

# ارائه‌ی یک مدل ریاضی جدید به منظور تخصیص وسایل نقلیه به سکوها و مسیریابی وسایل نقلیه به طور هم‌زمان در انبار عبوری با پنجره‌ی زمانی نرم

مهدی علینقیان\* (استادیار)

البرز حسن‌زاده (کارشناس ارشد)

علی زینل همدانی (استاد)

دانشکده‌ی مهندسی صنایع و سیستم‌ها، دانشگاه صنعتی اصفهان

مهندسی صنایع و مدیریت شریف، تابستان ۱۳۹۸  
دوره ۱-۳۵، شماره ۱/۲، ص. ۷۵-۸۴

کلیدی‌ترین عملیات‌ها در انبارهای عبوری، تخصیص سکوها به وسایل نقلیه و مسیریابی وسایل نقلیه است. در نظر گرفتن این دو موضوع به طور هم‌زمان باعث کاهش هزینه‌ها به طور چشم‌گیری می‌شود. در این مقاله این دو موضوع به طور هم‌زمان بررسی شده‌اند و یک مدل ریاضی عدد صحیح مختلط برای این مسئله ارائه شده است. با توجه به اهمیت زمان بازدید مشتریان در مدل پیشنهادی برای مشتریان پنجره‌ی زمانی نرم نیز در نظر گرفته شده است. در ادامه با توجه به NP-Hard بودن مسئله‌ی مطرح شده یک الگوریتم شبیه‌سازی تریید جدید ارائه شده است. به منظور بررسی عملکرد الگوریتم پیشنهادی نتایج حاصل در ابعاد کوچک با نتایج حاصل از روش دقیق و الگوریتم جست‌وجوی ممنوعه مقایسه شد. در مسائل با ابعاد بزرگ نیز نتایج حاصل از الگوریتم پیشنهادی با الگوریتم جست‌وجوی ممنوعه مقایسه شدند. نتایج نشان‌دهنده‌ی عملکرد مناسب الگوریتم پیشنهادی برای مسئله‌ی مطرح است.

واژگان کلیدی: الگوریتم جست‌وجوی ممنوعه، الگوریتم شبیه‌سازی تریید، انبار عبوری، تخصیص سکوها، مسیریابی وسایل نقلیه با پنجره‌ی زمانی نرم.

## ۱. مقدمه

به جابایی امکانات و تجهیزات داخل انبار عبوری، تخصیص سکوها به وسایل نقلیه، حمل‌ونقل خارجی، زمان‌بندی وسایل نقلیه، زمان‌بندی منابع و نیروی داخل انبار عبوری و در نهایت زمان‌بندی و مدیریت بارگیری و بارگذاری کالا در انبار عبوری است.<sup>[۱]</sup>

همان‌طور که ذکر شد یکی از فعالیت‌های کلیدی در داخل انبار عبوری مسئله‌ی تخصیص سکوها به وسایل نقلیه است. مسئله‌ی دیگر مسئله‌ی کمیته‌سازی انتقال مواد داخل انبار عبوری است. تخصیص مناسب سکوها به وسایل نقلیه و به سکوها دیگر از یک سو منجر به کاهش حجم انتقال مواد داخل انبار میانی می‌شود و از سوی دیگر باعث کاهش آشفته‌گی در انبار میانی می‌شود. تخصیص نامناسب وسایل نقلیه به سکوها سبب می‌شود که هزینه‌های جابه‌جایی کالا در انبار عبوری افزایش یابد و زمان انتظار کالا در انبار نیز افزایش یابد. در نظر گرفتن مسئله‌ی تخصیص سکو به وسایل نقلیه بدون توجه به مسیریابی وسایل نقلیه باعث پیدا شدن مسئله‌ی زیربهمینه می‌شود؛ زیرا بدون در نظر گرفتن مسیره‌های پیموده شده مقدار و نوع کالای حمل‌شده توسط وسیله‌ی نقلیه مشخص نمی‌شود و در نتیجه تخصیص صحیح

امروزه در دنیای رقابتی توزیع کالا، شرکت‌ها با کاهش هزینه‌ها در هر قدم از عملیات سعی می‌کنند کل هزینه‌ها را کاهش دهند. از جمله‌ی این هزینه‌ها، هزینه‌ی حمل‌ونقل است.<sup>[۱]</sup> از طرف دیگر مشتریان تقاضای خدمات بهتر و سریع‌تری دارند که بارگیری و انتقال سریع‌تر و زمان‌بندی شده کالاها روشی در پاسخ به این موضوع است. یکی از روش‌های بدیع در زمینه‌ی انبارداری که منجر به کاهش هزینه‌ها می‌شود، استفاده از انبار عبوری است. انبار عبوری، انباری برای سرعت بخشیدن به عملیات توزیع کالا است که معمولاً کالاها کمتر از ۲۴ ساعت در آن انبار می‌شوند. در این نوع انبارها تعدادی سکو برای بارگیری و بارگذاری کالا وجود دارد. از طرف دیگر انبار عبوری نقطه‌ی قوتی در جریان حرکت و انتقال مواد محسوب می‌شود. تصمیمات در یک انبار عبوری به چند گروه کلی تقسیم می‌شوند. این تصمیمات شامل، تصمیمات مربوط به مکان‌یابی انبار عبوری در یک شبکه‌ی زنجیره‌ی تأمین، تصمیمات مربوط

\* نویسنده مسئول

تاریخ: دریافت ۱۳۹۶/۶/۲۵، اصلاحیه ۱۳۹۷/۲/۵، پذیرش ۱۳۹۷/۳/۱۳.

DOI:10.24200/J65.2018.7184.1749

alinaghian@cc.iut.ac.ir  
a.hassanzadeh@in.iut.ac.ir  
hamadani@cc.iut.ac.ir

وسایل به سکوها امکان پذیر نیست. از سوی دیگر عدم توجه به مسیریابی وسایل نقلیه و توجه صرف به مسئله تخصیص وسایل نقلیه به سکوها می تواند منجر به ایجاد مسیرهای نامناسب شود. همچنین توجه به زمان مورد انتظار مشتریان برای بازدید از دیگر موارد پراهمیت در مسیریابی است. معمولاً مشتریان یک پنجره‌ی زمانی از قبل مشخصی را برای شرکت توزیع مشخص می کنند. تخطی از بازه‌ی زمانی مشخص شده منجر به کاهش رضایت مشتریان یا اعمال جریمه به شرکت توزیع می شود. در پیشینه‌ی موضوع انبار عبوری، محققان در موارد مختلف و متعددی به کاهش هزینه‌های عملیاتی انبار عبوری و هزینه‌های مسیریابی، به طور مجزا پرداختند. مقالات متعددی ثابت کردند که این مسائل از نوع NP-hard هستند. پس حل مسائل مطرح توسط الگوریتم‌های فراابتکاری یک راه حل معقول است.

در این مقاله دو موضوع کلیدی تخصیص سکو به وسایل نقلیه و مسیریابی وسایل نقلیه در انبار عبوری به طور هم زمان با وجود پنجره‌ی زمانی نرم بررسی می شوند نوآوری‌های این پژوهش به طور خلاصه به شرح زیر است: یک مدل ریاضی یک پارچه به منظور بهینه‌سازی هم زمان مسئله تخصیص سکوها و مسیریابی وسایل نقلیه همراه با پنجره‌ی زمانی، ارائه‌ی دو روش فراابتکاری مبتنی بر الگوریتم فراابتکاری شبیه‌سازی تریید جستجوی ممنوعه برای حل مسئله در ابعاد بزرگ، و در نهایت بررسی الگوریتم‌های ارائه شده در مسائل نمونه در ابعاد کوچک و بزرگ در مقایسه با روش حل دقیق.

در ادامه‌ی مقاله در ابتدا در بخش دوم پیشینه‌ی موضوع مسئله ارائه می شود. سپس در بخش سوم به تعریف مسئله پرداخته و مدل ریاضی یک پارچه ارائه می شود. بخش چهارم به روش حل پیشنهادی تخصیص یافته است. در بخش پنجم نتایج حاصل از حل مسائل نمونه در ابعاد کوچک و بزرگ بررسی می شود و در نهایت بخش ششم به نتیجه‌گیری تخصیص یافته است.

## ۲. مرور پیشینه‌ی موضوع

تعدادی از محققان به مسئله تخصیص سکو در انبارهای عبوری پرداخته‌اند. آه و همکاران<sup>[۲]</sup> مسئله تخصیص سکوها را برای یک مرکز پست در کره در نظر گرفتند و برای حل مسئله از یک الگوریتم ابتکاری با سه فاز و یک الگوریتم ژنتیک استفاده کردند. میاو و همکاران<sup>[۳]</sup> یک مسئله تخصیص سکو در یک انبار عبوری را با در نظر گرفتن هزینه‌های عملیاتی داخل انبار عبوری در نظر گرفته و مسئله را با دو الگوریتم فراابتکاری، الگوریتم جستجوی ممنوعه و الگوریتم ژنتیک حل کردند. کنور و گلیاس<sup>[۴]</sup> مسئله‌ی زمان بندی وسایل نقلیه را برای سکوهای ورودی در یک انبار عبوری بررسی کردند. در مقاله‌ی آن‌ها زمان رسیدن وسایل نقلیه به سکوهای بارگیری مشخص و قطعی نیست. میائو و همکاران<sup>[۵]</sup> یک مدل خطی برای مسئله تخصیص سکو ارائه کردند و مسئله را با یک الگوریتم جستجوی ممنوعه تطابقی حل کردند. در سال ۲۰۱۴ مدنی اصفهانی و همکاران<sup>[۶]</sup> یک مسئله با چند انبار عبوری را در نظر گرفتند و برای آن یک مدل برنامه‌ریزی عدد صحیح مختلط غیرخطی ارائه کردند. هدف کمینه کردن کل زمان‌های عملیاتی یا بیشینه کردن خروجی انبار عبوری بود. برای حل مسئله نیز از دو الگوریتم فراابتکاری کرم شب‌تاب و الگوریتم شبیه‌سازی تبرید بهره برده شده است. با تمرکز بر اهمیت هماهنگی سکوهای بارگیری و بارگذاری در مراکز انتقال و انبارهای عبوری، کلار و همکاران<sup>[۸]</sup> یک مسئله هماهنگی بین سکوها در یک انبار عبوری را بررسی کردند.

اندازه‌ی دسته‌ها، جفت کردن سکوهای ورود و خروج و تصمیم در مورد نگه‌داری موجودی از جمله مسائلی بود که در این مقاله بررسی شد. با توجه به پیچیدگی مسئله، آنها یک الگوریتم ابتکاری گسترش دادند.

نیکولوپولو و همکاران<sup>[۹]</sup> در یک مقاله در سال ۲۰۱۷ به بررسی تفاوت توزیع مستقیم و استفاده از انبارهای عبوری در شبکه‌های توزیع پرداختند. برای حل و بررسی تفاوت راهبردهای مختلف، آنها یک الگوریتم جستجوی محلی گسترش دادند و برای بررسی و ارزیابی کیفیت الگوریتم از مسائل نمونه‌ی موجود در پیشینه‌ی مسئله استفاده کردند. در نهایت یک مدل ترکیبی برای استفاده از توزیع مستقیم و انبار عبوری ارائه شده است.

گلاره و همکاران<sup>[۱۰]</sup> برای مسئله تخصیص سکو در انبار عبوری یک روش حل دقیق ارائه کرده‌اند. برای حل مسئله آنها یک الگوریتم شاخه و برش گسترش داده‌اند.

مولوی و همکاران<sup>[۱۱]</sup> یک الگوریتم ژنتیک ترکیبی با الگوریتم جستجوی محلی برای حل مسئله‌ی زمان بندی کامیون در انبارهای عبوری ارائه کردند. از فرضیات آنها داشتن پنجره‌ی زمانی برای تحویل کالا و موعد تحویل ثابت است. نتایج محاسباتی این تحقیق نشان می دهد که تعیین بهینه‌ی پنجره‌ی زمانی به بهبود و بهینه‌سازی عملیات در انبار عبوری کمک می کند.

گروه دیگری از پژوهشگران مسئله‌ی مسیریابی وسایل نقلیه در انبار عبوری را بررسی کرده‌اند. لیو و همکاران<sup>[۱۲]</sup> بر اهمیت استفاده از انبارهای عبوری برای کاهش هزینه‌های موجودی تأکید کرده‌اند و مسئله‌ی مسیریابی وسایل نقلیه را در نظر گرفتند.

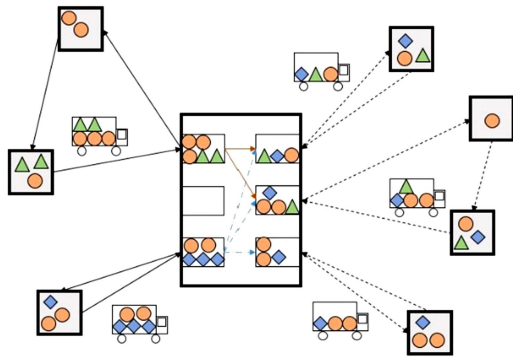
آن‌ها بیان کردند که بر خلاف اکثر تحقیقات گذشته که بر جنبه‌ی راهبردی مسائل انبارهای عبوری تمرکز کرده‌اند، تأکید آن‌ها بر جنبه‌ی عملیاتی کاهش هزینه‌ها در انبار عبوری است.<sup>[۱۳]</sup> مسئله‌ی مسیریابی وسایل نقلیه در انبار عبوری را بررسی کردند. هدف آن‌ها کمینه کردن هزینه‌های عملیاتی، هزینه‌های حمل و نقل، ارائه‌ی یک توالی بهینه از وسایل نقلیه و ارائه‌ی تعداد بهینه وسایل نقلیه است. موسوی و همکاران<sup>[۱۴]</sup> در سال ۲۰۱۳، یک مسئله مکان‌یابی و مسیریابی وسایل نقلیه را در

یک انبار عبوری در نظر گرفتند و برای این مسئله یک مدل برنامه‌ریزی عدد صحیح غیرخطی مختلط ارائه کردند. در مقاله‌ی آن‌ها بیان می شود که این دسته از مسائل در گروه مسائل NP-hard هستند. در نهایت مسئله با دو الگوریتم فراابتکاری حل شد. داندو و سردا<sup>[۱۵]</sup> مسئله‌ی مسیریابی تخلیه و بارگیری در یک انبار عبوری را به طور هم زمان در نظر گرفتند. آنها یک مدل ریاضی برای مسئله ارائه کردند؛ سپس بر اساس الگوریتم ابتکاری جابجایی محدودیت‌هایی به مدل اضافه کردند. این

تغییرات سبب افزایش سرعت و رسیدن به جواب‌های نزدیک به بهینه شد. موسوی و همکاران<sup>[۱۶]</sup> در مقاله‌ی دیگر مسئله‌ی مکان‌یابی و مسیریابی وسایل نقلیه را در یک انبار عبوری در شرایط فازی بررسی و یک مدل برنامه‌ریزی غیرخطی عدد صحیح مختلط برای مسئله ارائه می کنند. موراس و همکاران<sup>[۱۷]</sup> در مقاله‌ی که در سال

۲۰۱۴ منتشر شد یک مسئله‌ی مسیریابی در انبار عبوری را در نظر گرفته و به حل آن پرداخته‌اند. هدف آن‌ها پیدا کردن مسیرهای بهینه برای وسایل نقلیه بود. مقدم و همکاران<sup>[۱۸]</sup> مسئله‌ی مسیریابی و زمان بندی را در یک انبار عبوری با در نظر گرفتن امکان شکست تقاضا برای مشتریان در نظر گرفتند. در هنگام مسیریابی هنگامی که حجم بار هر مشتری نسبت به ظرفیت وسیله نقلیه بزرگ باشد شکست بار مشتری و حمل آن با تعدادی از وسایل نقلیه سبب کاهش تعداد وسایل نقلیه مورد

استفاده می شود. آگوستینا و همکاران<sup>[۱۹]</sup> مسئله‌ی زمان بندی و مسیریابی برای یک انبار عبوری برای غذا را بررسی و بیان کردند که استفاده از انبار عبوری برای کالاهایی مناسب است که زمان نگه‌داری آن‌ها باید کم باشد. آن‌ها مجموع هزینه‌ها که شامل



شکل ۱. مسئله‌ی مسیریابی و تخصیص در یک انبار عبوری.

$K$ : مجموعه‌ی تمام وسایل نقلیه  $K_1 \subset K$  مجموعه‌ی وسایل نقلیه‌ی پخش؛

$K_2 \subset K$ : مجموعه‌ی وسایل جمع‌آوری،  $k$  اندیس مربوط؛

$L$ : مجموعه‌ی تمام سکوها،  $l, q$  اندیس‌های مربوط به سکوها؛

$R$ : مجموعه‌ی کالاها و  $R$  اندیس مربوط.

### ۲.۳. پارامترهای مسئله

$C_{ij}$ : هزینه‌ی عبور از  $j - i$ ؛

$t_{ij}$ : زمان مورد نیاز برای رفتن از گره  $i$  به گره  $j$ ؛

$D_{lq}$ : فاصله‌ی سکوی  $l$  ام از سکوی  $q$  ام؛

$dp_{ir}$ : میزان تولید (تقاضای) کالای نوع  $r$  توسط تأمین‌کننده‌ی (مشتری)  $i$  ام؛

$CD_r$ : هزینه‌ی انتقال یک واحد کالای نوع  $r$  به میزان یک واحد مسافت داخل

انبار عبوری؛

$CP_k$ : ظرفیت وسیله‌ی نقلیه‌ی  $k$  ام؛

$ub_j$ : حد بالا پنجره‌ی زمانی مشتری  $j$ ؛

$lb_j$ : حد پایین پنجره‌ی زمانی مشتری  $j$ ؛

$Pl_j$ : جریمه‌ی یک واحد زودکرد مربوط به مشتری  $j$  ام؛

$Pu_j$ : جریمه‌ی یک واحد زودکرد مربوط به مشتری  $j$  ام.

### ۳.۳. متغیرهای تصمیم

$X_{ij}^k$ : اگر یال  $j - i$  با وسیله‌ی نقلیه‌ی  $k$  ام پیموده شود ۱ و در غیر این صورت ۰؛

$Z_i^k$ : اگر گره  $i$  ام به وسیله‌ی نقلیه‌ی  $k$  ام تخصیص یابد ۱ و در غیر این صورت ۰؛

$Y_q^k$ : اگر وسیله‌ی نقلیه‌ی  $k$  ام به سکوی  $q$  ام تخصیص یابد؛

$N_{lqr}$ : کل میزان کالای ارسال شده‌ی نوع  $r$  ام از سکوی  $l$  ام به سکوی  $q$  ام؛

$U_i^k$ : برای حذف زیرتورها؛

$AT_j$ : زمانی که به گره  $j$  سرویس داده می‌شود؛

$Vu_j$ : مقدار دیرکرد مربوط به مشتری  $j$  ام؛

$Vl_j$ : مقدار زودکرد مربوط به مشتری  $j$  ام.

### ۴.۳. مدل ریاضی پیشنهادی

در این قسمت مدل ریاضی برنامه‌ریزی غیرخطی مسئله با توضیح محدودیت‌ها و

تابع هدف ارائه می‌شود.

هزینه‌ی نگه‌داری موجودی و هزینه‌ی تحویل کالا، هزینه‌ی تأخیر یا زودکرد در تحویل است را کمیته کردند. بنی عامریان و همکاران<sup>[۲۰]</sup> مسیریابی برای یک انبار عبوری را با یک روش فرا ابتکاری ترکیبی حل کردند. آنها یک الگوریتم فراابتکاری ترکیبی از الگوریتم ژنتیک و الگوریتم وی‌ان‌اس ارائه دادند. نتایج نشان داد که الگوریتم ترکیب از دو الگوریتم پایه در حل مسئله عملکرد بهتری دارد.

مکمون و لاپورته به مسئله‌ی مسیریابی در چندین انبار عمومی پرداختند. آنها در مسئله‌ی خود می‌بایست هماهنگ‌سازی انبار عمومی را در نظر گرفتند و امکان بازدید یک وسیله‌ی نقلیه از بیش از یک انبار عمومی را در نظر گرفتند. در نهایت آنها مسئله‌ی مطرح را با یک الگوریتم فراابتکاری حل کردند.<sup>[۲۱]</sup>

با توجه به تأثیر مسیریابی پیموده شده توسط وسایل بر کالاهای حمل شده توسط آنها و در نتیجه تأثیر کالاهای حمل شده بر تخصیص سکوها به وسایل نقلیه، عدم توجه یک پارچه به مسیریابی وسایل و تخصیص وسایل نقلیه به سکوها باعث رسیدن به جواب‌های زیر بهینه خواهد شد. در این مقاله یک مدل ریاضی خطی یک پارچه برای مسئله‌ی مسیریابی و تخصیص سکوها برای انبارهای عبوری ارائه شده است.

### ۳. تعریف مسئله

در مسئله‌ی مطرح یک انبار عبوری و تعدادی گره شامل تولیدکنندگان و مشتریان وجود دارد. همچنین تعدادی وسیله نقلیه برای بارگیری و بارگذاری وجود دارند. هدف ارائه‌ی یک تخصیص بهینه از مشتریان و تولیدکنندگان به وسایل نقلیه و همچنین تخصیص وسایل نقلیه به سکوها به گونه‌ی است که هزینه‌های حمل‌ونقل و ازدحام داخل انبار عبوری و هزینه‌های مسیریابی وسایل نقلیه به طور هم‌زمان کاهش یابد. فرضیات مسئله به این صورت است: تعداد وسایل نقلیه برابر با تعداد سکوهاست و ارسال مستقیم از تأمین‌کنندگان به خرده‌فروشان امکان‌پذیر نیست. این فرض بدین معنی است که یک وسیله‌ی نقلیه فقط می‌تواند به مشتریان یا تأمین‌کنندگان خدمت‌دهی کند. هر وسیله‌ی نقلیه ظرفیت مشخصی دارد و ظرفیت وسایل نقلیه مربوط به بخش جمع‌آوری (تولیدکننده‌ها) بزرگ‌تر از ظرفیت وسایل نقلیه مربوط به بخش (مشتری‌ها) است. هر سکو در هر لحظه فقط برای یکی از عملیات بارگیری (دریافت) کالا یا بارگذاری (ارسال) کالا می‌تواند استفاده شود. اما در کل سکوها می‌توانند هم عملیات بارگیری انجام دهند و هم عملیات بارگذاری. برای هر تولیدکننده و هر مشتری یک پنجره‌ی زمانی برای جمع‌آوری یا تحویل کالا مشخص شده است. در صورت دریافت کالا از تولیدکنندگان یا تحویل کالا به مشتریان در خارج از این پنجره‌ی زمانی، یک مقدار جریمه به سیستم تحمیل خواهد شد. در شکل ۱ یک شبکه‌ی حمل‌ونقل و مسیریابی و تخصیص شامل انبار عبوری، تولیدکننده‌ها و مشتریان نشان داده شده است.

هر تولیدکننده می‌تواند چند نوع کالا تولید کند و هر مشتری می‌تواند تقاضایی متشکل از چند نوع کالا داشته باشد. متعاقباً هر سکو می‌تواند تا چند نوع کالا دریافت یا ارسال کند. هزینه برای حرکت از یک گره به گره دیگر برای هر وسیله‌ی نقلیه و گره‌های مختلف بسته به مسافت متفاوت است. در ادامه علائم اصلی استفاده شده در طول این مقاله معرفی و توضیح داده می‌شوند.

### ۱.۳. مجموعه‌ها و اندیس‌ها

$V$ : مجموعه‌ی تمامی گره‌ها،  $P \subset V$  مجموعه تأمین‌کنندگان و  $RE \subset V$

مجموعه‌ی مشتریان، که در آن  $i = 1$  انبار عبوری و  $j, i$  اندیس‌های مربوط؛

می‌کند که خرده‌فروشان یا مشتری‌های از وسایل نقلیه‌ی مخصوص به آن‌ها استفاده کنند (وسایل نقلیه‌ی کوچک‌تر). رابطه‌ی ۶ تضمین می‌کند که اگر وسیله‌ی نقلیه‌ی  $k$ ام به یک گره وارد شود، باید از آن خارج شود. محدودیت‌های ۷ و ۸ مربوط به حذف زیرتور هستند. محدودیت ۹ مربوط به تخصیص تأمین‌کنندگان و مشتریان به وسایل نقلیه است. محدودیت ۱۰ تضمین می‌کند که میزان محصولاتی که به یک وسیله تخصیص می‌یابد، کمتر از یا مساوی با ظرفیت وسیله نقلیه باشد. محدودیت‌های ۱۱ و ۱۲ مربوط به تخصیص سکوها به وسایل نقلیه‌اند. محدودیت‌های ۱۳ و ۱۴ مربوط به تعادل بارهای ورودی و خروجی از سکوها هستند. محدودیت ۱۳ تضمین می‌کند که میزان بار وارد شده از نوع  $r$  به یک سکوی ورودی برابر با میزان ارسال آن نوع کالا از آن سکو به سکوهایی خروجی است. محدودیت ۱۴ بیان می‌کند که میزان بار ورودی از کالای نوع  $r$  با میزان کالای ارسالی به مشتریان از آن سکو مساوی است. میزان کالاهای ارسال شده به مشتریان از سکوها را مشخص می‌کند. محدودیت ۱۵ زمان رسیدن یک وسیله‌ی نقلیه به گره  $j$  را محاسبه می‌کند که عبارت است از زمان رسیدن به گره قبلی ( $i$ ) به‌اضافه‌ی زمان مورد نیاز برای رفتن از گره قبلی ( $i$ ) به گره مورد نظر ( $j$ ). روابط ۱۶ و ۱۷ میزان جریمه‌ی ناشی از دیرکرد یا زودکرد تحویل سفارش‌ها به هر گره را مشخص می‌کنند و رابطه‌ی ۱۸ مربوط به وضعیت متغیرهای مدل است. در مورد زمان مورد نیاز برای طی کردن کمان  $j - i$  در بخش‌های بعدی توضیحات لازم ارائه خواهد شد.

### ۵.۳. خطی‌سازی مدل

مدل ریاضی مسئله یک مدل غیرخطی است. در این زیربخش برای تسهیل در محاسبات مدل ریاضی غیرخطی ارائه شده به یک مدل ریاضی خطی تبدیل می‌شود. برای خطی‌سازی از مفهوم ریاضی مقابل استفاده می‌شود. فرض کنید متغیرهای یک مسئله  $z$  و  $y$  باشند و در مدل مسئله از حاصل‌ضرب این دو متغیر  $(z, y)$  استفاده شود. اگر یک متغیر جدید به نام  $x$  تعریف کنیم که رابطه‌ی بین  $x$  و  $y$  و  $z$  به صورت مقابل باشد:  $x \leq z$  و  $x \leq y + z - 1$ ، در این صورت با جایگزینی حاصل‌ضرب این دو متغیر با متغیر جدید می‌توان مسئله را به یک مسئله‌ی خطی تبدیل کرد. در مدل ارائه شده روابط ۱۳ و ۱۴ و همچنین رابطه‌ی ۱۷ باعث غیرخطی شدن مدل مسئله شده‌اند. با تعریف  $ZY_{iq}^k = Z_i^k Y_q^k$  جایگزین کردن متغیر جدید به جای ضرب دو متغیر قبلی و همچنین اضافه کردن سه محدودیت زیر به مدل، مدل خطی می‌شود.

$$ZY_{iq}^k \leq Z_i^k \quad (19)$$

$$ZY_{iq}^k \leq Y_q^k \quad (20)$$

$$ZY_{iq}^k \geq Z_i^k + Y_q^k - 1 \quad (21)$$

برای خطی‌سازی معادله‌ی شماره‌ی ۱۷ نیز مانند قبل عمل می‌کنیم و روابط زیر را در مدل جایگزین می‌کنیم.

$$T_{ij}^k = AT_i^* x_{ij}^k \quad (22)$$

$$T_{ij}^k \leq AT_i \quad (23)$$

$$T_{ij}^k \leq M^* x_{ij}^k \quad (24)$$

$$T_{ij}^k \geq AT_i - M^*(1 - x_{ij}^k) \quad (25)$$

$$T_{ij}^k \geq 0 \quad (26)$$

Minimize

$$\sum_{i \in V} \sum_{j \in V} \sum_{k \in K} C_{ij}^k X_{ij}^k + \sum_{l \in L} \sum_{q \in L} \sum_{r \in R} D_{lq} C D_r N_{lqr} + \sum_{j \in v} (Pl_j.Vl_j + Pu_j.Vu_j) \quad (1)$$

St :

$$\sum_{j \in V} X_{ij}^k = 1 \quad k \in K \quad (2)$$

$$\sum_{i \in V} \sum_{k \in K} X_{ij}^k = 1 \quad j \in V/\Lambda \quad (3)$$

$$\sum_{i \in RE} \sum_{j \in V} \sum_{k \in K_1} X_{ij}^k = 0 \quad (4)$$

$$\sum_{i \in P} \sum_{j \in V} \sum_{k \in K_2} X_{ij}^k = 0 \quad (5)$$

$$\sum_{i \in V} X_{ij}^k - \sum_{i \in V} X_{ji}^k = 0 \quad k \in K, j \in V \quad (6)$$

$$\sum_{k \in K} U_i^k = 0 \quad (7)$$

$$U_i^k + 1 \leq U_j^k + M(1 - X_{ij}^k) \quad (8)$$

$$i \in V, j \in V/\Lambda, k \in K \quad (8)$$

$$\sum_{i \in V} X_{ij}^k = Z_j^k \quad j \in V/\Lambda, k \in K \quad (9)$$

$$\sum_{i \in V} \sum_{r \in R} dp_{ir} Z_i^k \leq CP_k \quad k \in K \quad (10)$$

$$\sum_{q \in L} Y_q^k = 1 \quad k \in K \quad (11)$$

$$\sum_{k \in K} Y_q^k = 1 \quad q \in L \quad (12)$$

$$\sum_{k \in K} \sum_{i \in P} dp_{ir} Z_i^k Y_q^k = \sum_{l \in L} N_{qlr} \quad (13)$$

$$q \in L, r \in R$$

$$\sum_{k \in K} \sum_{i \in RE} dp_{ir} Z_i^k Y_q^k = \sum_{l \in L} N_{lqr} \quad (14)$$

$$q \in L, r \in R$$

$$AT_j = (AT_i + t_{ij}) * \sum_{k \in K} x_{ij}^k \quad \forall i, j \in v/\Lambda \quad (15)$$

$$AT_j < ub_j + Vu_j \quad \forall j \in v \quad (16)$$

$$AT_j \geq lb_j - Vl_j \quad \forall j \in v \quad (17)$$

$$X_{ij}^k, Z_i^k, Y_q^k \in \{0, 1\}, \quad (18)$$

$$N_{lqr}, U_i^k \in R^+ \quad (18)$$

رابطه‌ی ۱ تابع هدف مسئله است، که مجموع کل هزینه‌ها شامل هزینه‌های حمل‌ونقل بین گره‌ها و هزینه‌های حمل‌ونقل داخل انبار عبوری است. محدودیت ۲ بیان می‌کند که تمام وسایل نقلیه از انبار عبوری خارج شوند. رابطه‌ی ۳ تضمین می‌کند که تمام نقاط بازدید شوند. رابطه‌ی ۴ تضمین می‌کند که تولیدکنندگان از وسایل نقلیه‌ی که به آن‌ها تخصیص داده شده است، استفاده کنند (وسایل نقلیه‌ی بزرگ‌تر) و ۵ تضمین

۰/۴۷	۰/۳۵	۰/۸۳	۰/۵۹	۰/۵۵	۰/۹۲	۰/۲۹	۰/۷۶	۰/۷۵	۰/۳۸	۰/۵۷	۰/۰۸
۲	۱		۳	۲		۱	۳	۴	۲	۳	۱
تامین کنندگان				مشتریان				سکوها			

شکل ۲. یک مثال عددی از نمایش جواب.

#### ۴. رویکرد حل مسئله

در این بخش در ابتدا رشته‌ی پیشنهادی استفاده شده در روش‌های فرابابتکاری توضیح داده می‌شود؛ در ادامه به توضیح الگوریتم فرابابتکاری پیشنهادی مبتنی بر الگوریتم شبیه‌سازی تبرید (SA) و الگوریتم فرابابتکاری جست‌وجوی ممنوعه (TS) پرداخته می‌شود و در نهایت نحوه‌ی تنظیم پارامترهای الگوریتم توضیح داده می‌شود.

##### ۱.۴. نمایش جواب

اولین گام در حل یک مسئله با استفاده از روش‌های فرابابتکاری، تعریف یک نمایش جواب است. برای تولید جمعیت اولیه ساختار نمایش جواب باید متناسب با ویژگی‌های مسئله و جواب‌های تولید شده باشد. برای این منظور در این مقاله از یک رشته جواب با طول  $(L) + (RE + K_2 - 1) + (P + K_1 - 1)$  استفاده شده است. مقادیر موجود در رشته جواب اعدادی بین صفر و یک هستند. تعداد  $P + K_1 - 1$  خانه‌ی اول مربوط به مسیریابی جمع‌آوری کالاهاست.  $RE + K_2 - 1$  خانه‌ی بعدی مربوط به مسیریابی توزیع است و  $L$  خانه‌ی نهایی مربوط به تخصیص وسایل نقلیه به سکوهاست. شکل ۲ نمونه‌ی از رشته‌ی تعریف شده برای مسئله‌ی با ۳ تولیدکننده و ۳ مشتری و ۲ وسیله‌ی نقلیه نوع ۱ و ۲ وسیله‌ی نقلیه نوع ۲ را نشان می‌دهد. چهار خانه‌ی اول مربوط به مسیریابی بخش جمع‌آوری از تأمین‌کنندگان است. برای تفسیر این بخش از نمایش جواب اعداد داخل خانه‌ها به ترتیب صعودی مرتب می‌شوند و مقادیر رتبه‌ی آنها به خانه‌ی مربوط تخصیص می‌یابد.  $P$  شماره‌ی اول نشان‌دهنده توالی مشتریان است و  $K_1 - 1$  شماره‌ی بعدی نشان‌دهنده نقاط تفکیک وسایل نقلیه است. در مثال ارائه شده مشتریان ۱ و ۲ با وسیله‌ی نقلیه‌ی اول خدمت‌رسانی می‌شوند و مشتری دوم ۳ با وسیله‌ی نقلیه‌ی دوم خدمت‌رسانی می‌شود. تفسیر بخش دوم رشته نیز مانند بخش اول است. اعداد رتبه‌ی حاصل در بخش آخر رشته نشان‌دهنده‌ی سکوی تخصیص‌یافته به هر وسیله‌ی نقلیه هستند.

##### ۲.۴. الگوریتم جست‌وجوی ممنوعه

الگوریتم جست‌وجوی ممنوعه در سال ۱۹۸۹ برای اولین بار توسط فرد گلاور معرفی شد. در میان الگوریتم‌های فرابابتکاری که برای حل مسائل مسیریابی وسایل نقلیه استفاده شده‌اند، این الگوریتم کارایی خوبی داشته است.<sup>[۲۱]</sup> بعد از معرفی این الگوریتم روش‌های متنوعی برای بهبود آن معرفی شده و ساختار این الگوریتم در حالت کلی به این صورت است که ابتدا یک جواب اولیه تولید می‌شود، سپس به تولید همسایگی در اطراف جواب اولیه می‌پردازد. در صورتی که تغییری در رشته سبب بهبود در جواب شود، این تغییر پذیرفته می‌شود و تغییر یاد شده در یک فهرست ممنوعه قرار می‌گیرد تا در چندین گام از قبل مشخص

شده این تغییر در رشته باقی بماند. از این فهرست برای جلوگیری از بازگشت به جواب‌هایی که قبلاً بررسی شده‌اند و باعث ایجاد بهبود در جواب مسئله شده‌اند، استفاده می‌شود که منجر به افزایش کارایی الگوریتم و در عین حال کاهش زمان اجرای آن می‌شود. در الگوریتم جست‌وجوی ممنوعه استفاده شده در این مقاله ابتدا یک فهرست از جواب‌های قابل قبول (ActionList) تولید می‌شود. با آزمایش‌های اولیه، طول فهرست مطرح در این مقاله وابسته به طول نمایش جواب تعریف شد. در تولید (ActionList) از سه عملگر جهش استفاده شده است که شرح این عملگرها در ادامه ارائه می‌شود. با فرض طول رشته برابر با  $RL$  تعداد  $RL(RL - 1)/2$  همسایگی با استفاده از جهش Scramble، تعداد  $RL(RL - 1)/2$  همسایگی با استفاده از جهش Reversion، و تعداد  $2RL$  همسایگی با استفاده از جهش Insertion تولید شد. پس از مشخص شدن تعداد جواب‌های قابل قبول (ActionList)، طول فهرست ممنوعه (TobuList) مشخص می‌شود. نتایج پیش‌آزمایش‌ها نشان دادند که یک فهرست ممنوعه با طولی برابر با  $1/4$  طول فهرست جواب‌های قابل قبول برای بررسی، که در قبل محاسبه شد، عملکرد خوبی ارائه می‌کند.

##### ۳.۴. الگوریتم شبیه‌سازی تبرید

الگوریتم شبیه‌سازی تبرید برای اولین بار در سال ۱۹۸۳ توسط کرکپاتریک و همکاران<sup>[۲۲]</sup> معرفی شد. این الگوریتم برای حل تعداد زیادی از مسائل پیچیده و مسائل مربوط به دنیای واقعی به کار گرفته شده است.<sup>[۲۳]</sup> ویژگی اصلی این الگوریتم فرابابتکاری برخلاف سایر الگوریتم‌های فرابابتکاری مبتنی بر جست‌وجوی همسایگی، این است که این الگوریتم در همسایگی جواب فعلی به دنبال بهینه‌ی کلی نمی‌گردد؛ بلکه SA یک راه حل تصادفی از همسایگی جواب فعلی انتخاب می‌شود و بر اساس تابع برازندگی یا تابع هزینه‌ی آن این راه حل جدید پذیرفته یا رد می‌شود. این الگوریتم شرایط لازم برای خارج شدن از بهینه‌ی محلی را نیز فراهم می‌کند که بر کارایی الگوریتم می‌افزاید.<sup>[۲۵]</sup> احتمال پذیرش جواب‌های نامطلوب به دمای فعلی بستگی دارد. الگوریتم SA به شرطی با احتمال ۱ به سمت جواب بهینه حرکت می‌کند که دمای اولیه بسیار زیاد و سرعت کاهش دما کم باشد؛ در غیر این صورت الگوریتم به سمت یک بهینه‌ی محلی همگرا خواهد شد که این بهینه‌ی محلی می‌تواند بهینه‌ی جهانی باشد. در این الگوریتم از مفهوم دما برای شبیه‌سازی فرایند سرد شدن فیزیکی استفاده می‌شود و دمای بالاتر به معنی احتمال بیشتر برای پذیرش جواب جدید است. در این مقاله در طراحی ساختار الگوریتم SA برای تولید جواب‌های همسایه از سه نوع جهش مشابه الگوریتم جست‌وجوی ممنوعه استفاده شد که شرح آن در ادامه خواهد آمد. شبه‌کد الگوریتم شبیه‌سازی تبرید به صورت شکل ۳ ارائه می‌شود.

##### ۴.۴. عملگرهای تولید همسایگی

در این مقاله از سه نوع مختلف تولید همسایگی در الگوریتم‌های مطرح بهره برده شد. جهش تکان (Scramble)، جهش الحاق (Insertion) و جهش معکوس‌سازی (Reversion) جهش‌های استفاده شده در این پژوهش‌اند که در پیش‌آزمایش‌ها نتایج خوبی ارائه کرده‌اند. در جهش تکان، دو نقطه به تصادف از یک رشته جواب انتخاب می‌شوند و عناصر بین آن دو نقطه به طور تصادفی در هم می‌آمیزند (شکل ۴). در جهش الحاق، یک عنصر به تصادف انتخاب می‌شود و در طول رشته‌ی جواب در محل دیگری قرار می‌گیرد (شکل ۵) و در نهایت در جهش معکوس‌سازی، دو نقطه

تولید یک جواب اولیه قابل قبول (x)  
 محاسبه تابع هزینه جواب فعلی (xCost)  
 برای it از ۱ تا بیشینه تعداد تکرارهای اول (MaxIt)  
 برای i از ۱ تا بیشینه تعداد تکرارهای دوم (MaxIt2)  
 تولید تعداد ۴/ ActionallList جواب تصادفی جدید (xnew) در همسایگی جواب فعلی  
 محاسبه تابع هزینه جواب تولید شده و انتخاب بهترین همسایگی تولید شده به عنوان جواب انتخابی (xnew)  
 $\theta = xnewCost - xCost$   
 اگر  $(xnewCost) \leq (xCost)$   
 جواب تولید شده (xnew) جایگزین جواب فعلی می شود  
 در غیر این صورت  
 تولید یک عدد تصادفی rand  
 اگر  $rand \leq e^{-(\theta/\Delta)}$   
 جایگزینی xnew با جواب فعلی  
 پایان حلقه i  
 به روز رسانی  $f_{ii} = \alpha \cdot f_{ii-1}$   
 پایان حلقه it  
 گزارش بهترین جواب یافته شده

شکل ۳. شبکه کد الگوریتم شبیه‌سازی تبرید.

#### ۵.۴. طراحی آزمایش‌ها

در الگوریتم‌های طراحی شده سطوح مختلف پارامترهای هر الگوریتم در بیش‌از‌پیش آزمایش‌ها بررسی شده‌اند و بهترین سطح برای انجام آزمایش‌ها در نظر گرفته شده است. جدول ۱ سطوح منتخب برای الگوریتم جست‌وجوی ممنوعه (TS) و الگوریتم شبیه‌سازی تبرید (SA) را نشان می‌دهد.

در این قسمت به بررسی عملکرد الگوریتم‌های فراابتکاری در ابعاد کوچک و بزرگ پرداخته می‌شود. بدین منظور در ابتدا نحوه تولید مسائل نمونه توضیح داده می‌شود و در ادامه به بررسی نتایج حاصل از الگوریتم‌ها در ابعاد کوچک و بزرگ پرداخته می‌شود. به منظور بررسی و مقایسه‌ی کارایی دو الگوریتم پیشنهاد شده در این مقاله، هر دو الگوریتم در محیط ۲۰۱۴ MATLAB کدنویسی شده‌اند. این الگوریتم‌ها توسط رایانه‌ی با مشخصات Intel(R) Core(TM) i5، ۴ GB of RAM، CPU ۳/۲ GHz مدل مسئله‌ی مطرح شده در نرم‌افزار GAMS ۲۳/۷ نیز کدنویسی شده است و نتایج با هم مقایسه شده‌اند. برای حل مسائل نمونه در نرم‌افزار GAMS یک محدودیت زمانی ۳۶۰۰ ثانیه‌ی در نظر گرفته شده است.

#### ۶.۴. تولید مسائل نمونه

به دلیل جدید بودن مسئله‌ی مطرح در این مقاله هیچ دسته‌ی مسائل نمونه‌ی برای مقایسه وجود ندارد. پس تعدادی مسئله‌ی نمونه طراحی شده است تا با استفاده از آن‌ها بتوان عملکرد الگوریتم‌ها را مقایسه کرد.

مسائل نمونه‌ی استفاده شده در این مقاله با تغییراتی به منظور نزدیک‌سازی مسائل نمونه به مسئله‌ی مطرح از مسائل نمونه‌ی آگرات و همکاران موجود در وب‌سایت <http://neo.lcc.uma.es/vrp/> به دست آمده‌اند. در این دسته از مسائل نمونه، ۲۷ مسئله‌ی نمونه برای مسئله‌ی مسیریابی با ظرفیت محدود ارائه شده است. برای مشخص کردن تعداد گره‌های مربوط به تولیدکننده‌ها و مشتریان



شکل ۴. جهش تک‌کان.



شکل ۵. جهش الحاق.



شکل ۶. جهش معکوس‌سازی.

به تصادف انتخاب می‌شوند و عناصر بین این دو نقطه برعکس می‌شوند (شکل ۶). بعد از اجرای عملگر جهش برای تبدیل رشته جواب‌ها به جواب‌های متناسب با ساختار مسئله مراحل قبلی تکرار می‌شوند. معیار پایان الگوریتم‌ها در این مقاله عدم بهبود جواب در ۵۰ تکرار متوالی در نظر گرفته شده است.

جدول ۱. تنظیم پارامتر الگوریتم جست و جوی ممنوعه و الگوریتم شبیه‌سازی تبرید.

نوع پارامتر	سطح پارامتر الگوریتم جست و جوی ممنوعه	سطح پارامتر الگوریتم شبیه‌سازی تبرید
تعداد تکرار تا توقف الگوریتم بدون بهبود جواب (MaxIt)	۵۰	۵۰
تولید همسایگی از جهش تک‌ان در هر تکرار	$RL(RL - 1)/2$	$RL(RL - 1)/8$
تولید همسایگی از جهش الحاق در هر تکرار	$RL(RL - 1)/2$	$RL(RL - 1)/8$
تولید همسایگی از جهش معکوس‌سازی در هر تکرار	۲.RL	RL/۲
طول فهرست ممنوعه (TabuList)	۱/۴.(ActionList)	-
دمای اولیه ( $t_0$ )	-	۱۰ جواب اولیه تولید شده
بیشینه‌ی تعداد تکرار دوم (MaxIt۲)	-	۱۰
مقدار پارامتر $\alpha$	-	۰٫۹۹

جدول ۲. نتایج حاصل از نرم‌افزار گمز، الگوریتم‌های شبیه‌سازی تبرید و جست و جوی ممنوعه برای مسائل در ابعاد کوچک.

#	P	RE	K	CPLEX		SA			TS		
				Opt Sol	Time	Opt Sol	Time	Gap%	Opt Sol	Time	Gap%
۱	۲	۲	۲	۶۹۰۵	۰٫۰۳	۶۹۰۵	۵٫۹۸	۰٫۰۰۰۰	۶۹۰۵	۰٫۰۵	۰٫۰۰۰۰
۲	۲	۴	۲	۷۴۷۵	۰٫۵	۷۴۷۵	۳۰٫۳۴	۰٫۰۰۰۰	۷۴۷۵	۰٫۱۲	۰٫۰۰۰۰
۳	۲	۶	۳	۸۶۶۶	۰٫۱۴	۹۴۸۰	۳۷٫۱۹	۰٫۰۸۵۹	۸۶۶۶	۰٫۴۸	۰٫۰۰۰۰
۴	۳	۶	۳	۹۱۱۶	۰٫۱۴	۹۹۶۲	۳۶٫۹۲	۰٫۰۸۴۹	۹۹۵۳	۰٫۵۰	۰٫۰۸۴۱
۵	۴	۶	۳	۱۰۸۶۴	۰٫۰۹	۱۰۸۶۴	۳۷٫۲۶	۰٫۰۰۰۰	۱۰۹۱۴	۰٫۶۵	۰٫۰۰۴۶
۶	۳	۸	۳	۹۵۸۳	۰٫۱۴	۱۰۴۵۰	۳۹٫۰۲	۰٫۰۸۳۰	۱۰۴۶۰	۰٫۹۲	۰٫۰۸۳۸
۷	۳	۸	۴	۱۱۴۹۴	۸٫۸۴	۱۱۴۹۴	۴۱٫۰۸	۰٫۰۰۰۰	۱۱۵۵۷	۱٫۵۱	۰٫۰۰۵۵
۸	۵	۸	۴	۱۳۹۰۹	۷٫۱۷	۱۴۱۴۷	۴۱٫۴۱	۰٫۰۱۶۸	۱۴۰۷۱	۲٫۷۲	۰٫۰۱۱۵
۹	۶	۸	۴	۱۳۵۵۳	۱۴٫۷۵	۱۳۵۵۳	۴۱٫۴۷	۰٫۰۰۰۰	۱۳۵۵۳	۲٫۸۳	۰٫۰۰۰۰
۱۰	۶	۱۰	۴	۱۵۱۵۰	۳۵٫۴۷	۱۵۲۳۲	۴۰٫۴۷	۰٫۰۰۵۴	۱۵۲۰۵	۴٫۷۱	۰٫۰۰۳۶
۱۱	۶	۱۰	۵	۱۵۹۹۵	۱۰۹۵٫۰۶	۱۶۳۸۷	۴۷٫۳۵	۰٫۰۲۳۹	۱۶۵۰۶	۶٫۴۲	۰٫۰۳۱۰
۱۲	۶	۱۲	۵	۱۶۱۵۱	۲۰۴٫۶۱	۱۶۱۶۶	۴۲٫۱۴	۰٫۰۰۰۹	۱۶۱۵۱	۸٫۹۶	۰٫۰۰۰۰
۱۳	۸	۱۲	۵	۱۶۷۴۰	۹۹۱٫۰۲	۱۶۸۱۵	۴۲٫۳۱	۰٫۰۰۴۵	۱۶۸۳۱	۱۳٫۳۵	۰٫۰۰۵۴
۱۴	۸	۱۳	۵	۷۱۰۲	۰٫۰۳	۷۱۰۲	۵٫۹۹	۰٫۰۰۰۰	۷۱۰۲	۰٫۰۷	۰٫۰۰۰۰
۱۵	۸	۱۴	۵	۷۶۷۵	۰٫۰۵	۷۶۷۵	۳۰٫۸۴	۰٫۰۰۰۰	۷۶۷۵	۰٫۱۳	۰٫۰۰۰۰
۱۶	۸	۱۵	۵	۸۸۳۲	۰٫۱۴	۹۴۲۳	۳۷٫۱۸	۰٫۰۶۲۷	۹۵۱۸	۰٫۴۴	۰٫۰۷۲۱
میانگین								۰٫۰۲۳۰	۰٫۰۱۸۸		

به این صورت عمل شده است: دو سوم تعداد کل گره‌ها (بدون در نظر گرفتن انبار عبوری) به طرف بالا گرد شده و به عنوان تعداد گره‌های مربوط به مشتری‌ها در نظر گرفته شدند و باقی‌گه‌ها به تولیدکننده‌ها تخصیص داده شدند. برای محاسبه ظرفیت وسایل نقلیه بدین صورت عمل شد که در هر مسئله ظرفیت وسایل نقلیه مسئله‌ی نمونه برای وسایل نقلیه‌ی کوچک در نظر گرفته شد و برای وسایل نقلیه بزرگ ظرفیت اولیه در عدد ۱/۵ ضرب شد. تعداد وسایل نقلیه به صورتی تعیین شد که ظرفیت وسایل نقلیه در حدود ۱/۲ تقاضا باشد. برای تمام مسائل نمونه دو نوع کالا در نظر گرفته شد. هزینه انتقال هر واحد کالا در داخل انبار عبوری برای کالای نوع ۱ برابر با ۱ و برای کالای نوع ۲ برابر با ۲ در نظر گرفته شد. فاصله‌ی بین سکوها در داخل انبار عبوری به این صورت محاسبه شد که سکوها روبه‌رو یا مجاور هم فاصله‌ی برابر با ۱۰ و در غیر این صورت فاصله‌ی سکوها برابر با ۲۰ در نظر گرفته شدند. میزان تقاضا یا تولید از هر کالا به صورت تصادفی تولید شده است. میزان تقاضا به صورت تصادفی در بازه‌ی ۱ تا ۲۰ و میزان تولید هر نوع کالا در بازه‌ی ۱ تا ۴۰ تولید شده است. در نهایت به منظور موازنه‌ی میزان تولید و تقاضا تغییراتی بر میزان تولید اعمال شد. مسائل کوچک و متوسط نیز از مسائل نمونه‌ی بزرگ و با حذف تعدادی از مشتریان و تأمین‌کنندگان حاصل شده‌اند. به منظور تولید پنجره‌ی زمانی مشتریان حداکثر طول تورهای بهینه‌ی مسئله‌ی اصلی در عدد ۲ ضرب شد و به عنوان بازه‌ی زمانی در دسترس در نظر گرفته شد. حال به ازای هر گره در بازه‌ی زمانی محاسبه شده عددی تصادفی تولید شد. این عدد به عنوان حد پایین پنجره‌ی زمانی گره در نظر گرفته شد و طول پنجره‌ی زمانی مشتری با تولید عددی تصادفی در بازه‌ی زمانی در دسترس تقسیم بر ۱۰ به دست آمد.

جدول ۳. نتایج حاصل از الگوریتم‌های شبیه‌سازی تبرید و جست‌وجوی ممنوعه برای مسائل در ابعاد بزرگ.

#	P	RE	K	SA			TS		
				Opt Sol	CPU Time	Gap%	Opt Sol	CPU Time	Gap%
۱	۱۰	۲۱	۵	۲۰۹۲۷	۴۲٫۶۳	۰	۲۱۰۷۵۰	۴۲٫۴۹	۰٫۰۰۷۰
۲	۱۰	۲۲	۵	۱۹۲۱۹	۴۲٫۸۲	۰	۱۹۷۵۱	۴۶٫۴۸	۰٫۲۶۹
۳	۱۰	۲۲	۶	۱۷۷۳۶	۴۶٫۲۰	۰	۱۸۰۹۹	۷۱٫۶۵	۰٫۰۲۰۰
۴	۱۱	۲۲	۵	۳۰۴۸۷	۴۲٫۶۶	۰	۳۰۵۵۸	۴۷٫۸۷	۰٫۰۰۲۳
۵	۱۱	۲۴	۵	۲۰۳۸۷	۴۳٫۶۳	۰	۲۱۱۶۱	۴۵٫۴۴	۰٫۰۳۶۶
۶	۱۲	۲۴	۵	۱۹۴۷۰	۴۶٫۳۹	۰	۱۹۵۱۰	۵۳٫۵۴	۰٫۰۰۲۰
۷	۱۲	۲۴	۶	۲۳۰۹۱	۵۰٫۱۰	۰	۲۳۴۷۱	۷۳٫۳۴	۰٫۰۱۶۲
۸	۱۲	۲۵	۵	۲۰۲۹۹	۴۱٫۹۷	۰	۲۰۶۸۳	۸۱٫۷۸	۰٫۰۱۸۶
۹	۱۲	۲۶	۵	۲۱۶۱۴	۴۵٫۵۷	۰	۲۲۱۱۶	۷۴٫۱۴	۰٫۰۲۲۷
۱۰	۱۲	۲۶	۶	۲۱۳۹۱	۴۵٫۴۶	۰	۲۲۱۵۲	۱۰۱٫۷۶	۰٫۰۳۴۴
۱۱	۱۴	۲۹	۷	۲۵۲۷۲	۵۴٫۰۱	۰	۲۵۸۶۷	۱۵۷٫۰۰	۰٫۰۲۳۰
۱۲	۱۴	۳۰	۶	۲۵۶۸۱	۵۱٫۴۴	۰	۲۶۲۳۳	۱۲۵٫۴۱	۰٫۰۲۱۰
۱۳	۱۴	۳۰	۷	۲۸۰۰۲	۵۴٫۸۴	۰	۲۹۱۹۰	۱۵۲٫۷۷	۰٫۰۴۰۷
۱۴	۱۵	۳۰	۷	۲۴۴۲۰	۵۴٫۵۰	۰	۲۴۴۲۰	۱۷۴٫۴۴	۰٫۰۰۸۱
۱۵	۱۵	۳۲	۷	۲۷۲۵۲	۵۵٫۳۴	۰	۲۹۶۹۶	۱۶۳٫۳۰	۰٫۰۸۲۳
۱۶	۱۷	۳۵	۷	۲۶۲۶۱	۵۴٫۷۰	۰	۲۸۳۹۱	۲۲۸٫۳۲	۰٫۰۷۵۰
۱۷	۱۷	۳۶	۷	۲۹۱۹۲	۵۳٫۷۸	۰	۳۰۷۴۵	۲۵۱٫۷۲	۰٫۰۵۰۶
۱۸	۱۸	۳۶	۹	۳۰۹۰۸	۶۱٫۶۸	۰	۳۲۸۴۶	۳۱۲٫۸۳	۰٫۰۵۹۰
۱۹	۱۹	۴۰	۹	۳۲۶۳۷	۶۱٫۷۳	۰	۳۵۱۶۵	۴۰۹٫۱۰	۰٫۰۷۱۹
۲۰	۲۰	۴۰	۹	۳۲۰۸۹	۶۲٫۰۱	۰	۳۴۱۸۶	۴۸۲٫۸۱	۰٫۰۶۱۳
۲۱	۲۰	۴۱	۸	۳۱۵۱۱	۵۸٫۲۹	۰	۳۲۵۶۸	۴۲۲٫۶۰	۰٫۰۳۲۵
۲۲	۲۰	۴۲	۹	۳۵۳۴۲	۵۸٫۰۵	۰	۳۶۹۱۸	۵۰۶٫۱۳	۰٫۰۴۲۷
۲۳	۲۰	۴۲	۱۰	۳۴۴۲۸	۶۳٫۹۷	۰	۳۶۰۳۵	۴۹۵٫۱۸	۰٫۰۴۴۶
۲۴	۲۱	۴۲	۹	۳۵۵۳۷	۵۶٫۰۴	۰	۳۷۳۴۹	۵۵۳٫۲۰	۰٫۰۴۸۵
۲۵	۲۱	۴۳	۹	۳۱۸۱۳	۵۹٫۶۰	۰	۳۳۸۷۳	۴۶۹٫۲۴	۰٫۰۶۰۸
۲۶	۲۲	۴۶	۹	۳۴۶۳۹	۶۱٫۵۷	۰	۳۶۸۰۸	۷۳۷٫۶۶	۰٫۰۵۸۹
۲۷	۲۶	۵۳	۱۰	۴۵۱۶۶	۶۷٫۹۰	۰	۴۶۷۳۱	۱۰۱۴٫۹۴	۰٫۰۳۳۵
میانگین					۵۳٫۲۱	۰		۲۷۰٫۱۹	۰٫۰۳۷۱

در این معادله  $Obj$  مقدار تابع هدف حاصل از الگوریتم فراابتکاری و  $BObj$  بهترین جواب پیدا شده برای مسئله است.

همان‌گونه که از جدول ۲ مربوط به حل مسائل در ابعاد کوچک مشخص است، هر دو الگوریتم با خطای در حدود ۲٪ عملکرد مناسبی داشته‌اند؛ اما از نظر زمان حل الگوریتم جست‌وجوی ممنوعه عملکرد بهتری را ارائه کرده است. بیشینه‌ی خطا در دو الگوریتم یکسان است که برابر با ۸٪ است. هر دو الگوریتم جست‌وجوی ممنوعه و شبیه‌سازی تبرید در تعداد ۷ مسئله از ۱۶ مسئله‌ی نمونه به جواب بهینه رسیده‌اند.

### ۲.۵. نتایج در ابعاد بزرگ

در این قسمت به بررسی عملکرد الگوریتم‌های جست‌وجوی ممنوعه و شبیه‌سازی تبرید در حل مسئله‌ی مطرح شده پرداخته می‌شود. همان‌گونه که در جدول ۳ مشخص است، الگوریتم شبیه‌سازی تبرید پیشنهادی در تمام مسائل با ابعاد بزرگ جوابی بهتر

## ۵. نتایج

در این بخش نتایج حل مسائل در ابعاد کوچک و بزرگ بررسی می‌شود.

### ۱.۵. نتایج در ابعاد کوچک

نتایج به دست آمده از حل مسائل نمونه با نرم‌افزار GAMS و دو الگوریتم طراحی شده در جداول ارائه شده‌اند. همان‌طور که در جداول مشخص است، نرم‌افزار GAMS قادر به حل مسائل با تعداد گره‌های بیشتر از ۲۰ و تعداد وسایل نقلیه‌ی بیشتر از ۵ نیست. روش دقیق در مسئله‌ی ۱۴ در زمان محدود ۳۶۰۰ ثانیه به جواب موجه نرسید؛ پس مسئله‌ی ۱۴ و بقیه‌ی مسائل بزرگ‌تر به عنوان مسائل متوسط و بزرگ در نظر گرفته شد. به منظور محاسبه‌ی مقدار خطا از رابطه‌ی ۲۷ بهره گرفته شد.

$$Gap = \frac{Obj - BObj}{BObj} \quad (27)$$



از سه نوع همسایگی مختلف بهره برده شد. به منظور بررسی عملکرد الگوریتم پیشنهادی ۱۶ مسئله در ابعاد کوچک و ۲۷ مسئله در ابعاد بزرگ حل شد و نتایج بررسی شد. از نظر عملکرد هر دو الگوریتم فرابابتکاری در ابعاد کوچک عملکرد نزدیک به هم داشتند. اما نتایج حاصل از حل مسائل با ابعاد بزرگ نشان دهنده برتری الگوریتم شبیه‌سازی تیرید پیشنهادی نسبت به الگوریتم جست‌وجوی ممنوعه است. در این ابعاد الگوریتم شبیه‌سازی تیرید جواب‌های حاصل از الگوریتم جست‌وجوی ممنوعه را در حدود سه درصد بهبود داد. از نظر زمان حل نیز الگوریتم شبیه‌سازی تیرید در زمانی در حدود ۵۳ ثانیه به طور متوسط هر مسئله را حل کرد. در حالی که الگوریتم جست‌وجوی ممنوعه به طور متوسط زمانی در حدود ۲۷۰ ثانیه برای حل مسائل صرف کرد. به منظور انجام تحقیقات و فعالیت‌های آتی می‌توان عملکرد سایر الگوریتم‌ها اعم از الگوریتم‌های دقیق مانند روش شاخه و کران و الگوریتم‌های فرابابتکاری دیگر را بررسی کرد. همچنین می‌توان مبحث زمان‌بندی را با توجه به اهمیت و کاربردی بودن آن در مسئله‌ی ارائه شده به طور هم‌زمان در یک مسئله بررسی کرد. از سوی دیگر مبحث هماهنگی زمانی وسایل نقلیه‌ی ورودی و خروجی در یک انبار عبوری مسئله‌ی پر اهمیتی است که می‌تواند در مدل مطرح شده بررسی شود.

از الگوریتم جست‌وجوی ممنوعه ارائه می‌کند. در عین حال زمان مورد نیاز الگوریتم شبیه‌سازی تیرید بسیار کمتر از زمان مورد نیاز الگوریتم جست‌وجوی ممنوعه است.

## ۶. نتیجه‌گیری

در این مقاله به منظور کمیته‌سازی هزینه‌های حمل‌ونقل داخلی و خارجی انبارهای عبوری یک مدل ریاضی جدید یکپارچه برای مسئله‌ی تخصیص وسایل نقلیه به سکوها و مسیریابی وسایل نقلیه ارائه شد. حل مسئله‌ی تخصیص سکوها در یک انبار عبوری یا حل مسئله‌ی مسیریابی در یک انبار عبوری جواب‌های غیرکارا برای مسئله ایجاد می‌کنند؛ زیرا این دو مسئله بر هم تأثیر می‌گذارند. در نظر گرفتن هم‌زمان دو مسئله‌ی مسیریابی در انبار عبوری و تخصیص وسایل نقلیه به سکوها می‌تواند جواب‌های کاراتری را برای مدیریت به همراه داشته باشد. با توجه به NP-Hard بودن مسئله‌ی مطرح شده، دو الگوریتم فرابابتکاری جست‌وجوی ممنوعه و شبیه‌سازی تیرید برای مسئله پیشنهاد شدند. در الگوریتم‌های ارائه شده ساختار رشته‌جواب متناسب با ساختار مسئله طراحی شده است. همچنین در هر دو الگوریتم، برای ایجاد همسایگی

## منابع (References)

- Kuo, Y. "Optimizing truck sequencing and truck dock assignment in a cross docking system", *Expert Systems with Applications*, **40**, pp. 5532-5541 (2013).
- Van Belle, J., Valckenaers, P. and Cattrysse, D. "Cross-docking: state of the art", *Omega*, **40**, pp. 827-846 (2012).
- Oh, Y., Hwang, H., Cha, C.N. and et al. "A dock-door assignment problem for the Korean mail distribution center", *Computers & Industrial Engineering*, **51**, pp. 288-296 (2006).
- Miao, Z., Lim, A. and Ma, H. "Truck dock assignment problem with operational time constraint within cross-docks", *European Journal of Operational Research*, **192**, pp. 105-115 (2009).
- Konur, D. and Golias, M.M. "Cost-stable truck scheduling at a cross-dock facility with unknown truck arrivals: a meta-heuristic approach", *Transportation Research Part E: Logistics and Transportation Review*, **49**, pp. 71-91 (2013).
- Miao, Z., Cai, S. and Xu, D. "Applying an adaptive tabu search algorithm to optimize truck-dock assignment in the crossdock management system", *Expert Systems with Applications*, **41**, pp. 16-22 (2014).
- Madani-Isfahani, M., Tavakkoli-Moghaddam, R. and Naderi, B. "Multiple cross-docks scheduling using two meta-heuristic algorithms", *Computers & Industrial Engineering*, **74**, pp. 129-138 (2014).
- Kellar Gregory, M., George, G. and Polak, Zhang X. "Synchronization, cross-docking, and decoupling in supply chain networks", *International Journal of Production Research*, **54.9**, pp. 2585-2599 (2016).
- Nikolopoulou, Amalia, I. and et al. "Moving products between location pairs: Cross-docking versus direct-shipment", *European Journal of Operational Research*, **256**(3), pp. 803-819 (2017).
- Gelareh, S.h. and et al. "A branch-and-cut algorithm for the truck dock assignment problem with operational time constraints", *European Journal of Operational Research*, **249**(3), pp. 1144-1152 (2016).
- Molavi, D., Shahmardan, A. and Sheikh Sajadieh, M. "Truck scheduling in a cross docking systems with fixed due dates and shipment sorting", *Computers & Industrial Engineering*, **117**, pp. 29-40 (2018).
- Lee, Y.H., Jung, J.W. and Lee, K.M. "Vehicle routing scheduling for cross-docking in the supply chain", *Computers & Industrial Engineering*, **51**, pp. 247-256 (2006).
- Liao, C.-J., Lin, Y. and Shih, S.C. "Vehicle routing with cross-docking in the supply chain", *Expert Systems with Applications*, **37**, pp. 6868-6873 (2010).
- Mousavi, S.M. and Tavakkoli-Moghaddam, R. "A hybrid simulated annealing algorithm for location and routing scheduling problems with cross-docking in the supply chain", *Journal of Manufacturing Systems*, **32**, pp. 335-347 (2013).
- Dondo, R. and Cerdà, J. "A sweep-heuristic based formulation for the vehicle routing problem with cross-docking", *Computers & Chemical Engineering*, **48**, pp. 293-311 (2013).
- Mousavi, S.M., Vahdani, B., Tavakkoli-Moghaddam, R. and et al. "Location of cross-docking centers and vehicle routing scheduling under uncertainty: a fuzzy

- possibilistic-stochastic programming model”, *Applied Mathematical Modelling*, **38**, pp. 2249-2264 (2014).
17. Morais, V.W.C., Mateus, G.R. and Noronha, T.F. “Iterated local search heuristics for the vehicle routing problem with cross-docking”, *Expert Systems with Applications*, **41**, pp. 7495-7506 (2014).
  18. Moghadam, S., Fatemi Ghomi, S.M.T. and Karimi, B. “Vehicle routing scheduling problem with cross docking and split deliveries”, *Computers & Chemical Engineering*, **69**, pp. 98-107 (2014).
  19. Agustina, D., Lee, C.K.M. and Piplani, R. “Vehicle scheduling and routing at a cross docking center for food supply chains”, *International Journal of Production Economics*, **152**, pp. 29-41 (2014).
  20. Baniamerian, A., Bashiri, M. and Zabihi, F. “A modified variable neighborhood search hybridized with genetic algorithm for vehicle routing problems with cross-docking”, *Electronic Notes in Discrete Mathematics*, **66**, pp. 143-150 (2018).
  21. Maknoon, Y. and Laporte, G. “Vehicle routing with cross-dock selection”, *Computers & Operations Research*, **77**, pp. 254-266 (2017).
  22. Semet, F., Toth, P. and Vigo, D., *Chapter 2: Classical Exact Algorithms for the Capacitated Vehicle Routing Problem. In Vehicle Routing: Problems, Methods, and Applications*, Second Edition. Society for Industrial and Applied Mathematics, pp. 37-57 (2014).
  23. Kirkpatrick, S., Daniel Gelatt, C. and Vecchi, M.P. “Optimization by simulated annealing”, *Science*, **220**(4598), pp. 671-680 (1983).
  24. Suman, B. and Kumar, P. “A survey of simulated annealing as a tool for single and multiobjective optimization”, *J Oper Res Soc*, **57**, pp. 1143-1160 (2005).
  25. Wang, C., Mu, D., Zhao, F. and et al. “A parallel simulated annealing method for the vehicle routing problem with simultaneous pickup-delivery and time windows”, *Computers & Industrial Engineering*, **83**, pp. 111-122 (2015). Gendreau, M. and Potvin, J.Y. Eds., *Handbook of Metaheuristics International Series in Operations Research & Management Science*, 146 pp. 1-39 (2010).