

مدل ریاضی و الگوریتم مبتنی بر اولین برآزش برای مسئله‌ی توزیع میلک‌ران در صنعت خودروسازی با در نظر گرفتن ملاحظات بسته‌بندی

فریور رنجبران (دانشجوی دکتری)

ابوالفضل کاظمی* (استادیار)

دانشکده‌ی مهندسی صنایع و مکانیک، دانشگاه آزاد اسلامی، واحد قزوین

علی حسین زاده کاشان (استادیار)

دانشکده‌ی مهندسی صنایع و سیستم‌ها، دانشگاه تربیت مدرس

مهندسی صنایع و مدیریت شریف، زمستان ۱۳۹۸ (۱۳۹۸)
دوری ۱-۳۵، شماره ۲/۲، ص. ۱۰۵-۱۱۸

سامانه‌ی توزیع در صنعت خودروساز پیچیدگی زیادی دارد و ملاحظات بسیاری باید در نظر گرفته شود. از جمله حجم و وزن پالت‌ها، ظرفیت حجمی و وزنی خودروها، پنجره‌های زمانی و سازگاری محموله‌ها. مدل‌سازی ریاضی سامانه‌ی توزیع در نظر گرفته شده در این مقاله، این گونه ملاحظات دنیای واقعی را با هدف کمینه کردن هزینه‌ی کل حمل‌ونقل در نظر گرفته است. در شبکه‌ی حمل‌ونقل، روش‌های ارسال می‌تواند به صورت ارسال مستقیم و توزیع میلک‌ران باشد. محموله‌های بزرگ به محموله‌های کوچک‌تر شکسته می‌شوند و به صورت مستقیم ارسال می‌شوند، و بقیه‌ی محموله‌ها می‌توانند در توزیع میلک‌ران شرکت کنند. حل مدل ریاضی ارائه شده برای مسائل کوچک، منتج به جواب بهینه می‌شود. برای مسائل بزرگ، الگوریتمی ابتکاری موسوم بر الگوریتم اولین برآزش ارائه شده است. الگوریتم ارائه شده با استفاده از نمونه مسائل تصادفی و داده‌های واقعی در صنعت خودرو ارزیابی شده است و نشان می‌دهد این الگوریتم توانایی ارائه‌ی جواب‌های با کیفیت را در زمان کوتاه داراست.

واژگان کلیدی: میلک‌ران، مدل ریاضی، الگوریتم اولین برآزش، مسئله‌ی برداشت و تحویل.

ranjbaran.f@gmail.com
abkaazemi@gmail.com
a.kashan@modares.ac.ir

۱. مقدمه و ادبیات موضوع

طراحی زنجیره‌ی تأمین، یکی از مسائل مهم است که بر کاهش هزینه اثر به سزایی دارد. از سوی دیگر، برای طراحی زنجیره‌ی تأمین باید محدودیت‌های شرایط دنیای واقعی در نظر گرفته شود. یکی از حوزه‌هایی که به مباحث لجستیک مربوط است، مسئله‌ی برداشت و تحویل (PDP) است که در آن، سفارش‌ها دارای یک مبدأ و مقصد هستند و باید از مبدأ برداشته و در مقصد تحویل داده شوند. مبحث مرتبط دیگر، میلک‌ران^۲ است که روشی از جمع‌بار است که در آن، بار از چند مبدأ تحویل گرفته می‌شود (در خودرو جمع‌بار می‌شود). با استفاده از این روش حمل‌ونقل، از فضای خودرو استفاده بیشتری می‌شود و باعث کاهش هزینه می‌شود. بر این اساس، مطالعه‌ی حاضر یک مدل ریاضی برای مسئله‌ی میلک‌ران ارائه می‌کند که این مدل بر اساس مسئله‌ی برداشت و تحویل است. مروری بر مسئله‌ی برداشت و تحویل در مقالاتی^۱ ارائه شده است. همچنین، روش‌های ابتکاری و فراابتکاری

برای مسایل بزرگ که در زمان مناسب به صورت دقیق قابل حل نیستند، روش‌های فراابتکاری برای حل این مسئله ارائه شده‌اند. برای مثال، یک الگوریتم تریبید ترکیب شده با جستجوی ممنوع ارائه شده است که در آن بعد از چند عدم بهبود در جواب، جستجو

* نویسنده مسئول

تاریخ: دریافت ۱۳۹۷/۹/۲۴، اصلاحیه ۱۳۹۸/۱/۲۴، پذیرش ۱۳۹۸/۲/۲۳

DOI:10.24200/J65.2019.52046.1937

مقایسه‌ی بین نتایج حل دقیق، الگوریتم ژنتیک و نتایج دنیای واقعی برای ۱۵ نمونه مسئله انجام شده است که نشان‌دهنده‌ی بهبود ۱۵٪ تا ۲۵٪ الگوریتم ژنتیک است. همچنین یک رویکرد بهینه‌سازی استوار برای این مسئله در مرجع [۲۱] طراحی شده است. محققان در پژوهش [۲۲] وضعیت فعلی سامانه‌ی میلک‌ران برای تأمین قطعات در توپوتا را بررسی کرده‌اند. در این پژوهش نشان داده شده است که پیاده‌سازی میلک‌ران باعث کاهش هزینه‌های حمل‌ونقل و بهبود شرایط ترافیکی در مناطق شهری تأیید شده است. محققان در مرجع [۲۳] یک مدل ریاضی و الگوریتم جستجوی هارمونی برای شبکه‌ی با حمل‌ونقل مستقیم و میلک‌ران ارائه دادند. خوانندگان علاقه‌مند می‌توانند مروری بر لجستیک قطعات در صنعت خودروسازی را در مرجع [۲۴] مطالعه کنند.

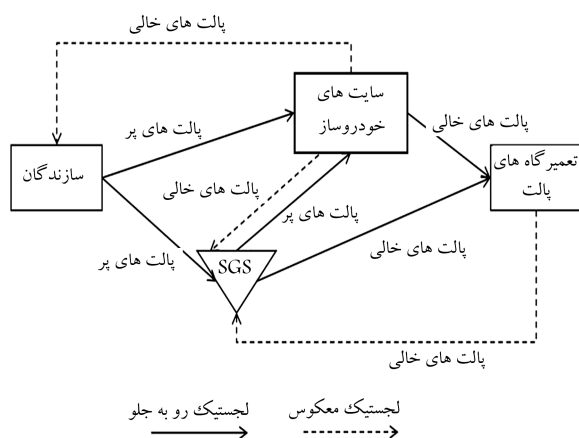
پژوهش حاضر در دو حوزه دارای نوآوری است: در ابتدا مدل ریاضی میلک‌ران برای مسئله‌ی لجستیک قطعات بر اساس مسئله‌ی برداشت و تحویل ارائه می‌شود که ملاحظات مربوط به پالت و برگشت پالت خالی را در نظر می‌گیرد. همچنین یک الگوریتم اولین برزاش مبتنی بر شباهت نیز برای حل مسائل بزرگ ارائه می‌شود و کارایی و اثربخشی آن نیز بررسی می‌شود.

محتوای این پژوهش به صورت زیر تقسیم‌بندی شده است: مسئله‌ی مورد بررسی در بخش ۲ بیان و فرمول‌بندی می‌شود. روش ابتکاری پیشنهادی در بخش ۳ ارائه می‌شود. بخش ۴ به ارزیابی کارایی الگوریتم ارائه شده اختصاص داده شده است. نتیجه‌گیری و پیشنهادها برای تحقیقات آینده در بخش ۵ ارائه می‌شود.

۲. تعریف مسئله و مدل ریاضی

پیکربندی شبکه‌ی لجستیک با توجه به فرایندهای لجستیک روبه‌جلو و معکوس به صورت شکل ۱ است. ناوگان از خودروها برای انجام عملیات مستقیم و معکوس مورد استفاده قرار می‌گیرند. به عبارت دقیق‌تر، پالت‌های پر سفارش‌ها باید از سازندگان به سایت‌های خودروسازی، از سازندگان به SGS و از SGS به سایت‌ها تعمیرگاه‌ها حمل شوند. علاوه بر این، حمل پالت‌های خالی از خودروسازان به SGS و تعمیرگاه‌ها هم جزء جریان مستقیم هستند. همچنین، پالت‌های خالی باید از تعمیرگاه‌ها به SGS و از خودروسازان به سازندگان و SGS حمل شوند که این حمل‌ها جزء لجستیک معکوس هستند.

در ابتدای هر روز، باید برنامه‌ی برای حمل قطعات از گره‌های عرضه به گره‌های تقاضا و حمل پالت‌های خالی از گره‌های مبدأ به گره‌های مقصد تهیه شود. این



شکل ۱. شبکه‌ی لجستیک.

از بهترین جواب آغاز می‌شود. [۷] یک الگوریتم ژنتیک با چند اپراتور تقاطع و جهش ارائه شده است که برای حل مسئله‌ی برداشت و تحویل هم‌زمان با پنجره‌ی زمانی استفاده شده است که این الگوریتم با الگوریتم ژنتیک پایه مقایسه شده است. [۸] در مرجع [۹] مقایسه‌ی از سه الگوریتم ژنتیک، الگوریتم تبرید و الگوریتم تپه‌نوردی برای حل مسئله‌ی برداشت و تحویل با پنجره‌ی زمانی انجام شده است. در پژوهشی دیگر یک اپراتور همسایه‌ساز هوشمند و یک الگوریتم ازدحام ذرات برای مسئله‌ی مسیریابی وسیله‌ی نقلیه با حمل برگشتی و پنجره‌ی زمانی ارائه شده که عملکرد مناسبی در حل مسائل نمونه‌ی مسیریابی وسایل نقلیه با پنجره‌ی زمانی داشته‌اند. [۱۰] یک الگوریتم ژنتیک که با اپراتورهای بهبود درون‌مسیری و بین‌مسیری ترکیب شده، در پژوهش [۱۱] ارائه شده است. محققان یک مدل جدید مبتنی بر جریان برای مسئله‌ی مسیریابی وسایل نقلیه با برداشت و تحویل و پنجره‌ی زمانی ارائه دادند. [۱۲] برای حل مسئله، یک الگوریتم حل برنامه‌ریزی پویا و یک رویکرد آزادسازی لاگرانژ ارائه شده است. مسئله‌ی مسیریابی بالگرد برای حمل‌ونقل در سکوی نفتی پارس جنوبی در تحقیق مربوطه [۱۳] بر مبنای مسئله‌ی برداشت و تحویل مدل‌سازی شده است و یک الگوریتم راهبرد تکاملی گروه‌بندی برای حل آن ارائه شده است. همچنین محققان مسئله‌ی برداشت و تحویل چندسفره با پنجره‌ی زمانی و همگام‌سازی را بررسی کردند. [۱۴] آنها برای این مسئله یک مدل ریاضی مختلط خطی و یک الگوریتم شاخه-برش-قیمت ارائه دادند. محققان گونه‌های مختلف مسئله‌ی مسیریابی وسایل نقلیه با حمل برگشتی، پنجره‌ی زمانی و بارگیری سه‌بعدی را در نظر گرفتند. [۱۵] برای این مسئله، آنها یک الگوریتم دومرحله‌ی بر اساس رویکرد اول بسته‌بندی و سپس مسیریابی ارائه دادند. پیاده‌سازی این الگوریتم برای حمل مسائل نمونه، نشان‌دهنده‌ی کارایی الگوریتم مذکور است. خلاصه‌ی از تحقیقات بررسی شده در جدول ۱ آورده شده است. ویژگی‌های مسئله مانند شکست بار، همگنی یا ناهمگنی خودروها، پنجره‌ی زمانی و مدل ریاضی در این جدول خلاصه شده‌اند. محدودیت‌های جانبی مرتبط با محصولات شامل بسته‌بندی، سازگاری محصولات با یکدیگر و خودروها هستند. علاوه بر این، مسئله‌ی میلک‌ران در صنعت خودروسازی در پیشینه‌ی موضوع بررسی شده است. با توجه به این‌که مسئله‌ی پژوهش حاضر بر اساس شرایط یکی از خودروسازان داخلی تعریف شده است، پژوهش‌های مرتبط در صنعت خودروسازی نیز مرور شده‌اند. در این شرکت خودروسازی، تأمین قطعات توسط یک شرکت لجستیک ثالث (۳PL) به نام سازه‌گستر سایپا (SGS) انجام می‌شود. برای این شرکت، مسئله بدین صورت است که شبکه‌ی لجستیک را به صورتی طراحی کند که حمل‌ونقل به صورت کارا و مؤثر از طریق میلک‌ران انجام شود. در نتیجه، مروری بر پیشینه‌ی مطالعات مرتبط با لجستیک قطعات در صنعت خودروسازی انجام شده است. قابلیت‌های میلک‌ران در لجستیک سبز در پژوهش [۱۶] بررسی شده است که در آن مثال‌هایی از شرکت‌های خودروسازی توپوتا، وباستو و آودی ارائه شده است. در مرجع [۱۷] یک الگوریتم جستجوی محلی برای مسئله‌ی میلک‌ران در لجستیک داخلی ارائه دادند. مفهوم تأمین به هنگام در مرجع [۱۸] بررسی شده است و یک مدل ریاضی برای میلک‌ران بر اساس مسیریابی وسایل نقلیه با پنجره‌ی زمانی ارائه شده است. برای حل این مسئله، یک رویکرد تولید ستون و یک الگوریتم جستجوی ممنوع ارائه شده است. مرجع [۱۹] یک سامانه‌ی ارسال خودروی بلادرنگ را برای تأمین خدمات میلک‌ران تحلیل کرده است. این سامانه دارای هفت بخش است که مدیریت داده‌ها، واکنش به تغییرات بلادرنگ و ایجاد کارایی جواب را انجام می‌دهد. نتایج تحلیل‌ها نشان می‌دهد که الگوریتم بهترین برزاش و 2-exchange بهترین الگوریتم‌ها برای ایجاد جواب اولیه و ایجاد بهبود درون مسیری هستند. در مرجع [۲۰] یک مدل ریاضی با هدف کمینه‌کردن هزینه‌های حمل‌ونقل و موجودی برای یک خودروساز ایرانی ارائه شده است. همچنین

جدول ۱. خلاصه‌ی ویژگی‌های تحقیقات بررسی شده و مطالعه‌ی حاضر.

مرجع	مسئله	مطالعه‌ی موردی	تعداد دپوها	شکست بار	تعداد محصولات	همگنی / ناهمگنی خودروها	پنجره‌ی زمانی	محدودیت‌های جانبی مرتبط با محصولات	مدل ریاضی	روش حل
[۳]	PDPTW ^۱	-	یک	غیرمجاز	یک	همگن	✓	-	✓	الگوریتم‌های شاخه و برش
[۴]	PDPLT ^۲	-	یک	غیرمجاز	یک	همگن	-	✓	✓	الگوریتم شاخه و برش
[۵]	PDPWHO ^۳	-	یک	غیرمجاز	یک	همگن	✓	✓	✓	الگوریتم‌های شاخه-قیمت-برش
[۶]	PDPTWL ^۴	-	یک	غیرمجاز	یک	همگن	✓	✓	-	الگوریتم ژنتیک
[۸]	SDPPTW ^۵	-	یک	غیرمجاز	یک	ناهمگن	✓	-	✓	الگوریتم ژنتیک
[۹]	PDPTW	-	یک	غیرمجاز	یک	همگن	✓	-	✓	الگوریتم ژنتیک، الگوریتم تبرید، الگوریتم تپه‌نوری
[۱۲]	VRPPDTW ^۶	-	یک	غیرمجاز	چند	ناهمگن	✓	✓	✓	آزادسازی لاگرانژ
[۱۴]	MT-PDTWS ^۷	-	یک	غیرمجاز	یک	همگن	✓	-	✓	الگوریتم شاخه - قیمت - برش
[۱۵]	3L-VRPBTW ^۸	-	یک	غیرمجاز	یک	همگن	✓	✓	-	رویکرد اول بسته‌بندی بعد مسیریابی
[۲۵]	VRPSPD ^۹	-	یک	غیرمجاز	یک	همگن	✓	-	✓	بهینه‌سازی انبوه ذرات
[۲۶]	VRPSPD	-	یک	غیرمجاز	یک	ناهمگن	-	-	-	الگوریتم ترکیبی بهینه‌سازی انبوه ذرات
[۲۷]	LRPSPD ^{۱۰}	-	چند	غیرمجاز	یک	همگن	-	-	-	الگوریتم تبرید با شروع چندگانه
[۲۸]	UPDPSL ^{۱۱}	-	یک	غیرمجاز	چند	همگن	✓	-	✓	جستجو با همسایگی متغیر
[۲۹]	MVSPDP ^{۱۲}	-	یک	غیرمجاز	یک	همگن	-	-	✓	جستجوی ممنوع، الگوریتم ژنتیک، جستجوی پراکنده
[۳۰]	GVRP ^{۱۳}	✓	چند	غیرمجاز	چند	ناهمگن	-	-	✓	-
پژوهش حاضر	مسئله‌ی میلک‌ران	✓	مجازی	مجاز برای سفارش‌های بزرگ	چند	ناهمگن	✓	✓	✓	الگوریتم اولین برازش (مبتنی برای شباهت First-fit)

^۱ pickup and delivery problem with time windows

^۲ multiple vehicle pickup and delivery problem with LIFO constraints and maximum time

^۳ pickup and delivery problem with time windows and handing operations

^۴ pickup and delivery problem with time windows and loading constraints

^۵ simultaneous delivery and pickup problem with time windows

^۶ VRP with pickup and delivery and time windows

^۷ multi – trip pickup and delivery problem with time windows and synchronization

^۸ vehicle routing problem with backhauls and time windows

^۹ vehicle routing problem with simultaneous pickup and delivery

^{۱۰} location routing problem with simultaneous pickup and delivery

^{۱۱} unpaired pickup and delivery problem with split loads

^{۱۲} multi – vehicle selective pickup and delivery problem

^{۱۳} green vehicle routing problem

- برنامه حاوی مسیر هر خودرو و مقداری از هر تقاضاست که باید از هر مبدأ به هر مقصد در خودرو حمل شود. هدف این است که برنامه‌ی تهیه شود که هزینه‌های حمل‌ونقل کمینه شود. این مسئله به عنوان یک مسئله‌ی برداشت و تحویل مدل‌سازی شده است که در آن میلک‌ران و ارسال مستقیم در نظر گرفته شده است. مسیرهای ممکن توزیع میلک‌ران در قیاس با ارسال مستقیم در شکل ۲ آمده است. شرایط زیر در مدل در نظر گرفته می‌شود:
- تعداد کافی از خودرو در دسترس است و ترکیب ناوگان توسط مدل تعیین می‌شود.
- خودروها دارای ظرفیت محدود وزنی و حجمی هستند.
- محدودیت‌های پنجره‌ی زمانی برای سفارش‌های پالت پر و خالی وجود دارد.
- زمان بارگیری و تخلیه‌ی هر پالت باید در نظر گرفته شود.
- برخی سفارش‌ها با یکدیگر سازگار نیستند؛ به عبارت دیگر آنها را نمی‌توان در یک خودرو قرارداد.
- برداشت و تحویل می‌تواند در یک گره به صورت هم‌زمان انجام شود، اما برای برخی از سازندگان، عملیات برداشت باید قبل از عملیات تحویل انجام شود (برای هر سفارش).
- اگر محتویات هر خودرو حاوی پالت‌های خالی و پر باشد، برداشت پالت‌های پر باید قبل از برداشت پالت‌های خالی باشد.
- برای پالت‌های پر، یک خودرو می‌تواند فقط یک بار از مجموعه‌ی گره‌های مبدأ

سازنده هستند و گره‌های سمت راست، گره‌های مجازی سازندگان هستند. مسیریابی که ناممکن هستند نیز در این شکل با خط چین نشان داده شده‌اند.

۱.۲. نمادگذاری

مجموعه‌ها، پارامترها و متغیرهای تصمیم مدل ریاضی به صورت زیر تعریف شده‌اند: مجموعه‌ها و پارامترها

در مدل پیشنهادی، سه دسته مجموعه‌ی اصلی تعریف شده است که مربوط به خودروها، گره‌ها و سفارش‌ها هستند. پارامترهای مربوط به مدل پیشنهادی نیز در ادامه تعریف شده‌اند.

مجموعه‌ها

K : مجموعه‌ی خودروها (k):

I : مجموعه‌ی گره‌ها (i, j) که در آن گره اول، یک گره مجازی است؛

M مجموعه‌ی سفارش‌ها (m, m') (MB):

$MB \in M$: مجموعه‌ی سفارش‌های بزرگ (mb).

پارامترها

$BM \setminus - BM \Delta$: پنج عدد بزرگ؛

$Epsilon$: یک عدد کوچک؛

L_k : طول کانتینر خودرو k ؛

WC_k : ظرفیت وزنی خودروی k (Kg)؛

w_m : وزن پالت سفارش m (Kg)؛

$Nw_{m,k}$: تعداد پالت از سفارش m که می‌توان در عرض خودروی k قرار داد؛

$Nh_{m,k}$: تعداد پالت از سفارش m که می‌توان روی هم در خودروی k قرار داد؛

pw_m : عرض پالت سفارش m ؛

LT_m : زمان بارگیری پالت سفارش m ؛

UT_m : زمان تخلیه‌ی پالت سفارش m ؛

f_m : تعداد پالت مربوط به سفارش m ؛

O_m : مبدأ سفارش m ؛

D_m : مقصد سفارش m ؛

O'_i : سفارش‌هایی که مبدأشان گره i است؛

D'_i : سفارش‌هایی که مقصدشان گره i است؛

$comp_{m,m'}$: پارامتری که نشان می‌دهد سفارش‌ها m و m' سازگار هستند ($=1$) یا ناسازگار ($=0$)؛

lb_m : حد پایین پنجره‌ی زمانی برای برداشت سفارش m ؛

ub_m : حد بالای پنجره‌ی زمانی برای تحویل زمانی m ؛

$c_{i,j,k}$: هزینه‌ی حمل‌ونقل از گره i به گره j توسط خودروی k ؛

$t_{i,j,k}$: هزینه‌ی حمل‌ونقل از گره i به گره j توسط خودروی k ؛

Ns : تعداد سازندگان؛

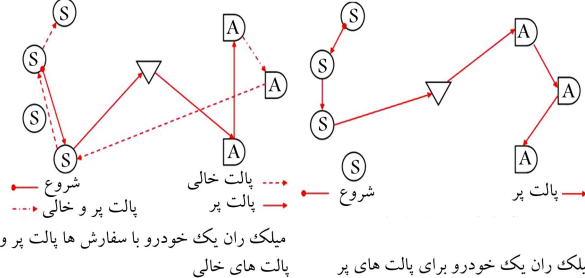
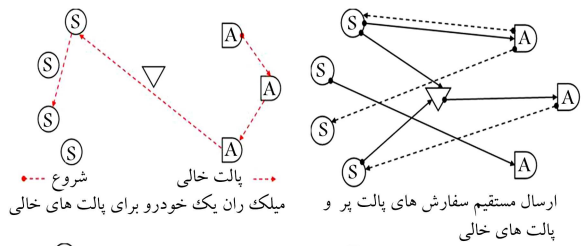
Na : تعداد سایت‌های خودروسازی؛

Nk : تعداد خودروها.

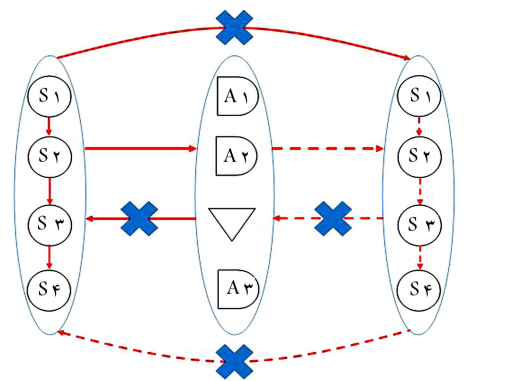
متغیرهای تصمیم

در مدل ریاضی پیشنهادی، ۹ دسته متغیر تصمیم تعریف شده است که چهار دسته متغیر صفر و یک، دو دسته متغیر عدد صحیح و سه دسته متغیر نامنفی وجود دارد که به صورت زیر تعریف شده‌اند:

$X_{i,j,k}$: یک متغیر صفر و یک برابر با ۱ است اگر خودروی k از گره i به گره j برود



شکل ۲. حالات مختلف مسیر.



شکل ۳. حمل‌ونقل بین گره‌های موجود در شبکه.

به مجموعه گره‌های مقصد حرکت کند. به صورت مشابه، برای برگشت پالت‌های خالی، یک خودرو می‌تواند فقط یک‌بار از مجموعه گره‌های مبدأ به مجموعه گره‌های مقصد حرکت کند.

- سفارش‌هایی که کمتر از ظرفیت وزنی و حجمی بزرگ‌ترین خودرو هستند، مجاز به شکست محموله نیستند.
- سفارش‌هایی که بزرگ‌تر از ظرفیت بزرگ‌ترین خودرو هستند، باید شکسته شوند. تمام خودروهایی که به ارسال این سفارش‌های بزرگ تخصیص داده می‌شوند، باید به صورت مستقیم ارسال شوند، به جز یک خودرو که می‌تواند در میلک‌ران شرکت کند.
- طول پالت موازی عرض خودرو است. به عبارت دیگر پالت‌ها به صورت ردیفی در عرض کانتینر خودرو چیده می‌شوند. همچنین، پالت‌های سفارش‌های متفاوت را در یک ردیف کانتینر نمی‌توان قرارداد.
- پالت‌های یک سفارش می‌تواند روی هم قرار داده شوند.

به دلیل این‌که گره‌های سازنده و SGS می‌توانند مبدأ ارسال قطعات و مقصد محموله‌ی پالت خالی باشند، با استفاده از گره‌های مجازی سازنده، این شبکه مدل شده است. شکل ۳ نشان‌دهنده‌ی این مفهوم است، که گره‌های سمت چپ گره‌های

$$\sum_{k \in K} Y_{i,k} \geq 1 \quad \forall i \in I, k \in K \quad (7)$$

$$\sum_{k \in K} X_{i,j,k} = 0 \quad \forall i \in I \quad (8)$$

$$X_{i,j,k} + X_{j,i,k} \leq 1 \quad \forall i \in I, j \in I, k \in K \quad (9)$$

$$\sum_{\substack{i \in I, i \neq 1, \\ i \leq ns+1}} \sum_{\substack{j \in I, j \geq ns+2, \\ j \leq ns+na+1}} X_{i,j,k} \leq 1 \quad \forall k \in K \quad (10)$$

$$\sum_{\substack{i \in I, i \geq ns+2, \\ i \leq ns+na+1}} \sum_{\substack{j \in I, \\ j \geq ns+na+2}} X_{i,j,k} \leq 1 \quad \forall k \in K \quad (11)$$

$$\sum_{\substack{i \in I, \\ i \geq ns+na+2}} \sum_{\substack{j \in I, j \neq 1, \\ j \leq ns+1}} X_{i,j,k} = 0 \quad \forall k \in K \quad (12)$$

$$\sum_{\substack{i \in I, i \neq 1, \\ i \leq ns+1}} \sum_{\substack{j \in I, \\ j \geq ns+na+2}} X_{i,j,k} = 0 \quad \forall k \in K \quad (13)$$

$$\sum_{\substack{i \in I, i \geq ns+2, \\ i \leq ns+na+1}} \sum_{\substack{j \in I, j \neq 1, \\ j \leq ns+1}} X_{i,j,k} = 0 \quad \forall k \in K \quad (14)$$

$$\sum_{\substack{i \in I, \\ i \geq ns+na+2}} \sum_{\substack{j \in I, j \geq ns+2, \\ j \leq ns+na+1}} X_{i,j,k} = 0 \quad \forall k \in K \quad (15)$$

محدودیت‌های برداشت و تحویل

$$P_{m,k} \leq f_m \times Y_{i,k} \quad \forall i \in I, i \neq 1, k \in K, m \in O'_i \quad (16)$$

$$\sum_{k \in K} P_{m,k} = f_m \quad \forall m \in M \quad (17)$$

$$D_{m,k} \leq f_m \times Y_{i,k} \quad \forall i \in I, i \neq 1, k \in K, m \in D'_i \quad (18)$$

$$P_{m,k} = D_{m,k} \quad \forall k \in K, m \in M \quad (19)$$

$$\sum_{m \in O'_i} P_{m,k} + \sum_{m \in D'_i} D_{m,k} \geq Y_{i,k} \quad \forall i \in I, i \neq 1, k \in K \quad (20)$$

محدودیت‌های مربوط به حجم

$$U_{i,k} + \sum_{m \in O'_j} \left[\frac{P_{m,k}}{Nw_{m,k} \times Nh_{m,k}} \right] \times pw_m -$$

$$\sum_{m \in D'_j} \left[\frac{D_{m,k}}{Nw_{m,k} \times Nh_{m,k}} \right] \times pw_m \quad \forall i \in I, j \in I,$$

$$\leq U_{j,k} + (1 - X_{i,j,k}) \times BM \quad j \neq 1, i \neq j, k \in K \quad (21)$$

$$U_{i,k} \leq L_k \quad \forall i \in I, k \in K \quad (22)$$

محدودیت‌های مربوط به وزن

$$W_{i,k} + \sum_{m \in O'_j} w_m P_{m,k} - \sum_{m \in D'_j} w_m D_{m,k} \leq \quad \forall i \in I, j \in I, W_{j,k}$$

$$+ (1 - x_{i,j,k}) \times BM \quad j \neq 1, i \neq j, k \in K \quad (23)$$

$$W_{i,k} \leq WC_k \quad \forall i \in I, k \in K \quad (24)$$

و برابر با صفر است در غیر این صورت؛

$Y_{i,k}$: یک متغیر صفر و یک برابر با 1 است اگر خودروی k گره i را ملاقات کند و برابر با صفر است در غیر این صورت؛

$\Delta_{m,k}$: یک متغیر صفر و یک برابر با 1 است اگر سفارش m در خودروی k باشد و برابر با صفر است در غیر این صورت؛

$\sigma_{mb,k}$: یک متغیر صفر و یک که برابر با یک است اگر سفارش بزرگ mb (که شکسته شده است) به صورت مستقیم توسط خودروی k ارسال می‌شود و برابر با 0 است در غیر این صورت؛

$P_{m,k}$: یک متغیر عدد صحیح که نشان‌دهنده‌ی تعداد پالت سفارش m است که توسط خودروی k برداشت می‌شود؛

$D_{m,k}$: یک متغیر عدد صحیح که نشان‌دهنده‌ی تعداد پالت سفارش m است که توسط خودروی k تحویل داده می‌شود؛

$U_{i,k}$: یک متغیر نامنفی که نشان‌دهنده‌ی طول اشغال شده از خودروی k است وقتی که گره i را ترک می‌کند؛

$W_{i,k}$: یک متغیر نامنفی که نشان‌دهنده‌ی مجموع وزن پالت‌های داخل خودروی k است وقتی که گره i را ترک می‌کند؛

$T_{i,k}$: یک متغیر نامنفی که نشان‌دهنده‌ی زمان ملاقات گره i توسط خودروی k است.

۲.۲. محدودیت‌های منطقی

برخی از شرایط عملیاتی باید در مدل در نظر گرفته شوند. این شرایط را می‌توان به صورت منطقی نوشت و سپس به محدودیت‌های خطی تبدیل کرد. این محدودیت‌ها مربوط به سازگاری پالت‌ها و شکست سفارش‌های بزرگ هستند:

۱. اگر پالت‌های دو سفارش ناسازگار بودند، آنها را نمی‌توان در یک خودرو قرارداد.

۲. سفارش‌هایی که بیشتر از ظرفیت بزرگ‌ترین خودرو هستند، شکسته شوند. این سفارش‌های بزرگ به صورت مستقیم ارسال می‌شوند، به جز یکی که کمتر از یک محموله‌ی خودرو است و می‌تواند در میله‌ران شرکت کند.

۳.۲. مدل ریاضی

با استفاده از نمادهای فوق، مدل ریاضی عدد صحیح مختلط (MIP) ^۴ مسئله به صورت زیر است:

$$\text{Minimize ObjF} = \sum_{i \in I} \sum_{j \in I} \sum_{k \in K} c_{i,j,k} \times X_{i,j,k} \quad (1)$$

محدودیت‌های مسیریابی

$$\sum_{j \in I} X_{1,j,k} \leq 1 \quad \forall k \in K \quad (2)$$

$$\sum_{i \in I} X_{i,1,k} \leq 1 \quad \forall k \in K \quad (3)$$

$$\sum_{j \in I} \sum_{k \in K} X_{1,j,k} \leq nk \quad (4)$$

$$\sum_{i \in I} X_{i,j,k} = \sum_{i \in I} X_{j,i,k} \quad \forall i \in I, k \in K \quad (5)$$

$$\sum_{j \in I} X_{i,j,k} = Y_{i,k} \quad \forall i \in I, k \in K \quad (6)$$

محدودیت‌های مربوط به سازگاری پالت‌ها

$$\Delta_{m,k} + \Delta_{m',k} \leq comp_{m,m'} + 1 \quad \forall m \in M, m' \in M, k \notin K \quad (25)$$

$$\left[\frac{P_{m,k}}{Nw_{m,k} \times Nh_{m,k}} \right] \times pw \leq L_k \times \Delta_{m,k} \quad \forall m \in M, k \in K \quad (26)$$

$$P_{m,k} \geq \Delta_{m,k} \quad \forall m \in M, k \in K \quad (27)$$

محدودیت عدم شکست بار برای محموله‌های کوچک

$$\sum_{k \in K} \Delta_{m,k} = 1 \quad \forall m \in M - MB \quad (28)$$

محدودیت شکست بار برای محموله‌های بزرگ

$$\Delta_{m,k} \leq (1 - \Delta_{mb,k}) + (1 - \sigma_{m,k}) \quad \forall m \in M, mb \in MB, m \neq MB, k \in K \quad (29)$$

$$\sigma_{m,b,k} \leq \Delta_{m,b,k} \quad \forall mb \in MB, k \in K \quad (30)$$

$$\sum_{k \in K} \sigma_{m,b,k} = \sum_{k \in K} \Delta_{m,b,k} - 1 \quad \forall mb \in MB \quad (31)$$

محدودیت‌های مربوط به زمان

$$T_{i,k} + \sum_{m \in O'_i} LT_m P_{m,k} + \sum_{m \in D'_i} UT_m D_{m,k} + t_{i,j,k} \leq T_{j,k} (1 - X_{i,j,k}) \times BM^3 \quad (32)$$

$$\forall i \in I, j \in I, j \neq 1, i \neq J, k \in K$$

$$T_{j,k} \geq T_{i,k} - (1 - \Delta_{m,k}) \times BM^4 \quad (33)$$

$$\forall k \in K, m \in M, i = O_m, j = D_m$$

$$T_{i,k} \geq lb_m \Delta_{m,k} \quad \forall m \in M, i \in o_m, j = D_m \quad (34)$$

$$T_{i,k} \leq Ub_m (1 - \Delta_{m,k}) \times BM^5 \quad \forall m \in M, i = D_m, k \in K \quad (35)$$

$$X_{i,j,k}, Y_{i,k}, \gamma_{k'} \Delta_{m,k'}, Z_{i,j,k'}, \sigma_{mb,k} \in \{0, 1\}; \quad (36)$$

$$P_{m,k}, D_{m,k} \in Z; U_{i,k}, W_{i,k}, T_{i,k} \geq 0$$

$$P_{m,k} \leq Q_{m,k} \quad \forall m \in M, k \in K \quad (37)$$

$$Q_{m,k} \leq \frac{P_{m,k}}{Nw_{m,k} \times Nh_{m,k}} + 1 - \varepsilon \quad \forall m \in M, k \in K \quad (38)$$

$$\frac{D_{m,k}}{Nw_{m,k} \times Nh_{m,k}} \leq E_{m,k} \quad \forall m \in M, k \in K \quad (39)$$

$$E_{m,k} \leq \frac{D_{m,k}}{Nw_{m,k} \times Nh_{m,k}} + 1 - \varepsilon \quad \forall m \in M, k \in K \quad (40)$$

$$U_{i,k} + \sum_{m \in Q'_j} Q_{m,k} \times pw_m - \sum_{m \in D'_j} E_{m,k} \times pw_m \leq U_{j,k} + (1 - X_{i,j,k}) \times BM^1$$

نمی‌تواند برود. محدودیت‌های ۱۴ ضمانت می‌کنند که یک خودرو از خودروسازان به گروه‌های تأمین‌کننده برگردد. محدودیت ۱۵ از حرکت یک خودرو از گروه‌های مجازی سازندگان به گروه‌های خودروسازان، جلوگیری می‌کند. محدودیت ۱۶ تعیین می‌کند که اگر یک خودرو یک گروه را ملاقات کرد و این گروه مبدأ یک سفارش است، بتواند آن سفارش را بردارد. محدودیت ۱۷ از برداشته شدن تمام سفارش‌ها اطمینان حاصل می‌کند. محدودیت‌های ۱۸ مشابه محدودیت‌های ۱۶ هستند با این تفاوت که برای تحویل سفارش‌ها هستند. محدودیت ۱۹ بیان می‌کند که تعداد پالت‌های برداشتی توسط هر خودرو باید با تعداد پالت‌های تحویلی توسط آن خودرو برابر باشد. مجموعه محدودیت‌های ۲۰ بیان می‌کند که اگر گروهی توسط خودرویی ملاقات شد، حداقل یک پالت باید از آن گروه برداشته شود. طول اشغال شده از کانتینر یک خودرو بعد از ترک یک گروه توسط محدودیت‌های ۲۱ محاسبه می‌شود. محدودیت طولی کانتینر خودرو توسط ۲۲ تضمین می‌شود. محدودیت‌های ۲۳ و ۲۴ شرایطی مشابه ۲۱ و ۲۲ برای وزن خودرو تضمین می‌کنند. سازگار بودن سفارش‌های داخل هر خودرو توسط ۲۵ بررسی می‌شود. محدودیت‌های ۲۶ و ۲۷ به ترتیب حداقل و حداکثر تعداد پالتی را که می‌تواند توسط یک خودرو برداشته شود، کنترل می‌کند. محدودیت ۲۸ تضمین می‌کند که سفارش‌های کوچک شکسته نشوند. مجموعه محدودیت‌های ۲۹-۳۱ برای شکسته شدن هر سفارش بزرگ هستند، به گونه‌ای که همه سفارش‌های شکسته شده به صورت مستقیم ارسال شوند و یکی بتواند در میلک‌ران شرکت کند. زمان رسیدن هر خودرو به هر گروه مسیری را محدودیت ۳۲ محاسبه می‌کند. محدودیت ۳۳ الزام می‌کند که هر گروه برداشت قبل از گروه تحویل مربوطه ملاقات شود. محدودیت‌های ۳۴ و ۳۵، محدودیت‌های پنجره‌ی زمانی سفارش‌ها هستند. در نهایت، متغیرها و نوعشان در ۳۶ بیان شده‌اند.

۴.۲. خطی‌سازی تابع هدف

تابع سقف در محدودیت‌های ۲۱ و ۲۶ استفاده شده‌اند، باعث غیرخطی شدن مدل هستند. این عبارات غیرخطی با استفاده از تغییر متغیر، خطی و به مدل اضافه می‌شوند. برای خطی‌سازی محدودیت‌ها، دو تغییر متغیری به صورت $E_{m,k} = \left[\frac{D_{m,k}}{Nw_{m,k} \times Nh_{m,k}} \right]$ و $Q_{m,k} = \left[\frac{P_{m,k}}{Nw_{m,k} \times Nh_{m,k}} \right]$ انجام می‌دهیم که $E_{m,k}$ و $Q_{m,k}$ متغیرهای عدد صحیح هستند. بر اساس رابطه‌ی $1 < a \leq [a] < a + 1$ می‌توانیم روابط ۳۷-۴۰ را بنویسیم. با قرار دادن $Q_{m,k}$ و $E_{m,k}$ در ۲۱ و ۲۶ به جای توابع سقف، ۴۱ و ۴۲ حاصل می‌شوند. در نتیجه، محدودیت‌های ۳۷-۴۲ با محدودیت‌های ۲۱ و ۲۶ جایگزین می‌شوند:

$$\frac{P_{m,k}}{Nw_{m,k} \times Nh_{m,k}} \leq Q_{m,k} \quad \forall m \in M, k \in K \quad (37)$$

$$Q_{m,k} \leq \frac{P_{m,k}}{Nw_{m,k} \times Nh_{m,k}} + 1 - \varepsilon \quad \forall m \in M, k \in K \quad (38)$$

$$\frac{D_{m,k}}{Nw_{m,k} \times Nh_{m,k}} \leq E_{m,k} \quad \forall m \in M, k \in K \quad (39)$$

$$E_{m,k} \leq \frac{D_{m,k}}{Nw_{m,k} \times Nh_{m,k}} + 1 - \varepsilon \quad \forall m \in M, k \in K \quad (40)$$

$$U_{i,k} + \sum_{m \in Q'_j} Q_{m,k} \times pw_m - \sum_{m \in D'_j} E_{m,k} \times pw_m \leq U_{j,k} + (1 - X_{i,j,k}) \times BM^1$$

تابع هدف ۱ هزینه‌ی کل حمل‌ونقل را نشان می‌دهد که باید کمینه شود. مجموعه‌ی محدودیت‌های ۲ و ۳ محدودیت‌های درجه هستند. مجموعه محدودیت‌های ۴ تعداد خودروها را محدود می‌کنند. محدودیت‌های ۵ پیوستگی مسیر را تضمین می‌کنند. محدودیت‌های ۶ و ۷ بررسی می‌کنند که آیا یک گروه ملاقات شده است یا خیر. مجموعه محدودیت‌های ۸ و ۹ از ایجاد حلقه در هر گروه و حرکت‌های برگشتی جلوگیری می‌کنند. مجموعه محدودیت‌های ۱۰ تضمین می‌کنند که یک خودرو حداکثر یک بار از مجموعه گروه‌های سازندگان به مجموعه گروه‌های خودروسازان حرکت کند. محدودیت‌های ۱۱ تضمین می‌کنند که یک خودرو حداکثر یک بار از خودروسازان به گروه‌های مجازی سازندگان برود. محدودیت‌های ۱۲ و ۱۳ تضمین می‌کنند که یک خودرو از گروه‌های مجازی سازندگان به گروه‌های سازندگان و عکس آن

به هم باشند، شبیه‌تر هستند و در نتیجه احتمال این‌که در یک خودرو قرار بگیرند، بیشتر است. به همین دلیل، یک گراف انطباق دوبرخی ایجاد می‌شود که در هر سمت آن گره‌ها نماینده سفارش‌ها هستند. این گره‌ها با وزن ضریب شباهت به هم متصل می‌شوند که این ضریب شباهت از متوسط شباهت مبدأ (رابطه‌ی ۴۳) و شباهت مقصد (رابطه‌ی ۴۴) به دست می‌آید. این ضرایب شباهت اعدادی بین ۰ و ۱ هستند که ۰ نشان‌دهنده دورترین مبادی (مقاصد) است و ۱ نشان‌دهنده یکسان بودن مبادی (مقاصد) است.

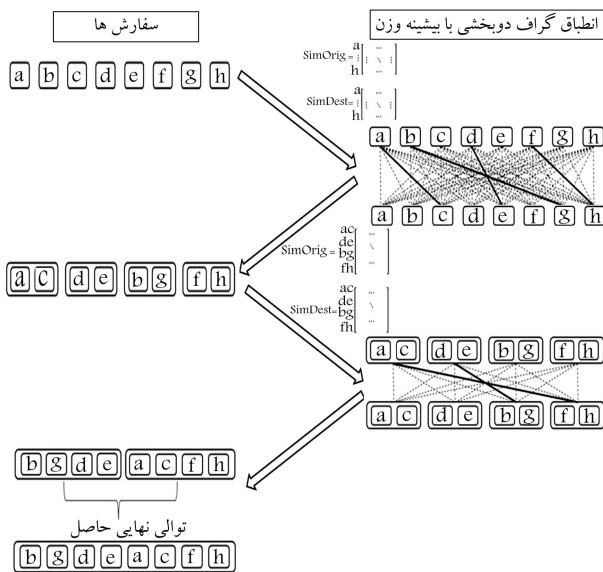
$$SimOrig(m, m') = 1 - \frac{dist(Orig_m, Orig_{m'})}{\max(dist)} \quad (43)$$

$$SimDest(m, m') = 1 - \frac{dist(Dest_m, Dest_{m'})}{\max(dist)} \quad (44)$$

مسئله‌ی انطباق گراف دوبرخی با حداکثر وزن حاصل به صورت حریصانه حل می‌شود. به عبارت دیگر، مقدار حداکثر شباهت پیدا می‌شود و سطر و ستون آن با هم منطبق می‌شوند و سطر و ستون حذف می‌شوند. این رویه مکرراً انجام می‌شود تا زمانی که هیچ سفارشی باقی نماند. این رویه به صورت گرافیکی در شکل ۴ نشان داده شده است. در این شکل، ۸ سفارش با هم منطبق شده‌اند تا ۴ جفت سفارش ایجاد کنند. این سفارش‌ها نیز با هم منطبق شدند تا گروه‌هایی با ۴ سفارش ایجاد کنند. در انتها، یک توالی ایجاد شده است تا به الگوریتم اولین برارزش به عنوان ورودی داده شود.

۲.۱.۳. الگوریتم اولین برارزش

الگوریتم اولین برارزش که بر مسئله‌ی پژوهش حاضر منطبق شده است، در شبه کد زیر آورده شده است. دو حالت در این الگوریتم بررسی شده‌اند. ابتدا، شکستن بارهای بزرگ تا جایی انجام شده است که اندازه‌ی پالت‌های باقی‌مانده از ظرفیت بزرگ‌ترین خودرو کمتر باشد. سفارش‌های شکسته شده به صورت تصادفی به خودروها تخصیص داده شده‌اند تا به صورت مستقیم ارسال شوند. پالت‌های باقی‌مانده و بقیه‌ی سفارش‌ها می‌توانند در میک‌ران شرکت داشته باشند. در ادامه، پالت‌ها به صورت ردیفی در خودروها چیده می‌شوند به گونه‌ی که طول پالت موازی با عرض



شکل ۴. الگوریتم SBO که به یک نمونه مسئله با ۵ سفارش اعمال شده است.

$$\forall i \in I, j \in I, j \neq i, i \neq j, k \in K \quad (41)$$

$$Q_{m,k} \times pw_m \leq L_k \times \Delta_{m,k} \quad \forall m \in M, k \in K \quad (42)$$

۳. روش حل

مدل ارائه شده فقط در اندازه‌های کوچک مسئله قابل حل در زمان مناسب است. برای حل مسائل دنیای واقعی، ارائه‌ی یک الگوریتم کارا لازم است. مسئله‌ی پژوهش حاضر مشابه مسائل گروه‌بندی است؛ زیرا پالت‌های سفارش‌ها (آتم‌ها) باید در خودروها (گروه‌ها) قرار گیرند. در نتیجه، الگوریتم‌های مسائل گروهی می‌توانند برای این مسئله استفاده شوند. در این پژوهش، یک الگوریتم مبتنی بر الگوریتم اولین برارزش (FF) ارائه می‌شود. این الگوریتم بدین صورت است که ابتدا یک ماتریس مشابهت تعریف می‌شود و با استفاده از آن برای حل مسئله‌ی انطباق گراف دوبرخی با وزن بیشینه، یک توالی از سفارش‌ها ایجاد می‌شود. سپس سفارش‌ها به اولین خودرو که تمام محدودیت‌های مسئله در آن لحاظ شده است، تخصیص داده می‌شود. بعد از این که یک جواب اولیه‌ی شدنی ایجاد شد، دو الگوریتم بهبوددهنده کاهش و ادغام روی جواب اعمال می‌شوند. الگوریتم کاهش، سفارش‌ها را در خودروهای کوچک‌تر قرار می‌دهد، در حالی که الگوریتم ادغام، سفارش‌های دو خودرو را در یک خودرو قرار می‌دهد. شبه کد الگوریتم روش حل به صورت زیر است:

روش حل

پارامترها را مقداردهی اولیه کن؛

شروع

(ایجاد یک جواب اولیه‌ی شدنی)؛

سفارش‌های بزرگ را مشخص کن، آنها را بشکن و به خودروها برای ارسال مستقیم، تخصیص بده؛

با استفاده از الگوریتم SBO (بر اساس قسمت ۱.۱.۳) یک توالی از سفارش‌های باقیمانده ایجاد کن؛

الگوریتم اولین برارزش را اعمال کن (بر اساس قسمت ۲.۱.۳)؛

(اعمال الگوریتم‌های ابتکاری بهبوددهنده الگوریتم کاهش را اعمال کن (بر اساس قسمت ۵.۱.۳)؛

الگوریتم ادغام را اعمال کن (بر اساس قسمت ۶.۱.۳)؛

پایان

۱.۱.۳. الگوریتم اولین برارزش مبتنی بر شباهت

الگوریتم اولین برارزش برای ایجاد یک جواب اولیه استفاده شده است. این الگوریتم، دو مرحله دارد که یکی برای کار با سفارش‌های بزرگ و دیگری برای سفارش‌های کوچک است. در مرحله اول، سفارش‌های بزرگ شکسته می‌شوند تا به صورت مستقیم ارسال شوند. سپس، پالت‌های باقیمانده از سفارش‌های بزرگ و بقیه‌ی سفارش‌های کوچک با استفاده از الگوریتم ایجاد توالی مبتنی بر شباهت (SBO) ۶، به صورت یک توالی ایجاد می‌شوند و با استفاده از الگوریتم اولین برارزش به خودروها تخصیص می‌یابند.

۱.۱.۳.۱. الگوریتم ایجاد توالی مبتنی بر شباهت

سفارش‌ها باید به ترتیب خاصی به خودروها تخصیص داده شوند. این ترتیب بر اساس شباهت سفارش‌ها به نسبت فاصله‌ی بین مبدأ و مقصدشان است. منطق این تعریف به این صورت است که اگر دو سفارش، دارای مبادی (مقاصد) نزدیک

$$\sum_{\substack{i \in I, i \geq ns+2 \\ i \leq ns+na+1}} \sum_{\substack{j \in I, \\ j \geq ns+na+2}} X_{i,j} \leq 1 \quad (51)$$

$$\sum_{\substack{i \in I, \\ i \geq ns+na+2}} \sum_{\substack{j \in I, j \neq 1, \\ j \leq ns+1}} X_{i,j} = 0 \quad (52)$$

$$\sum_{\substack{i \in I, i \neq 1, \\ i \leq ns+1}} \sum_{\substack{j \in I, \\ j \geq ns+na+2}} X_{i,j} = 0 \quad (53)$$

$$\sum_{\substack{i \in I, i \neq 1, \\ i \leq ns+1}} \sum_{\substack{j \in I, \\ j \geq ns+na+2}} X_{i,j} = 0 \quad (54)$$

$$\sum_{\substack{i \in I, \\ i \geq ns+na+2}} \sum_{\substack{j \in I, j \geq ns+2, \\ j \leq ns+na+1}} X_{i,j} = 0 \quad (55)$$

محدودیت‌های مربوط به حجم

$$u_i + \sum_{m \in O'_j} ol_m - \sum_{m \in D'_j} ol_m \leq U_j + (1 - X_{i,j}) \times BM \quad (56)$$

$$\forall i \in I, j \in I, j \neq 1, i \neq j \quad (56)$$

$$U_i \leq L \quad \forall i \in I \quad (57)$$

محدودیت‌های مربوط به وزن خودرو

$$W_i + \sum_{m \in O'_j} w_m P_m - \sum_{m \in D'_j} w_m D_m \leq W_j + (1 - x_{i,j}) \times BM \quad (58)$$

$$\forall i \in I, j \in I, j \neq 1, i \neq j \quad (58)$$

$$W_i \leq WC \quad \forall i \in I \quad (59)$$

محدودیت‌های مربوط به زمان

$$T_i + \sum_{m \in O'_i} LT_m P_{m,k} + \sum_{m \in D'_i} UT_m D_{m,k} + t_{i,j,k} \leq T_{j,k} (1 - X_{i,j,k}) \times BM \quad (60)$$

$$\forall i \in I, j \in I, j \neq 1, i \neq j \quad (60)$$

$$T_j \geq T_i \quad m \in M, i = orig_m, j = Dest_m \quad (61)$$

$$T_j \geq T_i \quad m \in M, i = orig_m \quad (62)$$

$$T_i \geq lb_m \quad \forall m \in M, i \in orig_m \quad (62)$$

$$T_i \leq Ub_m \quad \forall m \in M, i \in Dest_m \quad (63)$$

$$X_{i,j} \in \{0, 1\}; U_i, W_i, T_i, U_i \geq 0 \quad (64)$$

در فرایند تولید جواب، سفارش‌ها یک به یک در خودروهای موجود به طور موقت قرار داده شده‌اند و بر اساس آن، هر بار، یک مسیر شدنی و بهینه از نظر رعایت محدودیت حجم و وزن خودرو، پنجره‌های زمانی، سازگاری سفارش‌ها تعیین می‌شود. در نهایت، تخصیص نهایی هر سفارش بر اساس منطق نخستین برآزش به نخستین خودرویی است که برنامه‌ی مسیر برای آن شدنی است. برای تعیین مسیر بهینه از مدل ریاضی ۴۵-۶۴ استفاده می‌شود که در آن محدودیت‌های ۴۶-۴۹، محدودیت‌های مسئله‌ی فروشنده‌ی دوره‌گرد هستند و بقیه محدودیت‌های جانبی‌اند. خوشبختانه تعداد سفارش‌های تخصیص‌یافته به یک خودرو در عمل زیاد نیست (کمتر از ۱۰ است) و می‌توان مدل ریاضی را به سهولت در کسری از ثانیه حل کرد.

خودرو باشد. در نتیجه، طول اشغال شده از هر خودرو، حاصل ضرب تعداد ردیف پالت در عرض پالت است.

الگوریتم اولین برآزش

برای هر سفارش

برای هر خودروی باز و یک خودروی جدید

سفارش را موقتاً به یک خودرو تخصیص بده و شدنی بودن جواب را بررسی کن؛

اگر شدنی بود

سفارش را به خودرو تخصیص بده؛

خروج از برای (Break)؛

پایان اگر

پایان برای

پایان برای

۳.۱.۳. مسیریابی و حفظ شدنی بودن

تمام محدودیت‌های مسئله در الگوریتم در نظر گرفته می‌شوند، به گونه‌ی که جواب همواره شدنی باشد. خروجی الگوریتم اولین برآزش، از لحاظ طول خودرو، وزن خودرو و سازگاری سفارش‌ها شدنی است. برای در نظر گرفتن پنجره‌ی زمانی، مسیریابی هر خودرو باید انجام شود. پس از تخصیص هر سفارش به یک خودرو، شدنی بودن آن از طریق یک مدل ریاضی خلاصه شده تعیین می‌شود. این مدل برای مسیریابی خودرو و بررسی شدنی بودن آن است. تعریف پارامترها و متغیرها مشابه مدل ریاضی اولیه است با این تفاوت که اندیس مربوط به خودرو (k)، حذف شده است. در نتیجه، مسئله مشابه مسئله‌ی فروشنده‌ی دوره‌گرد است. علاوه بر این، مجموعه‌ها و پارامترهای مسئله روی گره‌هایی تعریف می‌شوند که برای برداشت یا تحویل سفارش‌های داخل آن خودرو، ملاقات شوند. پارامترها و متغیرهای اضافی تعریف شده و مدل خلاصه شده به صورت زیر هستند:

متغیر

u_i : یک متغیر نامنفی که برای حذف زیر تور تعریف شده است

پارامتر

n : تعداد گره‌های تعریف شده در مدل خلاصه شده

ol_m : طول اشغال شده از خودرو توسط پالت‌های سفارش m

مدل خلاصه شده:

$$\text{Minimize ObjF} = \sum_{i \in I} \sum_{j \in I} c_{ij} \times X_{i,j} \quad (45)$$

محدودیت‌های مسیریابی

$$\sum_{j \in I} X_{1,j} \leq 1 \quad \forall i \in I \quad (46)$$

$$\sum_{j \in I} X_{i,1} \leq 1 \quad \forall j \in I \quad (47)$$

$$u_i - u_j + nX_{i,j} \leq n - 1 \quad \forall j \in I, j \in I, i, j \geq 2, i \neq j \quad (48)$$

$$X_{i,j} = 0 \quad \forall i \in I \quad (49)$$

$$\sum_{\substack{i \in I, i \neq 1, \\ i \leq ns+1}} \sum_{\substack{j \in I, j \geq ns+2, \\ j \leq ns+na+1}} X_{i,j} \leq 1 \quad (50)$$

خلاصه شده، چک کن
اگر سازگار و شدنی بود
سفارش های v_1 و v_2 را به صورت دائم در vp قرار بده؛
$$\{vp\} \cup \{v_1, v_2\} = Vo$$

از حلقه خارج شو؛

پایان اگر

پایان برای

$$\text{counter} = \text{counter} + 1$$

پایان تا زمانی که

شایان ذکر است که الگوریتم ارائه شده دارای سه مرحله است که در مرحله نخست یک جایگشت از سفارش ها برای تخصیص به خودروها تولید می شود؛ در مرحله دوم سفارش ها به خودروها بر اساس منطق نخستین برآزش تخصیص می یابد به گونه ای که تمام محدودیت ها شدنی باقی بمانند؛ و در مرحله سوم از الگوریتم های بهبوددهنده کاهش و ادغام برای بهبود جواب ساخته شده در مرحله دوم استفاده می شود. در مراحل دوم و سوم از عملیات تصادفی استفاده می شود. با توجه به این که ناوگان همگن نیست، هر بار که یک خودرو جدید به خدمت گرفته می شود و سفارش ها به آن تخصیص می یابد، یک خودرو به تصادف از میان خودروهای موجود (به علت نبود اطلاعات پیرامون تناسب خودروی مورد نیاز) انتخاب می شود. همچنین در مرحله سوم نیز خودروها به طور تصادفی انتخاب و سفارش های آنها در یک خودرو با یکدیگر ادغام می شوند.

۴. محاسبات و نتایج

برای این که مدل ریاضی ارائه شده و الگوریتم ابتکاری مورد ارزیابی قرار گیرند، تعدادی نمونه تصادفی ایجاد شده است و این نمونه ها با استفاده از مدل ریاضی و الگوریتم ابتکاری ارائه شده حل شده اند. نمونه های تصادفی به گونه ای ایجاد شده اند که نماینده مسئله دنیای واقعی باشند. سفارش های روزانه شامل سفارش های پالت های پر (سفارش های روبه جلو) و سفارش های پالت خالی (سفارش های برگشتی) هستند. برای سفارش های روبه جلو، بر اساس تعداد سفارش ها، فهرستی از شناسه های فنی و تعداد قطعات به صورت تصادفی ایجاد می شود. بر اساس این اطلاعات، تأمین کننده و نوع پالت و تمام اطلاعات مربوط به بسته بندی استخراج می شود. تعداد قطعات برای هر سفارش از یک شناسه فنی مشخص از توزیع یکنواخت تولید می شود. حد بالا و پایین بازه نیز توسط داده های تاریخی موجود برای آن شناسه فنی تعیین می شود. حد پایین/بالا توسط حداقل/حداکثر تعداد قطعات در سفارش هایی که از آن شناسه فنی تحویل شده اند، تعیین می شود. داده های مربوط به مقصد و حد بالا و پایین پنجره زمانی به صورت تصادفی ایجاد می شوند. برای سفارش های برگشتی، نام پالت، تعداد پالت ها، مبدأ و مقصد و حد بالا و پایین پنجره زمانی به صورت تصادفی تولید می شوند. اندازه نمونه ها بر اساس تعداد سفارش ها تعیین می شود که بین ۶ تا ۱۲۲ سفارش است (ستون دوم جدول ۱.۲).

همچنین مسئله روزانه در SGS به صورت زیر است. در ابتدای هر روز، اطلاعات جدول ۳ برای هر سفارش مشخص است. با وجود این اطلاعات، هدف مسئله تعیین مجموعه خودروها، مسیر هر کدام و محموله هایی که باید در هر کدام

خوشبختانه در سامانه توزیع میلهک ران قطعات خودرو، که در مطالعه ی موردی مد نظر قرار گرفته است، تعداد سفارش هایی که به یک خودرو تخصیص می یابد و تعداد سازندگان/خودروسازان که در یک زمان معقول (۹ ساعته از ۸ تا ۱۷) توسط یک خودرو ملاقات می شوند، اندک است و می توان مدل ریاضی معرفی شده را توسط نرم افزار CPLEX حل کرد. لازم است توجه شود که مدل ریاضی ۴۵-۶۴، باید به طور متوالی و به دفعات زیاد در فرآیند ایجاد جواب استفاده شود. این امر می طلبد که از روشی سریع برای حل آن استفاده شود. به جای استفاده از روش های ابتکاری (که بعضاً خروجی آنها دور از بهینه است) یا روش های فرآیندکاری (که استفاده از آنها زمان بر است)، از گزینه ی حل بهینه توسط نرم افزار استفاده شده است. زیرا بهینگی (در صورت وجود) تضمین شده و زمان اجرا معقول است.

۴.۱.۳. الگوریتم های بهبوددهنده

دو الگوریتم بهبوددهنده روی جواب اولیه اعمال می شود. این دو الگوریتم برای بهبود کیفیت جواب ایجاد شده توسط الگوریتم اولین برآزش استفاده می شوند.

۴.۱.۳.۵. الگوریتم کاهش

بر اساس این پیش فرض که خودروهای کوچک تر هزینه کمتری دارند، الگوریتم کاهش بررسی می کند که آیا مجموعه سفارش های داخل یک خودرو می تواند با یک وسیله ی حمل کوچک تر ارسال شود یا نه. شبه کد الگوریتم کاهش به صورت زیر است:

الگوریتم کاهش

مجموعه Vo را به عنوان مجموعه خودروهای جواب قرار بده؛

برای هر خودروی vo در Vo ؛

مجموعه Vp را به عنوان مجموعه خودروهای کوچک تر از vo قرار بده؛

(که به صورت صعودی اندازه مرتب شده اند)؛

برای هر خودروی vp در Vp ؛

سفارش های vo را در vp قرار بده و شدنی بودن را با حل مدل خلاصه شده، چک کن؛ اگر شدنی بود

سفارش های vo را به صورت دائم در vp قرار بده؛

پایان اگر

پایان برای

پایان برای

۴.۱.۳.۶. الگوریتم ادغام

الگوریتم ادغام، سفارش های داخل دو خودرو را در بزرگترین خودرو از این دو خودرو یا یک خودروی بزرگ تر قرار می دهد. این الگوریتم می تواند به کاهش تعداد خودروها بیانجامد. سازگاری سفارش های دو خودرو قبل از ادغام چک می شود. شبه کد الگوریتم ادغام به صورت زیر است:

الگوریتم ادغام

پارامتر ξ را مقداردهی اولیه کن؛

مجموعه Vo را به عنوان مجموعه خودروهای جواب قرار بده؛

تا زمانی که $\text{counter} \leq \xi$

دو خودروی v_1 و v_2 را از Vo انتخاب کن؛

مجموعه Vp را به عنوان مجموعه خودروهایی که بزرگ تر یا مشابه بزرگترین خودرو بین v_1 و v_2 است، قرار بده (به صورت صعودی)؛

برای هر خودروی vp در Vp

سفارش های داخل v_1 و v_2 را به vp اختصاص بده و شدنی بودن را با حل مدل

جدول ۲. اطلاعات نمونه‌های تصادفی.

شماره‌ی نمونه	پارامترهای مسئله		
	تعداد سفارش‌ها	تعداد سفارش‌های روبه‌جلو	تعداد سفارش‌های برگشتی
۰۱	۶	۴	۲
۰۲	۶	۴	۲
۰۳	۶	۳	۳
۰۴	۶	۳	۳
۰۵	۷	۴	۳
۰۶	۸	۵	۳
۰۷	۱۰	۶	۴
۰۸	۱۱	۷	۴
۰۹	۱۲	۸	۴
۱۰	۱۴	۹	۵
۱۱	۱۸	۱۰	۸
۱۲	۱۹	۱۱	۸
۱۳	۲۲	۱۱	۱۱
۱۴	۳۰	۱۵	۱۵
۱۵	۳۶	۲۰	۱۶
۱۶	۴۲	۲۵	۱۷
۱۷	۵۴	۳۰	۲۴
۱۸	۷۴	۴۰	۳۴
۱۹	۸۳	۴۵	۳۸
۲۰	۹۳	۵۰	۴۳
۲۱	۱۰۳	۵۵	۴۸
۲۲	۱۱۲	۶۰	۵۲
۲۳	۱۲۲	۶۵	۵۷

جدول ۳. داده‌های سفارش‌های یک روز.

...	۶	تعداد پالت	...
...	۷۶۵	ارتفاع	...
...	۱۰۸۰	عرض	...
...	۱۲۰۰	طول	...
...	۱۵۰	وزن پالت خالی (<i>Kgr</i>)	...
...	<i>GP5</i>	نام پالت	...
...	۱۳۲۸۵	شناسه‌ی مقصد	...
...	۱۱۷۵	شناسه‌ی مبدا	...
...	<i>ep</i> ۱	نام سفارش	...
۷۶۵	۷۶۵	ارتفاع	...
۱۰۸۰	۱۰۸۰	عرض	...
۱۲۰۰	۱۲۰۰	طول	...
۴۳۸	۴۳۸	وزن پالت پر (<i>Kgr</i>)	...
۱۵۰	۱۵۰	وزن پالت خالی <i>Kgr</i>	...
۲۸۸	۲۸۸	وزن خالص قطعه در پالت (<i>gr</i>)	...
۱۸۰۰	۱۸۰۰	وزن قطعه (<i>gr</i>)	...
<i>B</i>	<i>B</i>	گرید قطعه	...
۱۶۰	۱۶۰	تعداد قطعه در پالت	...
<i>GP5</i>	<i>GP5</i>	نام پالت	...
۱۱۷۵	۱۱۷۵	شناسه‌ی محل تحویل	...
۱۳۲۸۵	۱۳۲۸۵	شناسه‌ی سازنده	...
۸۰۰	۸۰۰	تعداد پالت	...
<i>DN1204A</i>	<i>DN1205A</i>	شناسه‌ی فنی	...

GAMS/CPLEX مقایسه شده‌اند. مهلت زمانی اجرای GAMS، ۱۰۰۰ ثانیه است و برای نمونه‌هایی که تا این زمان، بهینه‌ی قطعی پیدا نشده است، بهترین جواب یافته شده گزارش شده است.

جدول ۴ نشان‌دهنده‌ی نتایج اجرای مسائل نمونه با GAMS/CPLEX و الگوریتم اولین برازش مبتنی بر شباهت (SBFF) است. تابع هدف (هزینه‌ی کل) و زمان اجرا برای حل هر نمونه ارائه شده است. برای نمونه‌هایی که محدودیت زمانی باعث توقف الگوریتم شده است (با * مشخص شده‌اند)، بهترین جواب یافته شده گزارش شده است. برای نمونه‌های ۱۳ تا ۲۰، هیچ جواب شدنی در مهلت زمانی ۱۰۰۰ ثانیه یافته نشده است (که با † مشخص شده‌اند). همان طور که دیده می‌شود، برای ۵ نمونه‌ی اول که ۶ یا ۷ سفارش دارند، جواب بهینه تضمین شده است. برای نمونه‌های ۶ تا ۱۵، جواب به دست آمده ممکن است بهینه نباشد.

در ستون با عنوان "*Gap (%)*"، الگوریتم SBFF با جواب بهینه یا بهترین جواب یافته شده مقایسه شده است. مقادیر *Gap* با استفاده از رابطه‌ی ۶۵ محاسبه می‌شود.

$$Gap_{obj} = (Obj_{SBFF} - Obj_{opt}) / Obj_{opt} \quad (65)$$

قرار گیرند است؛ به گونه‌ی که هزینه‌ی حمل‌ونقل کمینه شود و تمام شرایط عملیاتی در نظر گرفته شود. این مسئله نیز با الگوریتم ارائه شده حل شده است و نتایج با نتایج واقعی مقایسه شده‌اند.

بر اساس داده‌های جدول ۳، اطلاعاتی مانند نوع پالت (فلزی/چوبی)، سازگاری با دیگر پالت‌ها، تعداد پالتهای که در عرض هر خودرو می‌تواند قرار گیرد، تعداد پالتهای که می‌تواند روی هم در هر خودرو چیده شود و زمان و هزینه‌ی هر خودرو استخراج می‌شود. این مسئله مطالعه‌ی موردی نیز با GAMS/CPLEX حل می‌شود و نتایج حاصل با نتایج الگوریتم مقایسه می‌شود.

۱.۴. نتایج

۱.۴.۱. حل نمونه‌های تصادفی

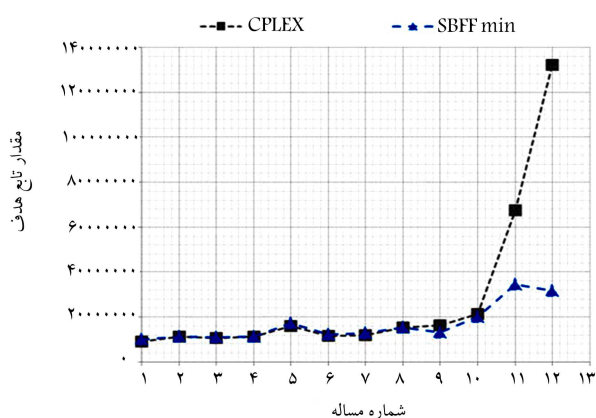
عملکرد الگوریتم ارائه شده با استفاده از نمونه‌های تصادفی ارزیابی شده است. کد الگوریتم اولین برازش مبتنی بر شباهت (SBFF) در نرم‌افزار MATLAB نوشته شده است و با یک رایانه‌ی پنتیوم IV با ۸ گیگابایت حافظه‌ی اصلی و پردازنده‌ی Core i7 (3/60 GHz) اجرا شده است. عملکرد الگوریتم با نتایج حل مسئله با

جدول ۴. نتایج اجرای الگوریتم.

شکاف (%)	SBFF						شماره نمونه
	زمان متوسط (ثانیه)	تابع هدف $\times 10^3$			جواب بهینه/بهترین جواب یافته شده		
		<i>max.</i>	<i>avg.</i>	<i>min.</i>	زمان (ثانیه)	تابع هدف ($\times 10^3$)	
۰٫۰۰	۱۴٫۳۵	۷۷۰۰	۷۵۷۵	۷۳۸۸	۹٫۳	۷۳۸۸	۰۱
۰٫۰۰	۴٫۲	۷۹۰۰	۷۹۰۰	۷۹۰۰	۷٫۶	۷۹۰۰	۰۲
۰٫۰۰	۴٫۱۵	۱۰۸۰	۹۹۲۰	۸۶۰۰	۲٫۷	۸۶۰۰	۰۳
۷٫۸۳	۲۱٫۹۲	۱۹۹۲۰	۱۲۸۴۰	۹۹۲۰	۷٫۴۷	۹۲۰۰	۰۴
۰٫۰۰	۲۶٫۵۱	۲۰۱۷۴	۱۵۶۴۸٫۶	۱۱۴۰۰	۹۸۱٫۱۱	۱۱۴۰۰	۰۵
۱٫۴۷	۳۸٫۱۲	۱۶۰۹۲	۱۲۸۰۲٫۶	۱۰۸۰۰	۱۰۰۴/۷۱*	۱۰۶۴۴	۰۶
۰٫۰۰	۴۳٫۹۵	۲۷۷۱۸	۱۹۲۵۳٫۲	۱۱۴۰۰	۱۰۰۳/۹۱*	۱۱۴۰۰	۰۷
۷٫۶۵	۴۸٫۶۳	۲۵۵۸۸	۲۲۳۰۴٫۲	۱۷۳۱۶	۱۰۱۸/۳۳*	۱۶۰۸۵	۰۸
۶٫۰۹	۴۹٫۸۵	۲۳۸۲۸	۲۰۷۸۹٫۶	۱۲۲۰۰	۱۰۰۵/۸۷*	۱۱۵۰۰	۰۹
۷٫۸۵	۵۲٫۱۲	۲۷۵۴۸	۱۷۵۳۲٫۹	۱۲۹۷۶	۱۰۰۳/۶۹*	۱۲۰۳۲	۱۰
-۱٫۲۰	۵۹٫۸۸	۳۵۴۹۶	۲۲۲۶۹٫۵	۱۵۱۸۸	۱۰۰۶/۳۱*	۱۵۳۷۲	۱۱
-۱۸٫۸۶	۶۶٫۰۱	۲۸۲۰۲	۲۱۴۵۲٫۵	۱۳۲۱۶	۱۰۰۴/۵۷*	۱۶۲۸۸	۱۲
-۴٫۷۲	۷۵٫۲۶	۳۵۸۳۸	۲۷۸۱۹٫۶	۲۰۲۰۰	۱۰۰۵/۲۰*	۲۱۲۰۰	۱۳
-۴۸٫۷۴	۱۰۸٫۹۷	۵۰۰۴۶	۴۲۰۵۱٫۷	۳۴۶۰۴	۱۰۰۷/۸۴*	۶۷۵۰۲	۱۴
-۷۶٫۰۶	۱۵۴٫۹۹	۵۱۴۸۸	۴۰۳۹۶	۳۱۶۲۸	۱۰۰۹/۸۰*	۱۳۲۱۲۲	۱۵
-	۱۸۳	۷۰۹۰۵	۵۱۴۰۱٫۳	۴۰۲۲۸	۱۰۰۳/۵۸*	†	۱۶
-	۲۳۱٫۱	۶۹۹۰۶	۶۰۲۲۶٫۷	۵۴۲۳۶	۱۰۰۷/۷۱*	†	۱۷
-	۴۷۸٫۳۶	۹۹۳۰۰	۸۵۷۵۱٫۹	۷۱۷۸۲	۱۰۰۴/۲۶*	†	۱۸
-	۵۶۸٫۰۴	۹۰۵۲۲	۷۷۸۷۵٫۸	۷۲۳۱۵	۱۰۰۳/۶۵*	†	۱۹
-	۶۶۱٫۶۵	۱۰۵۹۸۶	۹۷۷۱۸٫۹	۸۱۱۶۶	۱۰۰۶/۲۴*	†	۲۰
-	۸۰۸٫۳	۱۱۱۷۲۶	۱۰۱۹۲۸٫۶	۹۳۵۵۲	۱۰۰۲/۵۸*	†	۲۱
-	۹۳۸٫۱۶	۱۳۱۵۴۳	۱۲۱۰۷۳٫۸	۱۱۱۷۹۱	۱۰۰۳/۲۸*	†	۲۲
-	۱۱۷۰٫۹۹	۱۴۱۴۳۹	۱۲۶۱۶۰٫۷	۱۱۲۳۸۰	۱۰۰۶/۹۶*	†	۲۳

* مهلت زمانی اجرا به اتمام رسیده است.

† جواب عدد صحیح یافت نشد.



شکل ۵. تابع هدف به دست آمده توسط CPLEX و SBFF.

که در آن مقدار تابع هدف SBFF است و Obj_{OPT} مقدار تابع هدف جواب بهینه یا بهترین جواب یافته شده است. مقدار منفی این شاخص نشان می‌دهد که الگوریتم ارائه شده جوابی بهتر از بهترین جواب یافته شده توسط GAMS/CPLEX ارائه داده است.

با توجه به مقادیر شکاف در جدول ۴، در ۵ مورد از ۲۳ مورد این مقدار مثبت بوده است، که نشان می‌دهد حل دقیق جواب بهتری نسبت به الگوریتم ارائه داده است؛ اما در ۵ مورد از ۲۳ مورد (۱۰٪) الگوریتم توانسته است تابع هدف مشابه حل دقیق به دست آورد. در حالی که در ۵ مورد از ۲۳ نمونه مسئله مقدار Gap منفی است که نشان‌دهنده عملکرد بهتر الگوریتم SBFF است. این دو حالت که الگوریتم مشابه یا بهتر از حل دقیق عمل کرده است با زیرخط نشان داده شده است. مقادیر تابع هدف در نمونه مسائل مختلف که توسط CPLEX و SBFF حل شده‌اند، در شکل ۵ نشان داده شده‌اند. این شکل نشان می‌دهد که اختلاف این

جدول ۵. نتایج حل مسئله‌ی مطالعه‌ی موردی.

متوسط زمان (ثانیه)	تابع هدف		کمینه	تابع هدف زمان (ثانیه)	†
	بیشینه	میانگین			
۳۵۴/۳۹	۸۹۳۶۹۰۰۰	۷۶۲۳۷۰۰۰	۶۸۲۹۴۰۰۰	۱۰۰۹/۸۹*	

دنیای واقعی توزیع پالت قطعات از سازندگان به سایت‌های خودروسازی و برگرداندن پالت‌های خالی از سایت‌های خودروسازی به سازندگان است. یک مدل ریاضی عدد صحیح مختلط برای این مسئله ارائه شده است که مسیریابی یک ناوگان از خودروهای ناهمگن را برای حمل‌ونقل روبه‌جلو پالت‌های پر و حمل‌ونقل برگشتی پالت‌های خالی را انجام می‌دهد و محدودیت‌های پنجره‌ی زمانی، ظرفیت حجمی و وزنی و سازگاری پالت‌ها را در نظر می‌گیرد.

به منظور حل مسائل دنیای واقعی که به صورت کارا با استفاده از مدل ریاضی قابل‌حل نیستند، یک الگوریتم ابتکاری ارائه شده است. این الگوریتم حاوی یک الگوریتم انطباق‌گراف دوبخشی بایشینه‌ی وزن، یک الگوریتم اولین برآزش و دو الگوریتم بهبوددهنده است. عملکرد این الگوریتم روی مسائل کوچک با نتایج بهینه‌ی به دست آمده از حل مدل ریاضی مقایسه شده است. نشان داده شده است که الگوریتم اولین برآزش ارائه شده جواب‌های مناسبی ارائه می‌کند. سپس این الگوریتم برای حل یک مسئله‌ی مطالعه موردی استفاده شده است که نشان می‌دهد این الگوریتم می‌تواند جواب‌های مناسبی را در زمان مناسب برای مسئله‌ی پژوهش ارائه کند.

برای پژوهش‌های آتی، می‌توان الگوریتم‌های فراابتکاری را برای مسئله طراحی و ارائه کرد. توابع هدف دیگر مانند مجموع میزان آلاینده‌ی خودروهای مختلف را در نظر گرفت. بارگیری سه‌بعدی نیز از شرایطی است که می‌تواند در مدل در نظر گرفته شود.

دو روش حل در مسائل ۱۴ و ۱۵ بسیار زیاد است و از نمونه‌های ۱۶ تا ۲۳، CPLEX نتوانسته است جواب شدنی عدد صحیح در مهلت زمانی مشخص شده ارائه دهد.

عملکرد کلی SBFF در ۲۳ نمونه تصادفی ایجاد شده بر حسب کمینه، متوسط و بیشینه مقدار تابع هدف در ۱۰ بار اجرای الگوریتم برای هر نمونه نشان‌گر آن است که به دلیل اختلاف کم این سه مقدار در بیشتر نمونه مسائل، این الگوریتم عملکرد نسبتاً یکنواخت و پایداری را نشان می‌دهد.

۲.۱.۴. حل مسئله‌ی مطالعه‌ی موردی

مسئله‌ی مطالعه‌ی موردی معرفی شده توسط SBFF و GAMS/CPLEX حل شده است و نتایج در جدول ۵ ارائه شده‌اند. برای این مسئله، GAMS/CPLEX نتوانسته است یک جواب عدد صحیح در مهلت زمانی مشخص پیدا کند و مهلت زمانی علت توقف الگوریتم است († در جدول ۵). در مقابل، الگوریتم SBFF ارائه شده به صورت متوسط نتوانسته است در حدود ۳۵ ثانیه، جواب شدنی مناسبی ارائه کند.

۵. نتیجه‌گیری و پیشنهادها

در این پژوهش، یک سامانه‌ی توزیع مطالعه شده است که بر اساس یک مسئله‌ی

پانوشته‌ها

1. pickup and delivery problem
2. milk run
3. third-party logistics
4. mixed integer programming
5. first-fit
6. similarity-based ordering algorithm
7. similarity-based first-Fit algorithm

منابع (References)

1. Berbeglia, G., Cordeau, J.-F. Gribkovskaia, I. and et al. "Static pickup and delivery problems: a classifica-

tion scheme and survey", *Top*, **15**(1), pp. 1-31 (2007). <https://doi.org/10.1007/s11750-007-0009-0>

2. Ropke, S. and Cordeau, J.-F. "Branch and cut and price for the pickup and delivery problem with time windows", *Transportation Science*, **43**(3), pp. 267-286 (2009). <https://doi.org/10.1287/trsc.1090.0272>
3. Ropke, S., Cordeau, J. F. and Laporte, G. "Models and branch-and-cut algorithms for pickup and delivery problems with time windows", *Networks*, **49**(4), pp. 258-272 (2007).
4. Benavent, E., Landete, M., Mota, E. and et al. "The multiple vehicle pickup and delivery problem with LIFO constraints", *European Journal of Operational Research*, **243**(3), pp. 752-762 (2015). <https://doi.org/10.1016/j.ejor.2014.12.029>

5. Veenstra, M., Cherkesly, M., Desaulniers, G. and et al. "The pickup and delivery problem with time windows and handling operations", *Computers and Operations Research*, **77**, pp. 127-140 (2017). <https://doi.org/10.1016/j.cor.2016.07.014>
6. Cherkesly, M., Desaulniers, G., Irnich, S. and et al. "Branch-price-and-cut algorithms for the pickup and delivery problem with time windows and multiple stacks", *European Journal of Operational Research*, **250**(3), pp.782-793 (2015). <https://doi.org/10.1016/j.ejor.2015.10.046>
7. Li, H. and Lim, A. "A metaheuristic for the pickup and delivery problem with time windows", *International Journal on Artificial Intelligence Tools*, **12**(2), pp. 173-186 (2003). Retrieved from <http://www.worldscientific.com/doi/abs/10.1142/S0218213003001186>
8. Wang, H. F. and Chen, Y. Y. "A genetic algorithm for the simultaneous delivery and pickup problems with time window", *Computers and Industrial Engineering*, **62**(1), pp. 84-95 (2012). <https://doi.org/10.1016/j.cie.2011.08.018>
9. Hosny, M. I. and Mumford, C. L., "The single vehicle PDPTW: Intelligent operators for heuristic and metaheuristic algorithms", *Journal of Heuristics*, **16**(3), pp. 417-439 (2010). <https://doi.org/10.1007/s10732-008-9083-1>
10. Belmecheri, F., Prins, C., Yalaoui, F. and et al. "Particle swarm optimization algorithm for a vehicle routing problem with heterogeneous fleet, mixed backhauls, and time windows", *Journal of Intelligent Manufacturing*, **24**(4), pp. 775-789 (2013). <https://doi.org/10.1007/s10845-012-0627-8>
11. Cherkesly, M., Desaulniers, G. and Laporte, G. "A population-based metaheuristic for the pickup and delivery problem with time windows and LIFO loading", *Computers and Operations Research*, **62**, pp. 23-35 (2015). <https://doi.org/10.1016/j.cor.2015.04.002>
12. Mahmoudi, M. and Zhou, X. "Finding optimal solutions for vehicle routing problem with pickup and delivery services with time windows: A dynamic programming approach based on state-space-time network representations", *Transportation Research Part B: Methodological*, **89**, pp. 19-42 (2016). <https://doi.org/10.1016/j.trb.2016.03.009>
13. Abbasi-Pooya, A. and Husseinzadeh Kashan, A. "New mathematical models and a hybrid grouping evolution strategy algorithm for optimal helicopter routing and crew pickup and delivery", *Computers and Industrial Engineering*, **112**, pp. 35-56 (2017). <https://doi.org/10.1016/j.cie.2017.08.007>
14. Bettinelli, A., Chacchiani, V., Crainic, T. G. and et al. "A branch-and-Cut-and-Price algorithm for the multi-trip separate pickup and delivery problem with time windows and customers and facilities", *European Journal of Operational Research*, **279**(3), pp.824-839 (2019).<http://DOI:10.1015/j.ejor.2019.06.32>.
15. Reil, S., Bortfeldt, A. and Mönch, L. Heuristics for vehicle routing problems with backhauls, time windows, and 3D loading constraints", *European Journal of Operational Research*, **266**(3), pp. 877-894 (2018). <https://doi.org/10.1016/j.ejor.2017.10.029>
16. Nemoto, T., Hayashi, K. and Hashimoto, M., "Milk-run logistics by japanese automobile manufacturers in thailand", *Procedia - Social and Behavioral Sciences*, **2**(3), pp. 5980-5989 (2010). <https://doi.org/10.1016/j.sbspro.2010.04.012>
17. Gyulai, D., Pfeiffer, A., Sobottka, T. and et al. "Milkrun vehicle routing approach for shop-floor logistics", *Procedia CIRP*, **7**, pp. 127-132 (2013).
18. Chuah, K. H. and Yingling, J. C., "Routing for a just-in-time supply pickup and delivery system", *Transportation Science*, **39**(3), pp. 328-339 (2005). <https://doi.org/10.1287/trsc.1040.0092>
19. Du, T., Wang, F. K. and Lu, P. Y. "A real-time vehicle-dispatching system for consolidating milk runs", *Transportation Research Part E: Logistics and Transportation Review*, **43**(5), pp. 565-577 (2007). <https://doi.org/10.1016/j.tre.2006.03.001>
20. Sadjadi, S. J., Jafari, M. and Amini, T. "A new mathematical modeling and a genetic algorithm search for milk run problem (an auto industry supply chain case study)", *International Journal of Advanced Manufacturing Technology*, **44**(1-2), pp. 194-200 (2009). <https://doi.org/10.1007/s00170-008-1648-5>
21. Jafari-Eskandari, M., Sadjadi, S. J., Jabalameli, M. S. and et al. "A robust optimization approach for the Milk Run problem (An auto industry Supply Chain Case Study)", *International Conference On Computers And Industrial Engineering*, CIE 2009, pp. 1076-1081 (2009). <https://doi.org/10.1109/ICCIE.2009.5223541>
22. Nemoto, T. and Rothengatter, W. "Efficient Green Logistics in Urban Areas: Milk Run Logistics in the Automotive Industry", *In Sustainable Transport for Chinese Cities*, **3**, pp. 319-337 (2012). [https://doi.org/10.1108/S2044-9941\(2012\)0000003017](https://doi.org/10.1108/S2044-9941(2012)0000003017)
23. Hosseini, S. D., Shirazi, M. A. and Ghomi, S. M. T. F. "Harmony search optimization algorithm for a novel transportation problem in a consolidation network", *Engineering Optimization*, **46**(11), pp. 1538-1552 (2014). <https://doi.org/10.1080/0305215X.2013.854350>
24. Boysen, N., Emde, S., Hoeck, M. and et al. "Part logistics in the automotive industry: Decision problems, literature review and research agenda", *European Journal of Operational Research*, **242**(1), pp. 107-120 (2015). <https://doi.org/10.1016/j.ejor.2014.09.065>
25. Ai, T. J. and Kachitvichyanukul, V. "A particle swarm optimization for the vehicle routing problem with simultaneous pickup and delivery", *Computers and Operations Research*, **36**(5), pp. 1693-1702 (2009). <https://doi.org/10.1016/j.cor.2008.04.003>
26. Goksal, F. P., Karaoglan, I. and Altiparmak, F. "A hybrid discrete particle swarm optimization for vehicle routing problem with simultaneous pickup and delivery", *Computers and Industrial Engineering*, **65**(1), pp. 39-53 (2013). <https://doi.org/10.1016/j.cie.2012.01.005>
27. Yu, V. F. and Lin, S., (2014) "Multi-start simulated annealing heuristic for the location routing problem with simultaneous pickup and delivery", *Applied Soft Computing Journal*, **24**, pp. 284-290 (2013). <https://doi.org/10.1016/j.asoc.2014.06.024>

28. Chen, Q., Li, K. and Liu, Z. "Model and algorithm for an unpaired pickup and delivery vehicle routing problem with split loads", *Transportation Research Part E: Logistics and Transportation Review*, bf 69, pp. 218-235 (2014). <https://doi.org/10.1016/j.tre.2014.06.010>
29. Ting, C.-K., Liao, X.-L., Huang, Y.-H. and et al. "Multi-vehicle selective pickup and delivery using metaheuristic algorithms", *Information Sciences*, **406-407**, pp. 146-169 (2017). <https://doi.org/https://doi.org/10.1016/j.ins.2017.04.001>
30. Soleimani, H., Chaharlang, Y. and Ghaderi, H., "Collection and distribution of returned-remanufactured products in a vehicle routing problem with pickup and delivery considering sustainable and green criteria", *Journal of Cleaner Production*, **172**, pp. 960-970 (2018). <https://doi.org/10.1016/J.JCLEPRO.2017.10.124>.