود که به پیکربندی زنجیره تأمین اشاره دارد و به عنوان مسئله زیر ساختاری در مدیریت زنجیره تأمین، اثرات دیرپایی روی سایر تصمیمات تاکتیکی و عملیاتی شرکت دارد. در حالت کلی پروژه طراحی شبکه با مشخص کردن جایگاه‌های مکان‌یابی و ظرفیت‌های مورد نیاز برای تسهیلات جدید و برنامه‌ریزی خرید، تولید، توزیع و نگهداری محصولات مواجه است.

افلیچمن^۶ (۲۰۰۱) به عنوان اولین کار در این حوزه به ارائه یک مدل برنامه‌ریزی خطی عدد صحیح آمیخته برای طراحی یکپارچه زنجیره تأمین مستقیم و معکوس پرداخت. در شبکه مورد بررسی ابتدا محصولات مصرف شده از مشتریان جمع‌آوری و پس از احیا به طور مجدد توزیع می‌شوند. در واقع می‌توان گفت هر چند این شبکه یک شبکه حلقه بسته است و دارای هر دو جریان مستقیم و معکوس است، لیکن تولید و توزیع محصولات نو مورد توجه قرار نگرفته است لذا با مسامحه می‌توان این مدل را طراحی یکپارچه با تعریف ارائه شده خواند.

پیشوایی و همکاران (۲۰۱۲) یکپارچه‌سازی در طراحی شبکه زنجیره تأمین را به دو دسته (۱) یکپارچه‌سازی عمودی و (۲) یکپارچه‌سازی افقی طبقه‌بندی کرده‌اند. یکپارچه‌سازی

-
1. Supply
 2. Procurment
 3. Production
 4. Distribution
 5. Supply Chain Network Design
 6. Fleischmann

عمودی به معنای یکپارچه‌سازی تصمیمات در سطوح استراتژیک^۱ (بلند مدت)، تاکتیکی^۲ (میان مدت) و عملیاتی^۳ (کوتاه مدت) در طراحی شبکه است. طراحی شبکه زنجیره تامین در ماهیت یک تصمیم استراتژیک است که به طور معمول شامل تعیین محل تسهیلات، ظرفیت آن‌ها، تعداد رده‌ها در زنجیره و نحوه ارتباط تسهیلات می‌باشد. لذا باید توجه داشت که یکپارچه کردن تصمیمات سطوح پایین‌تر در طراحی شبکه باید با حفظ تصمیمات سطح استراتژیک باشد. در صورت یکپارچه‌سازی این تصمیمات با تصمیمات سطح تاکتیکی نظیر مدیریت موجودی و یا سطح عملیاتی مانند مسیریابی، می‌توان از ایجاد زیربهینگی جلوگیری کرد. خوانندگان را جهت مطالعه بیشتر درباره این نوع یکپارچه‌سازی به مقاله مروری گویندان و همکاران (۲۰۱۷) ارجاع می‌دهیم. به دلیل اینکه این نوع یکپارچه‌سازی سطوح تصمیم‌گیری را به طور عمودی ادغام می‌کند این نوع یکپارچه‌سازی، یکپارچه‌سازی عمودی نام گرفته است.

کو^۴ و اوان^۵ (۲۰۰۷) یک مدل برنامه‌ریزی غیرخطی عدد صحیح آمیخته برای طراحی شبکه زنجیره تامین یکپارچه برای فراهم آوردن خدمات زنجیره تامین طرف سوم ارائه شده است. برای از عهده برآمدن عدم قطعیت موجود در این شرایط پیشنهاد گردیده که مدل در دوره‌های مختلف اجرا گردد تا از پویایی لازم برخوردار باشد، لذا پارامترهای مسئله برای هر دوره معین می‌گردند، و در دوره بعد مدل به طور مجدد برای پارامترهای جدید حل می‌شود. در این مقاله نیز از تسهیلات ترکیبی استفاده شده است. کار مشابهی توسط مین^۶ و کو (۲۰۰۸) نیز ارائه شده است.

یکی از دیگر مقالاتی که به خوبی به طراحی یکپارچه شبکه زنجیره تامین مستقیم و معکوس پرداخته است مقاله لی و دانگ^۷ (۲۰۰۸) است. در این مقاله از یک نوع تسهیلات ترکیبی که

-
1. Strategic
 2. Tactical
 3. Operational
 4. Ko
 5. Evans
 6. Min
 7. Lee and Dong

هم نقش مراکز توزیع (انبار) در جریان مستقیم و هم نقش مراکز جمع‌آوری در جریان معکوس را به عهده دارد، برای طراحی شبکه زنجیره تامین محصولات رایانه‌ای استفاده شده است. این مسئله با استفاده از برنامه ریزی خطی عدد صحیح آمیخته^۱ مدل شده و به دلیل پیچیدگی زیاد با استفاده از یک روش ابتکاری آمیخته^۲ با روش فراابتکاری جستجوی ممنوع^۳ حل شده است. اما این مقاله دارای نقاط ضعفی است که از آن جمله می‌توان به فرض‌های ساده سازی مانند مشخص بودن تعداد تسهیلات ترکیبی، استفاده از تنها یک کارخانه اشاره کرد. همچنین این مقاله در جریان زنجیره تامین معکوس تنها به فرآیند احیا توجه کرده است. وانگ^۴ و سو^۵ (۲۰۱۰) یک مدل برای طراحی شبکه زنجیره تامین حلقه‌بسته^۶ که شامل تأمین‌کنندگان، کارخانه‌های تولیدی، مراکز توزیع و مراکز اسقاط می‌باشد، ارائه دادند. در این مقاله نویسندگان با توجه به غیر صحیح بودن مقادیر برگشت محصولات که ضریبی از تقاضای مشتریان فرض شده است، به تبدیل مدل پیشنهادی به برنامه ریزی خطی عدد صحیح پرداخته‌اند.

در این حوزه طبق مقالات مروری گویندان (۲۰۱۷) مقالات سال‌های ۲۰۱۰ تا ۲۰۱۷ در زمینه طراحی شبکه از نظر نوع مدل‌سازی مورد بررسی قرار گرفته اند که در آن به دسته بندی در این حوزه از لحاظ مدل‌سازی و روش‌های حل این نوع مسائل پرداخته شده است. روش‌های متعددی در حوزه روش‌های نمایش جواب مسائل طراحی شبکه توسعه داده شده است که در اینجا به دو مورد از برترین‌ها اشاره می‌کنیم. روش اول شماره‌های پروفرف^۷ است که از مفهوم درخت پوشا به عنوان نمایش جواب‌های الگوریتم‌های فراابتکاری که اولین بار توسط سیاریف^۸ (۲۰۰۲) معرفی شد، بهره می‌گیرد. در این روش با آمیختن مفهوم پروفرف^۹

1. Mixed Integer Linear Programming
2. Hybrid Heuristics
3. Tabu Search
4. Wang
5. Hsu
6. Closed-loop Supply Chain Network Design
7. Prufer Numbers
8. Syarif
9. Prufer Codes

و درخت پوشا، این روش را بر روی یک الگوریتم ژنتیک^۱ توسعه دادند. در این روش یک درخت n رأسی شامل $n-2$ گره است، و می‌توان آن را با یک الگوریتم تکرار شونده ساده تولید کرد.

روش بعدی روش کدگذاری اولویت محور^۲ است که در آن از اعداد غیر تکراری برای انتخاب ترتیب مقدار دهی به گره‌های متوالی زنجیره تامین استفاده می‌شود. مزیت این روش نسبت به روش مورد استفاده در این مقاله، ساده‌تر و قابل فهم‌تر بودن کدنویسی و رمزگشایی از جواب‌هاست. یکی دیگر از نقاط قوت این روش عدم نیاز به فرایند تطبیق مجدد^۳ جواب‌ها بعد از به کارگیری دیگر اپراتورهای الگوریتم می‌باشد. در مقابل روش مورد استفاده درخت پوشا در این مقاله کدگذاری را با کمترین میزان درایه در یک جواب ماتریسی انجام می‌دهد که این امر نقش مهمی در کمتر کردن سرعت حل الگوریتم‌ها دارد. اما نقطه ضعف روش درخت پوشا^۴ مورد استفاده در عملیات تطبیق مجدد بعد از به کارگیری برخی از اپراتورها و در نتیجه دشواری بیشتر برای کدنویس است!

در جدول (۱) پیشینه تحقیق از لحاظ رویکردهای حل و همچنین روش‌های نمایش جواب فراابتکاری طبقه‌بندی شده است.

همانطور که مشاهده می‌شود، با وجود ادبیات غنی در حوزه طراحی شبکه‌های زنجیره تامین، در اغلب مدل‌های ارائه شده روش‌های حل دقیق برای حل مدل‌ها در ابعاد کوچک و متوسط به کار گرفته شده است. همچنین روش درخت پوشا با توجه به سرعت بالای این روش که از کمترین آرایه‌های ممکن در ماتریس جواب بهره می‌گیرد تا به حال تنها در دو مقاله به کار برده شده است. از مهمترین دلایل عدم استفاده از این روش می‌توان به دشواری‌های کدنویسی این روش و نیز عملیات تطبیق مجدد جواب‌ها پس از به کارگیری برخی اپراتورها نام برد. در این پژوهش تلاش شده پس از تشریح یک مدل زنجیره تامین حلقه بسته غیرخطی،

-
1. Genetic Algorithm
 2. Priority based Encoding
 3. Repair Mechanism
 4. Spanning Tree

به فرایند خطی کردن مدل پردازیم که در نتیجه آن مکان‌های بهینه برای تاسیس کارخانه‌ها، مراکز توزیع و مراکز جمع‌آوری کالای بازگشتی مشخص شود. و در ادامه برای اولین بار از روش الگوریتم شبیه‌سازی تبرید مبتنی بر ابر در مسائل طراحی شبکه استفاده کنیم. همچنین برای نمایش جواب از روش درخت پوشا استفاده شده و نتایج آن با مقالاتی در همین زمینه مقایسه شده است.

جدول ۱: خلاصه ادبیات طراحی شبکه زنجیره تامین با تاکید بر رویکردهای حل مدل

روش حل	نمایش جواب	
Forward networks		
آزادسازی لاگرانژی ^۱	---	Jayaraman and Pirkul (۲۰۰۱)
الگوریتم انجماد تدریجی ^۲	غیره	Jayaraman, Gupta, and Pirkul (2003)
حل دقیق	---	Li, Chen, and Wang (۲۰۱۱)
حل دقیق	---	Tsiakis and Papageorgiou (2008)
روش تعاملی فازی ^۳	---	Pishvaei and Razmi (۲۰۱۲)
الگوریتم ژنتیک	غیره	I. Syarif, Prugel-Bennett, and Wills (2012)
آزادسازی لاگرانژی	---	Elhedhli and Merrick (۲۰۱۲)
Reverse networks		
حل دقیق	---	Krikke, van Harten, and Schuur (1999)
الگوریتم ژنتیک	غیره	Min and Ko (2008)
الگوریتم جستجوی ممنوعه	غیره	Aras and Crowther (2008)
حل دقیق	---	Gírio et al. (2010)
حل دقیق	---	Govindan, Khodaverdi, and Jafarian (2013)

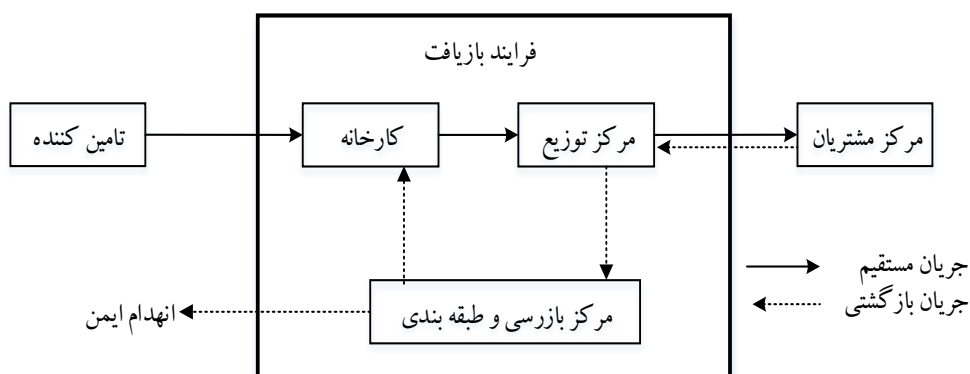
1 Lagrangian Relaxation

2 Simulated Annealing

3 Interactive Fuzzy Solution Approach

closed-loop networks		
Fleischmann et al. (2001)	---	حل دقیق
Lu and Bostel (2007)	---	آزادسازی لاگرانژی
Salema, Póvoa, and Novais (2009)	---	الگوریتم شاخه و حد ^۱
Pishvaei and Razmi (2012)	---	روش تعاملی فازی
Wang and Hsu (2010)	درخت پوشا	الگوریتم ژنتیک
Devika, Jafarian, and Nourbakhsh (2014)	اولویت محور	فراابتکاری ترکیبی
Yadegari, Ekhtiari, Zandieh, and Alem-Tabriz (2014)	غیره	الگوریتم ایمنی ^۲
Yadegari, Najmi, Ghomi-Avili, and Zandieh (2015)	چندمسیره گسسته	الگوریتم ممتیک ^۳
Yadegari, Zandieh, Yadegari, Zandieh & Najmi, (2015)	درخت پوشا	فراابتکاری ترکیبی
Ghayebloo, Tarokh, Venkatadri, and Diallo (2015)	---	حل دقیق
Kaya and Urek (2016)	اولویت محور	ابتکاری ترکیبی
Yi, Huang, Guo, and Shi (2016)	غیره	الگوریتم ژنتیک
This Paper	درخت پوشا	شبیه‌سازی تبرید مبتنی بر ابر ^۴

1. Branch and Bound
2. Immune Algorithm
3. Memetic Algorithm
4. Cloud Theory Based Simulated Annealing



شکل ۱: ساختار کلی شبکه رو به تامین و جمع‌آوری کالا مورد بررسی

برخی از ویژگی‌ها و شرایط مساله مورد نظر به شرح زیر است:

۱. مکان تأمین‌کنندگان، مشتریان و مرکز اسقاط معلوم و ثابت است.
۲. مکان‌های بالقوه برای تاسیس مراکز کارخانه‌ها، مراکز توزیع/جمع‌آوری، مراکز بازرسی معلوم و به صورت گسسته می‌باشد.
۳. جریان مواد تنها بین دو سطح متوالی از لایه‌های شبکه می‌تواند برقرار باشد و همچنین میزان مرادده بین تسهیلات در یک لایه وجود ندارد.
۴. ظرفیت تسهیلات در شبکه محدود می‌باشد.
۵. تقاضا به صورت قطعی در نظر گرفته شده است.
۶. میزان کالای برگشتی به اندازه درصد ثابتی از تقاضا در نظر گرفته می‌شود.
۷. زمان تدارک‌ها به صورت معلوم و ثابت در نظر گرفته شده است.
۸. کیفیت محصولات تعمیر شده برابر کیفیت محصولات تولیدی از کارخانه‌ها می‌باشند.
۹. مراکز جمع‌آوری تا پایان هر دوره تمام کالاهای برگشتی از تمام مشتریان را جمع‌آوری می‌کنند.

جهت درک بهتر مدل ریاضی پیشنهادی ابتدا به صورت کلامی مدل را توضیح می‌دهیم:

تابع هدف به صورت کلامی:

هزینه کمینه = هزینه‌های ثابت بازگشایی + هزینه‌های حمل و نقل

محدودیت‌های مدل به صورت کلامی:

- ارضای تمام تقاضاها در جهت مستقیم و معکوس
- توازن جریان بین گره‌ها
- محدودیت‌های ظرفیت
- محدودیت‌های منطقی مربوط به سطوح متفاوت ظرفیت
- محدودیت‌های غیر منفی بودن و صفر و یک

در ادامه به تشریح مجموعه‌های مدل پرداخته می‌شود. لازم به ذکر است که مدل مربوطه برگرفته از مقاله وانگ و سو (۲۰۱۰) است، لذا جهت مطالعه دقیق‌تر لطفاً به آن مقاله مراجعه بفرمایید.

مجموعه‌ها

$i = 1, 2, \dots, I$	مجموعه مکان‌های ثابت تأمین‌کنندگان	i
$j = 1, 2, \dots, J$	مجموعه مکان‌های بالقوه برای مراکز کارخانه‌ها	j
$k = 1, 2, \dots, K$	مجموعه مکان‌های بالقوه برای مراکز توزیع/جمع‌آوری	k
$l = 1, 2, \dots, L$	مجموعه مکان‌های ثابت مشتریان	l
$m = 1, 2, \dots, M$	مجموعه مکان‌های بالقوه برای مراکز بازرسی	m

تابع هدف مساله به کمینه کردن هزینه‌ها می‌پردازد که این هزینه‌ها شامل هزینه‌های حمل و نقل در شبکه و هزینه‌های ثابت ایجاد واحدها در مکان‌های بالقوه می‌باشند. محدودیت‌های مدل عموماً دو قسم می‌باشند: محدودیت‌های ظرفیت تسهیلات و محدودیت‌های تعامل جریان

در گره‌ها عبارات (۴) و (۵) مربوط به محدودیت ظرفیت تامین کنندگان و کارخانه‌ها در جریان رفت می‌باشند.

در مدل لجستیک حلقه بسته ارائه شده به تعداد $(I + 2J + 4k + 4M)$ محدودیت داریم و تعداد متغیرها $(I \times J + J \times k + k \times L \times L \times k + k \times M + M + J + J \times k + 2M)$ می‌باشد که $(I + k + M)$ تای آن‌ها متغیرهای صفر و یک می‌باشند. از این تعداد $2M$ متغیر و M محدودیت اضافه سهم فرآیند خطی کردن مدل می‌باشند.

از آنجایی که مساله طراحی شبکه یکپارچه لجستیک شامل مساله جابه‌جایی تسهیلات با ظرفیت‌های مختلف می‌باشد که به عنوان مسائل NP-Completed شناخته شده‌اند، مساله ارائه شده جزء مسائل NP-Hard می‌باشد. برای حل این گونه مسائل نمی‌توان در ابعاد واقعی از روش‌های دقیق استفاده کرد. لذا در این تحقیق الگوریتم‌های شبیه‌سازی ذوب و در نهایت از یک الگوریتم ترکیبی استفاده شده است.

کدگذاری جواب‌ها^۱ با روش درخت پوشای اصلاح شده

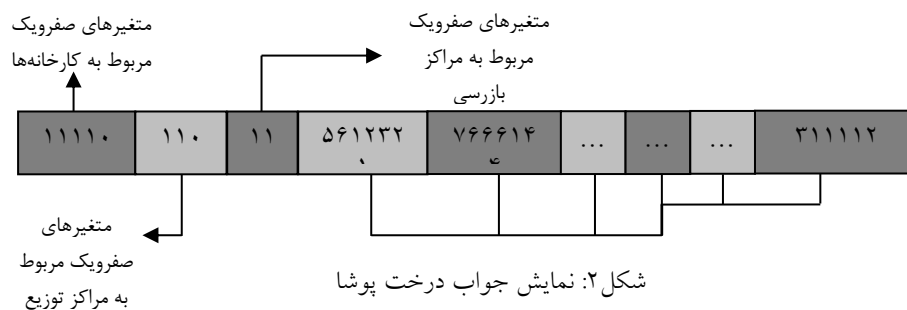
از آنجایی که مساله طراحی شبکه زنجیره تامین حلقه بسته جز مسائل مکان‌یابی / تخصیص ظرفیت‌دار^۲ است و می‌تواند به عنوان یک مساله کوله پشتی چند انتخابی^۳ در نظر گرفته شود، به عنوان یک مساله NP-hard شناخته شده است.

روش کدگذاری درخت پوشا به صورت کلی در مسائلی به کار می‌رود که در آن‌ها حلقه وجود نداشته‌باشد. اگرچه مساله طراحی شبکه یکپارچه دارای حلقه می‌باشد ولی می‌توان آن را به بخش‌های کوچک‌تر که حلقه‌ای نیستند تقسیم کرد.

در مساله طراحی شبکه لجستیک یکپارچه مورد بررسی دو سطح اول یعنی سطح تامین کنندگان و سطح کارخانه‌ها که با روش کدگذاری درخت گسترده نمایش داده شده است.

-
1. Solution Encoding
 2. Capacitated Location-Allocation Problem
 3. Multiple-Choice Knapsack Problem

یک راه حل برای مسأله مورد نظر، با یک لیست از پارامترها نشان داده می‌شود که به آن‌ها نمایش جواب یا ژنوم^۱ می‌گویند. ژنوم‌ها عموماً به صورت یک رشته ساده از داده‌ها نمایش داده می‌شوند. در الگوریتم تبرید به جهت نمایش ژنوم‌ها از همان روش نمایش جواب کد گذاری به روش درخت گسترده که در بالا شرح داده شده پرداخته می‌شود. بنابراین هر ژنوم (جواب) شامل ۱۲ قسمت و حاوی $2 \times (I+J+J+K+K+L+L+M+L+I)$ آرایه می‌باشد. هر دو قسمت متوالی نشان دهنده جریان بین دو سطح متوالی در شبکه زنجیره تامین می‌باشد.



رمزگشایی از جواب‌ها

با یک مثال به توضیح نحوه رمزگشایی از یک جواب و تبدیل و تفسیر آن به جواب‌های ملموس می‌پردازیم. فرض کنید ماتریس C یک نمونه جواب است که ارتباط ۹ گره را با یکدیگر نمایش می‌دهد.

۳	۴	۲	۵	۸	۵	۳	۹
---	---	---	---	---	---	---	---

شکل ۳: ماتریس جواب به روش درخت پوشا

حد مقادیر در ماتریس جواب به روش درخت پوشا بین اعداد ۱ تا مجموع گره‌های مبدا و مقصد است که در این مثال این حد بین ۱ تا ۹ می‌باشد. از آن جایی که ماتریس C شامل ۸

قسمت می‌باشد، ما می‌توانیم بفهمیم که در مساله ۹ گره وجود دارد، زیرا در این روش تعداد مؤلفه‌های ماتریس همواره یک واحد کمتر از تعداد گره‌های مساله می‌باشد. برای درک رابطه بین گره‌ها ابتدا ما باید یک ماتریس 1×8 شامل اعداد ۲ تا ۹ بسازیم، بنابراین ماتریس C' به صورت زیر نمایش داده می‌شود:

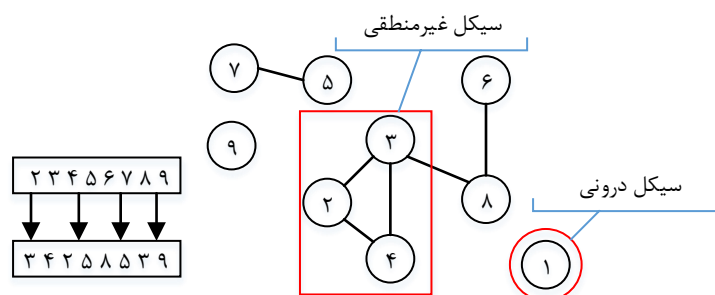
۲	۳	۴	۵	۶	۷	۸	۹
---	---	---	---	---	---	---	---

شکل ۴: موقعیت هر آل و در ماتریس جواب

اگر این ۲ ماتریس (اشکال شماره ۳ و ۴ که در مجموع در شکل شماره ۵ نمایش داده شده- اند) را کنار یکدیگر قرار دهیم، رابطه بین گره‌ها بدست می‌آید. بدین گونه که مؤلفه اول را از ماتریس C' و مؤلفه دوم را از ماتریس C برمی‌داریم. بنابراین روابط به صورت زیر قابل بیان می‌باشند:

$$(۹,۹), (۸,۳), (۷,۵), (۶,۸), (۵,۵), (۴,۲), (۳,۴), (۲,۳)$$

درخت حاصل شده یک درخت معیوب می‌باشد و احتیاج به عملیاتی جهت ترمیم دارد (شکل ۵).



شکل ۵: درخت معیوب حاصل شده

همانطور که در شکل (۵) نمایش داده شده است سه حالت غیرموجه در جواب نمایش داده شده وجود دارد. اولی مربوط به حالت گردش بین گره‌ها و ایجاد یک سیکل درونی در آنها

می‌باشد که این در مورد گره‌های ۲، ۳ و ۴ می‌باشد. این بدان معناست که موادی که از گره ۲ شروع به حرکت می‌کنند به آن گره باز می‌گردند و این سیکل غیرمنطقی ادامه پیدا می‌کند. دومین شکل مربوط به حالت انعکاسی در گره‌های ۵ و ۹ است که در آن‌ها مواد از گره‌ای خارج و به همان گره باز می‌گردد که این مساله کدگذاری به روش درخت گسترده را غیرموجه می‌کند. و سومین حالت ناموجه در صورتی اتفاق می‌افتد که در گراف C عدد ۱ یا به عبارتی اولین حلقه از تسهیلات قرار نداشته باشد.

برای رفع این حالات ناموجه از نمایش جواب میبایست:

۱. از مقداردهی تکراری در یک رشته از جواب‌ها خودداری کرد یا به عبارتی برای محققینی که از برنامه متلب برای کدنویسی استفاده می‌کنند از تابع randperm برای تولید جواب‌ها استفاده شود.

۲. بعد از مقداردهی بررسی شود که مقدار هیچ آلل متناظری از ماتریس‌های C و C' یکسان نباشند.

۳. در صورتی که عدد یک در هیچ آلی از جواب مورد بررسی قرار داده نشده، به صورت تصادفی یکی از اجزای جواب با عدد یک جابجا شود.

همانطور که از شکل (۲) پیداست جواب اولیه ما دارای ۲ بخش می‌باشد: بخش اول مربوط به باز یا بسته بودن هر یک از تسهیلات می‌باشد و بخش دوم مربوط به میزان جریان مابین گره‌های شبکه می‌باشد که در ادامه به آن‌ها می‌پردازیم.

برای تشکیل دادن قسمت اول ماتریس که طول آن برابر است با $(j+k+M)$ یعنی برابر با مجموع مکان‌های بالقوه برای تاسیس کارخانه‌ها، مراکز توزیع و مراکز بازرسی، ابتدا با خانه‌های مربوط به آن‌ها را به صورت تصادفی با اعداد تصادفی صفر و یک پر کرد که در آن یک معنای باز بودن و صفر معنای بسته بودن آن تسهیل می‌باشد. قدم دوم این است که ببینیم که مجموع ظرفیت تسهیلات قادر به پاسخگویی به نیاز مشتریان می‌باشد یا خیر؟

میزان جریان‌ها در ماتریس نمایش جواب

در قسمت دوم از ژنوم، بر طبق فرضیات مدل پیشنهادی، محصولات باید تنها در سطوح غیر یکسان و متوالی شبکه جریان پیدا کند. بنابراین ما در ماتریس نمایش جواب‌ها باید قسمت‌هایی از ماتریس را محدود به اعداد خاص کنیم تا بدین وسیله ۲ مشکل ذکر شده در ماتریس نمایش جواب را ترمیم کنیم. به عنوان مثال اگر I تعداد تامین کنندگان و J تعداد کارخانه‌های ما باشد طول ماتریس جواب ما $I+J-1$ خواهد بود. برای $I-1$ خانه اول ماتریس ما باید اعداد $I+1$ تا $I+J$ را درون ماتریس به صورت تصادفی قرار دهیم و برای J خانه دوم ماتریس باید اعداد 1 تا I را قرار دهیم. بدین گونه ما می‌توانیم به هر دو مشکل ذکر شده در نمایش ماتریس پاسخ فائق آییم.

نمایش جواب کدگذاری به روش درخت گسترده شامل $I+J-1$ پل ارتباطی بین تامین کنندگان و کارخانه‌ها می‌باشد این در حالی است که تعداد واقعی خطوط ارتباط $I \times J$ می‌باشد. در واقع با این روش ما قادر خواهیم بود در حجم زمانی الگوریتم صرفه‌جویی کرده و از هزینه‌های ناشی از آن جلوگیری کنیم.

ترتیب مقداردهی در ماتریس جواب

در شبکه لجستیک حلقه بسته دو نوع سیستم حمل و نقل وجود دارد: سیستم فشاری^۱ و سیستم کششی^۲. سیستم فشاری برای شبکه رفت به کار می‌رود و مواد را از تامین کنندگان به مشتریان ارسال می‌کند سیستم کششی به صورت معکوس مواد را از مشتریان به مراکز کارخانه‌ها و یا مرکز انهدام می‌رساند. ترتیب مقداردهی به ماتریس جواب می‌بایستی ابتدا از سیستم کششی در جهت معکوس شروع شده و پس از آن با سیستم فشاری در جهت مستقیم ادامه یابد. بدین ترتیب ما می‌توانیم مقدار تقاضای کارخانه‌ها از تامین کنندگان را به درستی تعیین کنیم و تمام هزینه‌ها به صورت منطقی قابل محاسبه خواهند بود.

1 Pull

2 Push

تعیین میزان جریان‌ها و میزان تابع هدف

ماتریس جواب‌ها با استفاده از تابع هدف (تابع شایستگی^۱) ارزیابی می‌شوند. مسائل واقعی معمولاً همراه با محدودیت‌هایی هستند که در مدل‌سازی ظاهر می‌شوند. برای رسیدن به بهینه کلی باید فضای شدنی مسئله به طور وسیع جستجو شود. برای این منظور باید بین جواب‌های شدنی و غیر شدنی تعادل برقرار باشد، در غیر این صورت تنوع در جواب‌ها وجود نخواهد داشت و فقط روی قسمت محدودی از فضای شدنی متمرکز می‌شویم. در نتیجه بهینه بدست آمده محلی خواهد بود. نحوه این عمل در قسمت رمزگشایی از ماتریس جواب به صورت مفصل توضیح داده شده است.

ایده اولیه‌ای که بعدها مبنای رویکرد الگوریتم شبیه‌سازی ذوب قرار گرفت اولین بار توسط متروپلیس^۲ در ۱۹۵۳ بر اساس فرآیند سرمایش^۳ یا بازپخت^۴ مواد در علم ترمودینامیک آماری مطرح گردید.

استفاده از فرآیند سرمایش در مباحث بهینه‌سازی، اولین بار توسط کرک پاتریک^۵ در ۱۹۸۰ تحت عنوان شبیه‌سازی تبرید پیشنهاد گردید. رویکرد شبیه‌سازی ذوب جزء روش‌های جستجوی موضعی (همسایگی) می‌باشد که به دلیل پذیرش حرکت‌های غیر بهبوددهنده تابع هدف، برخلاف سایر روش‌های جستجوی موضعی به نقطه شروع (حل اولیه) وابسته نبوده و می‌تواند از دام بهینه‌های موضعی تا حد زیادی رهایی یابد. همچنین کیفیت حل بدست آمده به نرخ کاهش پارامتر دما^۶ (t) بستگی خواهد داشت. بر طبق قواعد ترمودینامیک آماری، احتمال افزایش انرژی سیستم به اندازه ΔE در دمای t برابر است با:

$$P(\Delta E) = e^{\frac{-\Delta E}{kt}} \quad (29)$$

-
- 1.Fitness Function
 - 2.Metropolis
 - 3.Cooling
 4. Annealing
 - 5.Kirkpatric
 - 6.Cooling Shedule

در شبیه‌سازی ذوب احتمال پذیرش حرکت‌های غیر بهبوددهنده در هر دما طبق رابطه فوق محاسبه می‌گردد. بطوریکه ΔE مبین مقدار تغییرات تابع هدف از حل جاری x به حل جدید x' می‌باشد. با کاهش مقدار دما، احتمال فوق نیز کاهش می‌یابد چرا که لازم است در دماهای پایین‌تر زمان بیشتری برای یافتن بهینه‌های موضعی صرف گردد.

$$\Delta E = E(x') - E(x) \quad \min E \text{ and } E(x') > E(x) \quad (30)$$

در حالت کلی برای کاهش دما از تابعی به فرم زیر استفاده می‌شود، بطوریکه T_r مبین مقدار دما در تکرار r ام الگوریتم می‌باشد:

$$T_{(r+1)} = \alpha T_r \quad 0 < \alpha < 1 \quad (31)$$

تعریف نمادها:

N	: تعداد جواب‌های پذیرفته شده در هر دما (معیار خروج از حلقه داخلی)
R	: حداکثر انتقالات دما (معیار توقف و خروج از حلقه خارجی)
T_0	: دمای اولیه
α	: ضریب کاهش دما
X	: جواب شدنی
$f(X)$: مقدار تابع هدف با ازای هر X
n	: شمارنده تعداد انتقالات دما
r	: شمارنده تعداد انتقالات دما

```

 $r = 0, T = T_0, X_{best} = \emptyset$ 
Generate  $X_0$ 
 $X_{best} = X_0$ 
Do(Out side loop)
 $n = 0$ 
while  $T < 10^{-10}$ 
Do insert and swap mutation  $X_n$  as :  $\Delta C = C(X_{new}) - C(X_{best})$ 
If  $\Delta C < 0$ 
 $X_{best} = X_{new}; n = n + 1; X_n = X_{new}$ 
Else
Generate  $y \rightarrow U(0,1)$  Randomly
Set  $Z = \text{Exp}(-\Delta C / Tr)$ 
if  $y < z$ 
 $n = n + 1; X_n = X_{new}$ 
End
End

```

شکل ۶: الگوریتم بهینه‌سازی شبیه‌سازی تبرید

در ادامه فعالیت‌های تحقیقاتی برای توسعه شبیه‌سازی تبرید، لو، یوآن و ژانگ^۱ (۲۰۰۹) روشی مبتنی بر تئوری ابر^۲ را ارائه نمودند که امکان جستجوی بهتر همسایگی‌ها و دستیابی به جواب‌های بهتر را فراهم نمود.

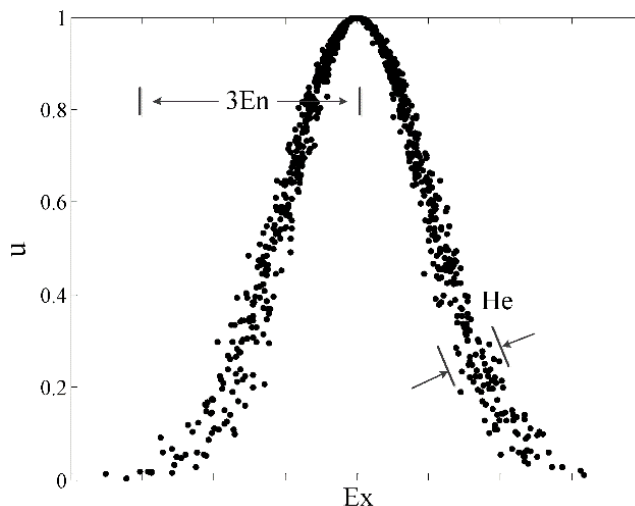
مدل ابری مدلی است که حاوی انتقال مفاهیم کیفی و نمایش کمی است که برای این منظور از زبان طبیعی بهره می‌گیرد. دیی و همکاران (۱۹۹۵).

اگر سه ویژگی دیجیتال (En, Ex, He) و یک μ_0 خاصی در نظر بگیریم، یک قطره قطره ابر (X_i, μ_0) می‌تواند توسط یک تولید کننده داده ابری به شرح زیر تولید شود که نام Y را بر آن قرار می‌دهیم.

-
1. Lu, Yuan and Zhang
 2. Cloud Theory

$INPUT : (Ex, En, He)$
 $Output : ((x_i, \mu_0), \dots, (x_n, \mu_0))$
 for $i = 1 : n$
 $En' = randn(En, He)$
 $X_i = Ex \pm En' \sqrt{-2 \ln(\mu_0)}$
 $drop(x_i, \mu_0)$
 Endfor

شکل ۷: فرایند ژنراتور y



شکل ۸: سه مشخصه یک پاسخ ابری (LV et al., 2009)

الگوریتم شبیه سازی مبتنی بر ابر قوانین متروپولیس را دنبال می کند و از تولید کننده ابر طبیعی Y برای تولید دما استفاده می کند که بر روند تولید و پذیرش جوابها تأثیر می گذارد که با قوانین فیزیکی مطابقت بیشتری دارد. شکل ۹ به مراحل انجام الگوریتم با توجه به شرایط y توضیح داده شده می پردازد.

```

Let  $T_0 = 0.1$ 
 $T = T_0, s = s_0, k = k_0$ 
While  $T > 10^{-10}$ 
   $He = T; En = T, \mu = 1 - T;$ 
   $En' = (En + He - 1 / 3rand(0,1));$ 
   $T' = En' \sqrt{-2 \ln(\mu_0)};$ 
   $L_1 = \text{Value of spanning tree based decoded fitness function};$ 
   $s' = \text{Do swap and insertion mutation randomly}$ 
   $L_2 = \text{Value of the new solution}$ 
  If  $L_2 \leq L_1$ 
     $S_0 = S';$ 
  Else
    with probability of  $P = e^{-\frac{\Delta}{T'}}$ 
  End
   $k = k + 1$ 
   $T = T_0 \lambda^k$ 
End While

```

شکل ۹: الگوریتم بهینه‌سازی شبیه‌سازی تبرید مبتنی بر ابر

تغییرات تصادفی دما پس از وارد کردن تئوری ابر می‌تواند تنوع افراد جستجو را بیش‌تر کرده و از افتادن الگوریتم به دام بهینه محلی به نسبت الگوری شبیه‌سازی تبرید معمولی بهتر جلوگیری کند. علاوه بر این، گرایش پایدار تغییرات دما می‌تواند جواب‌های بهتر را سریع‌تر تشخیص داده و در نتیجه کارایی الگوریتم تبرید را بالاتر ببرد. در واقع با تغییر الگوی تغییر دما در الگوریتم SA به احتمال زیاد شاهد سرعت و دقت بیشتری در حل مسائل NP-hard خواهیم بود که در ادامه به بررسی نتایج حاصل در مسئله طراحی شبکه حلقه بسته می‌پردازیم.

تحلیل نتایج

برای مسأله طراحی یکپارچه شبکه لجستیک مستقیم و معکوس ۲۰ نمونه مسأله آزمایشی از اندازه کوچک تا بزرگ تولید شده است. این مسائل با کمک مسائل مورد استفاده در مقالات مشابه تولید شده است. داده‌های دیگر مسأله نیز از مقاله وانگ^۱ و سو^۲ (۲۰۱۰) و یادگاری و همکاران (۲۰۱۵) استخراج شده است. (جداول ۲-۴)

جدول ۲: نمونه مسائل آزمایشی

شماره مساله	تأمین کنندگان	کارخانه‌ها	مراکز توزیع	مشتریان	مراکز بازرسی
۱	۳	۵	۳	۴	۲
۲	۶	۱۰	۶	۸	۴
۴	۱۲	۲۰	۱۲	۱۶	۸
۶	۱۸	۳۰	۱۸	۲۴	۱۲
۷	۲۱	۳۵	۲۱	۲۸	۱۴
۸	۲۴	۴۰	۲۴	۳۲	۱۶
۹	۲۷	۴۵	۲۷	۳۶	۱۸
۱۰	۳۰	۵۰	۳۰	۴۰	۲۰
۱۲	۳۶	۶۰	۳۶	۴۸	۲۴
۱۳	۴۰	۶۶	۴۰	۵۳	۲۶
۱۵	۴۶	۷۶	۴۶	۶۱	۳۰
۱۷	۴۹	۸۱	۴۹	۶۵	۳۲
۱۹	۵۵	۹۱	۵۵	۷۳	۳۶
۲۰	۵۸	۹۶	۵۸	۷۷	۳۸

1. Wang
2. Hsu

جدول ۳: مقادیر پارامترها

پارامتر	محدوده
ظرفیت تامین کنندگان	$\sim uniform(350, 700)$
ظرفیت کارخانه‌ها	$\sim uniform(300, 600)$
هزینه ثابت تامین کنندگان	$\sim uniform(800, 2300)$
ظرفیت مراکز توزیع/جمع‌آوری	$\sim uniform(600, 950)$
هزینه ثابت تاسیس مراکز توزیع/جمع‌آوری	$\sim uniform(800, 1800)$
تفاضل	$\sim uniform(300, 500)$
ظرفیت مراکز بازرسی	$\sim uniform(350, 800)$
هزینه ثابت تاسیس مراکز بازرسی	$\sim uniform(800, 1200)$
pl_m, pc_l, pd_k	٪۱۰

جدول ۴: محدوده هزینه‌های حمل و نقل در هر سطح

محدوده	هزینه حمل و نقل
$\sim uniform(2, 10)$	تأمین کننده - کارخانه
$\sim uniform(2, 9)$	کارخانه - توزیع کننده
$\sim uniform(2, 8)$	توزیع کننده - مشتری
$\sim uniform(2, 8)$	مشتری - توزیع کننده
$\sim uniform(1, 6)$	توزیع کننده - بازرسی
$\sim uniform(2, 10)$	بازرسی - کارخانه

در این بخش عملکرد الگوریتم شبیه‌سازی تبرید مبتنی بر ابر با عملکرد دو الگوریتم در ادبیات موضوع که از روش نمایش درخت پوشا برای حل مساله NP-hard استفاده کرده‌اند

مقایسه می‌شود. الگوریتم اول الگوریتم ژنتیک تطبیق داده شده در مقاله وانگ و سو (۲۰۱۰) و دوم مقاله شبیه سازی تبرید توسعه داده شده در مقاله یادگاری و همکاران (۲۰۱۵). سه الگوریتم فرا ابتکاری مورد بررسی از نظر کارایی و اثربخشی تجزیه و تحلیل می‌شوند. نتایج به دست آمده از اجرای الگوریتم‌ها بر روی مسائل داده شده، مربوط به بهترین مقدار تابع هدف در مساله طراحی شبکه حلقه بسته می‌باشد. این هزینه‌ها شامل هزینه‌های ثابت تاسیس و هزینه‌های حمل و نقل در شبکه می‌باشند. هر یک از الگوریتم‌ها، روی ۱۴ نمونه مسأله با اندازه‌های کوچک، متوسط و بزرگ، هر کدام به تعداد ۱۰ بار اجرا شده و پس از هر اجرا، نتایج به دست آمده مربوط به بهترین مقدار تابع هدف (کمترین هزینه) در ۲۵٪ زمان و کل زمان در نظر گرفته شده برای توقف الگوریتم‌ها و همچنین اولین زمان رسیدن به بهترین مقدار تابع هدف ثبت گردیده است. برای این منظور هر یک از الگوریتم‌ها با استفاده از نرم افزار MATLAB7.11.0 (R2014a) کدگذاری شده‌اند. تمامی مسائل روی کامپیوتری با پردازشگر Dual-core processor 2.20 Gh, 2.2 GHz و حافظه اصلی 3.00 GB و با استفاده از سیستم عامل Windows 7 Home اجرا شده‌اند. پس از اجرای الگوریتم‌های فرا ابتکاری بر روی مسائل در نظر گرفته شده، به منظور انجام تجزیه و تحلیل‌های آماری دقیق از روش آنالیز واریانس یک طرفه با استفاده از نرم افزار MINITAB 16 انجام شده است.

نتایج به دست آمده از اجرای سه الگوریتم فرا ابتکاری ژنتیک، و شبیه‌سازی تبرید (SA) و شبیه‌سازی تبرید مبتنی بر ابر بر روی ۱۴ مساله نمونه در اندازه‌های کوچک، متوسط و بزرگ، در جدول ۵ و ۶ ارائه شده است. نتایج به دست آمده از اجرای الگوریتم‌ها، از دو نقطه نظر بهترین مقدار تابع هدف، به عنوان معیار اثربخشی، اولین زمان رسیدن به بهترین مقدار تابع هدف و بهترین مقدار تابع هدف در ۲۵٪ اول کل بازه زمانی ذکر شده به عنوان معیار کارایی تجزیه و تحلیل می‌شوند.

در این تحقیق برای ارزیابی الگوریتم‌های فرا ابتکاری مختلف در سه دسته مسائل کوچک، متوسط و بزرگ، از معیار درصد انحراف نسبی^۱ (RPD) استفاده می‌شود. نحوه محاسبه آن در

1. Relative percentage deviation.

رابطه شماره (۴-۱) آورده شده است، در این رابطه Alg_{sol} نشان دهنده جواب به دست آمده توسط الگوریتم‌های توسعه داده شده و Min_{sol} کوچک‌ترین مقدار مورد نظر از هر اجرای الگوریتم‌ها در هر سه نمونه از مسائل در نظر گرفته شده می‌باشد. مقدار RPD نشان می‌دهد که جواب‌ها در هر الگوریتم تا چه اندازه از بهترین جواب به دست آمده فاصله دارند. هر چقدر این فاصله بیشتر باشد به این معنی است که الگوریتم‌ها جواب‌های بدتری تولید می‌کنند، و متقابلاً هر چقدر این فاصله کمتر باشد نشان می‌دهد که الگوریتم‌ها جواب‌های بدتری تولید می‌کنند و الگوریتم‌های مناسب‌تری هستند.

$$PRD = \frac{Alg_{sol} - Min_{sol}}{Min_{sol}} \times 100 \quad (32)$$

با توجه به معادله مورد نظر، نتایج به دست آمده از اجرای الگوریتم‌های مختلف در هر اجرا، توسط معیار درصد

انحراف نسبی به یک مقیاس مساوی برای تمام ۱۴ نمونه مساله در نظر گرفته شده، تبدیل می‌شوند. فرضیه مورد آزمایش در مورد برابری میانگین جواب‌های به دست آمده از نظر بهترین مقدار تابع هدف در بازه‌های زمانی تعیین شده اولین زمان رسیدن به بهترین جواب توسط چهار الگوریتم توسعه داده شده در اندازه‌های مختلف می‌باشد.

همان طور که در تعریف مساله بیان شد، در این تحقیق هدف حداقل کردن هزینه در نظر گرفته شده است. در نتیجه فرضیه اول در مورد برابری میانگین بهترین مقدار تابع هدف توسط سه الگوریتم توسعه داده شده در سه اندازه کوچک، متوسط و بزرگ، به صورت زیر تعریف می‌شود:

$$\left\{ \begin{array}{l} H_0: \mu_{GA(cost)} = \mu_{SA(cost)} = \mu_{CSA(cost)} \\ H_1: \text{حداقل یکی از الگوریتم‌ها میانگین متفاوتی با بقیه الگوریتم‌ها دارد} \end{array} \right. \quad (33)$$

در فرضیه اول، H_0 بیان می‌کند که میانگین به دست آمده از الگوریتم‌های توسعه داده شده تفاوت معناداری با یکدیگر ندارند، ولی H_1 بیان می‌کند که حداقل یکی از الگوریتم‌ها

میانگین متفاوتی با بقیه الگوریتم‌ها دارد. برای آزمودن این فرضیه از آنالیز واریانس یک طرفه^۱ استفاده شده است و نتایج حاصل از آن در شکل (۱۰) نشان داده شده است. با توجه به شکل (۱۰)، میزان p-value به دست آمده از نتایج آنالیز واریانس یک طرفه برابر صفر می‌باشد. عدد صفر بیان کننده میزان تاییده‌ای است که فرضیه H₀ هیچ تأییدیه‌ای از نمونه دریافت نکرده است، در نتیجه فرضیه H₁ رد می‌شود. همان گونه که مشخص است، در سطح اطمینان ۹۵٪ فرض H₀ رد و فرض H₁ تایید شده است. به این معنی که میان الگوریتم‌ها تفاوت معناداری از نظر بهترین مقدار تابع هدف وجود دارد. حال که مشخص شد میان الگوریتم‌ها تفاوت معناداری وجود دارد می‌بایست الگوریتم‌ها در مقایسه با هم ارزیابی شده و مشخص شود که میان کدام الگوریتم‌ها این تفاوت معناداری وجود دارد و به چه میزان است. همچنین الگوریتم‌ها می‌بایست از نظر اثر بخشی رتبه‌بندی شوند. در نتیجه برای تحلیل بیشتر و یافتن اختلاف معناداری الگوریتم‌ها از آزمون توکی^۲ استفاده شده است. آزمون توکی با گروه بندی الگوریتم‌ها به صورت دودویی، الگوریتم‌ها را از نظر تفاوت معناداری و میزان آن با یکدیگر مقایسه می‌کند.

در شکل‌های (۱۰) و (۱۱) خروجی حاصل از آنالیز واریانس و نمودار میانگین و فاصله اطمینان ۹۵٪ برای بهترین مقدار تابع هدف هدف در ۲۵٪ اول زمان حل جهت سنجش سرعت همگرایی الگوریتم‌ها در ابتدای کار هر یک از آن‌ها مورد بررسی قرار می‌گیرد. طبق این دو تحلیل انجام شده بر اساس این معیار الگوریتم‌های SA و CSA نسبت به الگوریتم GA برتری دارند و با سرعت بالاتری در ابتدای بازه زمانی مقایسه شده همگرا می‌شوند. اما بین الگوریتم‌های SA و CSA تفاوت معناداری دیده نمی‌شود.

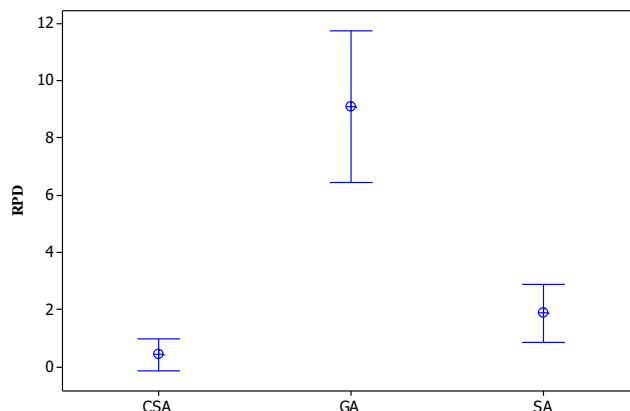
1. Oneway-ANOVA

2. Tukey's test

Source	DF	SS	MS	F	P
Factor	۰,۰۰۰	۳۶,۰۹	۳۰۱,۲۶	۶۰۲,۵۱	۲
Error	۸,۳۵	۳۲۵,۵۱	۳۹		
Total	۹۲۸,۰۲	۴۱			

S = 2.889 R-Sq = 64.92% R-Sq(adj) = 63.13%

شکل ۱۰: خروجی به دست آمده از آنالیز واریانس برای بهترین مقدار تابع هدف در ۲۵٪ اول زمان حل



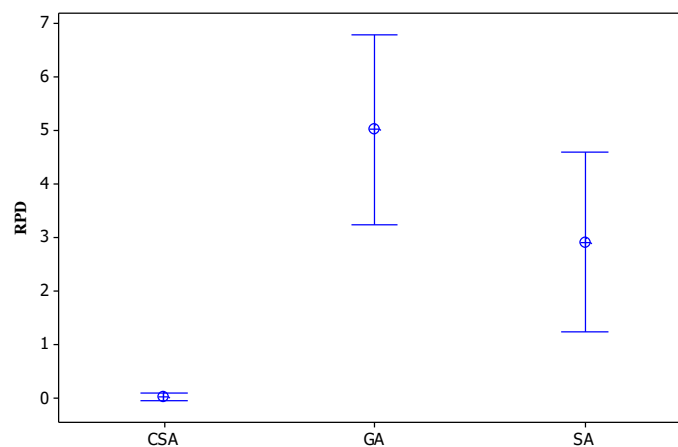
شکل ۱۱: نمودار میانگین و فاصله اطمینان ۹۵٪ برای بهترین مقدار تابع هدف در ۲۵٪ اول زمان حل

در شکل‌های (۱۲) و (۱۳) خروجی حاصل از آنالیز واریانس و نمودار میانگین و فاصله اطمینان ۹۵٪ برای بهترین مقدار تابع هدف پس از رسیدن به شروط توقف جهت سنجش اثربخشی الگوریتم‌ها مورد بررسی قرار می‌گیرد. طبق این دو تحلیل انجام شده فرضیه H₀ رد شده و بر اساس این معیار الگوریتم CSA نسبت به الگوریتم SA و الگوریتم SA نسبت به GA برتری دارد. برتری دارند و با سرعت بالاتری در ابتدای بازه زمانی مقایسه شده همگرا می‌شوند. اما بین الگوریتم‌های SA و CSA تفاوت معناداری دیده نمی‌شود.

Source	DF	SS	MS	F	P
Factor	۰,۰۰۰	۱۵,۱۶	۸۶,۷۰	۱۷۳,۴۱	۲
Error	۵,۷۲	۲۲۳,۰۸	۳۹		
Total	۳۹۶,۴۸	۴۱			

S = 2.392 R-Sq = 43.74% R-Sq(adj) = 40.85%

شکل ۱۲: خروجی به دست آمده از آنالیز واریانس برای بهترین مقدار تابع هدف در کل زمان حل



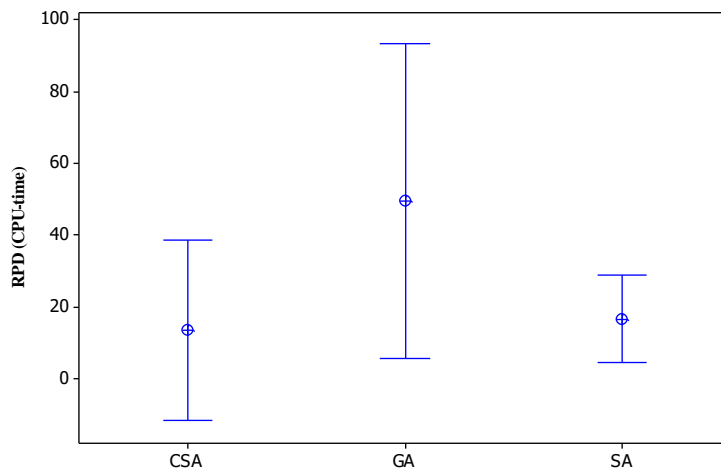
شکل ۱۳: نمودار میانگین و فاصله اطمینان ۹۵٪ برای بهترین مقدار تابع هدف در کل زمان حل

در شکل‌های (۱۴) و (۱۵) خروجی حاصل از آنالیز واریانس و نمودار میانگین و فاصله اطمینان ۹۵٪ برای اولین زمان رسیدن به بهترین جواب جهت سنجش کارایی الگوریتم‌ها مورد بررسی قرار می‌گیرد. طبق این دو تحلیل انجام شده و بر اساس این معیار الگوریتم CSA بین الگوریتم‌های SA، CSA و GA تفاوت معناداری دیده نمی‌شود.

Source	DF	SS	MS	F	P
Factor	۰,۳۹۹	۰,۹۴	۸۵۲۱۰,۴	۱۷۰۴۲,۰۸	۲
Error	۹۰۴۸۴۴	۳۵۲۸۸۹۱۴	۳۹		
Total	۳۶۹۹۳۱۲۲	۴۱			

S = 951.2 R-Sq = 4.61% R-Sq(adj) = 0.00%

شکل ۱۴: خروجی به دست آمده از آنالیز واریانس برای اولین زمان رسیدن به بهترین جواب



شکل ۱۵: نمودار میانگین و فاصله اطمینان ۹۵٪ برای اولین زمان رسیدن به بهترین جواب

نتایج آزمون توکی نشان داد که الگوریتم CSA با دیگر الگوریتم‌های SA و GA از لحاظ بهترین میزان تابع هدف اختلاف معناداری دارد، اما الگوریتم GA و SA با یکدیگر اختلاف معناداری ندارند. از طرفی این آزمون در مورد بهترین مقدار تابع هدف در ۲۵٪ اول زمان حل اجرا شد که طی آن مشخص شد که الگوریتم GA از این لحاظ با دو الگوریتم دیگر اختلاف معناداری دارد و از طرفی در این مورد دو الگوریتم SA و CSA اختلاف معناداری ندارند. در نتیجه می‌توان به این نتیجه رسید که الگوریتم CSA و SA از لحاظ سرعت همگرایی در ابتدای اجرای الگوریتم‌ها در وضعیت بهتری نسبت به GA قرار دارند.

نتیجه‌گیری

در این مقاله پس از توضیحاتی درباره مدل طراحی شبکه زنجیره تامین حلقه بسته و بحث در مورد ضرورت و اهمیت مساله به بررسی آثار و مقالات موجود در این زمینه پرداخته و نیازهای تحقیقاتی برگرفته از این بررسی شناسایی شده و با در نظر گرفتن انواع ویژگی‌ها و شرایط دنیای واقع الگوریتم جدیدی در این حوزه با روش نمایش پیچیده درخت پوشا جهت

بهبود زمان حل کمتر و جواب دقیق‌تر در مسائلی با اندازه‌های مختلف ارائه شد. برای ارزیابی کیفیت الگوریتم طراحی شده، مسایل آزمایشی در ابعاد مختلف و با مشخصات مختلف بطوریکه شرایط مختلف دنیای واقعی را شبیه سازی کند طراحی شدند. این مسائل توسط سه الگوریتم ژنتیک (برگرفته از ادبیات)، شبیه‌سازی ذوب (برگرفته از ادبیات) و الگوریتم شبیه‌سازی ذوب مبتنی بر ابر حل شدند. در ادامه این الگوریتم‌ها هم از لحاظ کیفیت جواب‌های بدست آمده و هم از نظر زمان حل در ابعاد کوچک، متوسط و بزرگ با یکدیگر مقایسه گردیدند. نتایج حاصل از این مقایسه در سطح اطمینان ۹۵٪ عبارتند از:

- در ۲۵٪ اول زمان توقف:

GA-2 SA-1 CSA-1

در واقع کیفیت جواب‌های الگوریتم ارائه شده از الگوریتم GA در ادبیات بهتر و نسبت به الگوریتم SA موجود در ادبیات تفاوت معناداری نداشت.

- از نقطه نظر اولین زمان رسیدن الگوریتم‌ها به جواب بهینه در سطح اطمینان ۹۵٪ الگوریتم:

GA-1 SA-1 CSA-1

در نتیجه از نظر سرعت همگرایی الگوریتم‌ها تفاوت معناداری بین الگوریتم SA و CSA وجود نداشت اما هر دو الگوریتم‌های سریعتری نسبت به GA بودند.

- از نقطه نظر بهترین میزان تابع هدف در کل بازه زمانی مطرح شده در سطح اطمینان ۹۵٪ الگوریتم:

GA-3 SA-2 CSA-1

در نتیجه از نظر اثربخشی الگوریتم‌ها تفاوت معناداری بین الگوریتم هر سه الگوریتم وجود داشت و CSA عملکرد بهتری نسبت به بقیه الگوریتم‌ها از خود نشان داد.

در جهت گسترش تحقیقات آتی در این حوزه پیشنهاد می‌شود روش نمایش درخت پوشا که از آرایه‌های کمتری برای نمایش جواب‌های الگوریتم‌های فراابتکاری در حوزه طراحی شبکه

بهره می‌گیرد را در الگوریتم‌های فراابتکاری دیگر مورد استفاده قرار گیرد. همچنین روش - های فراابتکاری جدید که کارایی خود را در مدل‌های بهینه‌سازی ترکیباتی نشان داده‌اند در فضای طراحی شبکه‌های حلقه بسته که به نسبت دیگر حوزه‌ها استفاده از این الگوریتم‌ها کمتر در آن‌ها مورد توجه بوده، استفاده شود.

منابع

- Aras, G & ,Crowther, D. (2008). (Governance and sustainability: An investigation into the relationship between corporate governance and corporate sustainability. *Management Decision*, 46. ۴۴۸-۴۳۳, (۳)
- Devika, K ,Jafarian, A & ,Nourbakhsh, V. (2014). Designing a sustainable closed-loop supply chain network based on triple bottom line approach: A comparison of metaheuristics hybridization techniques. *European Journal of Operational Research*, 235. ۶۱۵-۵۹۴, (۳)
- Deyi, L ,Haijun, M & ,Xuemei, S. (1995). (Membership clouds and membership cloud generators [J]. *Journal of Computer Research and Development*, 6 .
- Elhedhli, S & ,Merrick, R. (2012). Green supply chain network design to reduce carbon emissions. *Transportation Research Part D: Transport and Environment*, 17. ۳۷۹-۳۷۰, (۵)
- Fleischmann ,M., Beullens, P., BLOEMHOF-RUWAARD, J. M & , Wassenhove, L. N. (۲۰۰۱). The impact of product recovery on logistics network design. *Production and Operations Management*, 10. ۱۷۳-۱۵۶, (۲)
- Ghayebloo, S ,Tarokh, M .J., Venkatadri, U & ,Diallo, C. (2015). (Developing a bi-objective model of the closed-loop supply chain network with green supplier selection and disassembly of products: the impact of parts reliability and product greenness on the recovery network. *Journal of Manufacturing Systems*, 36. ۸۶-۷۶ ,
- Gírio, F. M ,Fonseca, C., Carvalheiro, F., Duarte, L. C., Marques, S & , Bogel-Lukasik ,R. (2010). Hemicelluloses for fuel ethanol: a review . *Bioresource technology*, 101. ۴۸۰۰-۴۷۷۵, (۱۳)
- Govindan, K ,Fattahi ,M & ,Keyvanshokoh, E. (2017). Supply chain network design under uncertainty: A comprehensive review and future research directions. *European Journal of Operational Research* .
- Govindan, K ,Khodaverdi, R & ,Jafarian, A. (2013). A fuzzy multi criteria approach for measuring sustainability performance of a supplier based on triple bottom line approach. *Journal of Cleaner Production* . ۳۵۴-۲۴۵, ۴۷ ,
- Jayaraman, V ,Gupta, R & ,Pirkul, H. (2003). Selecting hierarchical facilities in a service-operations environment. *European Journal of Operational Research*, 147. ۶۲۸-۶۱۳, (۳)

Jayaraman, V & Pirkul, H. (2001). Planning and coordination of production and distribution facilities for multiple commodities. *European Journal of Operational Research*, 133. ۴۰۸-۳۹۴, (۲)

Kaya, O & Urek, B. (2016). A mixed integer nonlinear programming model and heuristic solutions for location, inventory and pricing decisions in a closed loop supply chain. *Computers & Operations Research*, 65. ۱۰۳-۹۳,

Ko, H. J & Evans, G. W. (2007). A genetic algorithm-based heuristic for the dynamic integrated forward/reverse logistics network for PLs. *Computers & Operations Research*, 34. ۳۶۶-۳۴۶, (۲)

Krikke, H, van Harten, A & Schuur, P. (1999). Business case Oco: reverse logistic network re-design for copiers. *OR-Spektrum*. ۴۰۹-۳۸۱, (۳)۲۱,

Lee, D.-H & Dong, M. (2008). A heuristic approach to logistics network design for end-of-lease computer products recovery. *Transportation Research Part E: Logistics and Transportation Review*, 44. ۴۷۴-۴۵۵, (۳)

Li, J., Chen, J & Wang, S. (2011). Introduction *Risk Management of Supply and Cash Flows in Supply Chains* pp. 1-48): Springer.

Lu, Z & Bostel, N. (2007). A facility location model for logistics systems including reverse flows: The case of remanufacturing activities. *Computers & Operations Research*, 34. ۳۲۳-۲۹۹, (۲)

LV, P., Yuan, L & Zhang, J. (2009). Cloud theory-based simulated annealing algorithm and application. *Engineering Applications of Artificial Intelligence*, 22. ۷۴۹-۷۴۲,

Min, H & Ko, H.-J. (2008). The dynamic design of a reverse logistics network from the perspective of third-party logistics service providers. *International Journal of Production Economics*, 113. ۱۹۲-۱۷۶, (۱)

Pishvaei, M. S & Razmi, J. (20۱۲). Environmental supply chain network design using multi-objective fuzzy mathematical programming. *Applied Mathematical Modelling*, 36. ۳۴۴۶-۳۴۳۳, (۸)

Salema, M. I. G., Póvoa, A. P. B & Novais, A. Q. (2009). A strategic and tactical model for closed-loop supply chains. *OR spectrum*, 31. ۵۹۹-۵۷۳, (۳)

Syarif, A, Yun, Y & Gen, M. (2002). Study on multi-stage logistic chain network: a spanning tree-based genetic algorithm approach. *Computers & Industrial Engineering*, 43. ۳۱۴-۲۹۹, (۱)

Syarif, I, Prugel-Bennett, A & Wills, G. (2012). *(Unsupervised clustering approach for network anomaly detection)*. Paper presented at the International Conference on Networked Digital Technologies.

Tsiakis, P & ,Papageorgiou, L. G. (2008). Optimal production allocation and distribution supply chain networks .*International Journal of Production Economics*, 111 . ۴۸۳-۴۶۸ ,(۲)

Wang, H.-F & ,Hsu, H.-W. (2010). A closed-loop logistic model with a spanning-tree based genetic algorithm .*Computers & Operations Research*, 3 . ۳۸۹-۳۷۶ ,(۲)۷

Yadegari, E ,Ekhtiari, M., Zandieh, M & ,Alem-Tabriz, A. (2014). An Artificial Immune Algorithm for a Closed-Loop Supply Chain Network Design Problem with Different Delivery Paths .*International Journal of Strategic Decision Sciences) IJSDS*, 5 . ۴۶-۲۷ ,(۳)

Yadegari, E ,Najmi, H., Ghomi-Avili, M & ,Zandieh, M. (2015). A Flexible Integrated Forward/Reverse Logistics Model with Random Path-based Memetic Algorithm .*Iranian Journal of Management Studies*, 8 . ۲۸۷ ,(۲)

Yadegari, E ,Zandieh, M & ,Najmi, H. (2015). A hybrid spanning tree-based genetic/simulated annealing algorithm for a closed-loop logistics network design problem .*International Journal of Applied Decision Sciences*, 8 . ۴۲۶-۴۰۰ ,(۴)

Yi, P., Huang ,M., Guo, L & ,Shi, T .(۲۰۱۶) .A retailer oriented closed-loop supply chain network design for end of life construction machinery remanufacturing .*Journal of Cleaner Production*, 124 . ۲۰۳-۱۹۱ ,