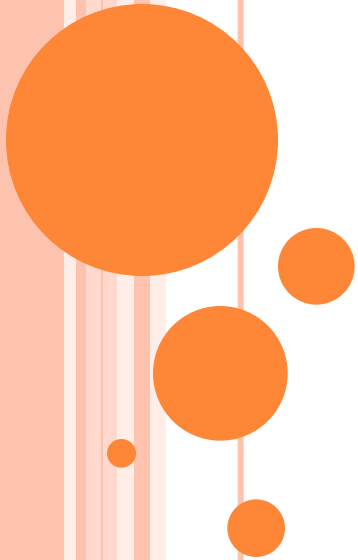


مسئله مسیریابی موجودی غیر قطعی برای جمع آوری زباله‌های پزشکی واگیردار

ارائه دهنده: توحید آریافر
استاد درس: دکتر ستاک



فهرست مطالب

(۱) مقدمه و بیان مفاهیم کلی

(۲) کاربردهای مسئله

(۳) مرور ادبیات

(۴) مدل ریاضی و شرح آن

(۵) روشهای حل استفاده شده

(۶) نتایج عددی و کاربردی به دست آمده

(۷) ارائه پیشنهادهایی برای توسعه مسئله

مقدمه و بیان مفاهیم کلی

➤ امحاء نا امن زباله های مراقبتهای پزشکی ریسکهای زیادی را برای سلامت عمومی در پی دارد و چنانچه جمع آوری و دفن آنها با رعایت اصول مرتبط انجام نشود می تواند تمامی افراد مرتبط از جمله کارکنان بخش سلامت، حمل کنندگان زباله و حتی عابرین پیاده را در معرض خطر قرار دهد. به همین دلیل سازمان بهداشت جهانی خطوط راهنمای صریحی را برای مدیریت زباله های واگیردار ترسیم کرده است.

➤ زباله های پزشکی که در این پروژه مد نظر قرار گرفته اند زباله های ناشی از مراقبتهای پزشکی در خانه و بیماران خود درمان است. این زباله ها عمدتاً شامل اجسام تیزی مانند سوزنها و سرنگها هستند که باید با مراقبت بسیار جمع آوری شوند تا از گسترش آلودگی جلوگیری شود.

➤ در روشی که اکنون انجام می پذیرد بیماران جعبه های خالی را از داروخانه ها دریافت کرده و پس از پر شدن به داروخانه بر می گردانند و مسئول داروخانه آنها را درون وسایل حمل (container) قرار داده و پس از پر شدن وسیله حمل تا حد معینی با مقامات محلی تماس برقرار می کند تا جعبه ها را تحویل بگیرند. جعبه های پر شده توسط یک وسیله نقلیه جمع آوری شده و در مراکز جمع آوری به منظور امحاء انبارش می شوند.

مقدمه و بیان مفاهیم کلی



(a) Medical waste box



(b) Medical waste container



مقدمه و بیان مفاهیم کلی

➤ روش فعلی سه مشکل عمده دارد:

۱- هزینه های حمل بی جهت زیاد است.

۲- با تکمیل ظرفیت وسیله حمل و ورود جعبه های جدید، برای بیمار و داروخانه مشکلات عدیده بوجود می آید.

۳- برای مسئول جمع آوری زباله تصویر واضحی از وظایف آینده اش وجود ندارد.

در این پروژه برای حل این سه مشکل، از قرار دادن تگهای RFID در جعبه ها استفاده می شود. با قراردادن این تگها هم تعداد دقیق جعبه های پر شده در هر زمان و هم آماره نرخ ورود جعبه ها در دسترس است.

بر مبنای تعداد دقیق جعبه ها و آماره آنها و همچنین فاصله بین داروخانه ها سیاستی را تحت عنوان سناریو RFID تعریف می کنیم.

➤ در این پروژه چهار شاخص عملکردی حداقل سازی ریسک مربوط به سلامت عمومی، رضایتمندی داروخانه داران، رضایتمندی مقام محلی و هزینه های مالی تورهای جمع آوری در نظر گرفته شده است.

کاربردهای مسئله

این مقاله بخشی از پروژه ای به نام Trace-de-TIC است که از طرف آژانس مدیریت انرژی و محیط زیست فرانسه حمایت شده است. همچنین این پژوهش با پروژه PAC ID DASRI که توسط منطقه PACA و اتحادیه اروپا حمایت مالی شده، در ارتباط است.

هدف پروژه DASRI بهبود قابلیت ردیابی زباله های واگیردار از طریق توسعه وسایل حمل مکانیزه شده بوسیله تکنولوژی RFID است. توسط این وسایل حمل که در داروخانه ها استقرار یافته اند، جعبه های حامل زباله های واگیردار تا مرحله نهایی عملیات امحاء، قابل شناسایی و ردیابی هستند.

در پروژه Trace-de-TIC و در این مقاله، این موضوع که چگونه اطلاعات جدید فراهم شده بوسیله RFID می تواند در برنامه ریزی عملیات های امحاء کمک کننده باشد را مورد بررسی قرار می دهیم.

مرور ادبیات

۱- مسائل مسیریابی موجودی

Bertazzi و همکاران - ۲۰۰۸

Federgruen و Simchi-Levi - ۱۹۹۵

۲- مسائل جمع آوری زباله دنیای واقعی

Baptista و همکاران - ۲۰۰۲

Nourtio و همکاران - ۲۰۰۶

۳- مسائل مسیریابی وسائل نقلیه غیر قطعی

Laporte و Louveaux - ۱۹۹۸

Tricoire و همکاران - ۲۰۱۲

مدل ریاضی و شرح آن

این مقاله در مورد یک مسأله SCMRP است.

دو تصمیمی که باید در این مسأله اخذ شود:

۱- زمان بازدید از هر داروخانه ۲- چگونگی ترتیب مسیرهای وسیله نقلیه
○ فرضیات مسأله:

- ✓ برای مسیرهای جمع آوری یک وسیله نقلیه در نظر گرفته می شود.
- ✓ وسیله نقلیه در دپو مستقر است و در پایان مسیر نیز در همانجا مستقر می شود.
- ✓ در هر روز حداکثر یک سفر مجاز است.
- ✓ ظرفیت وسیله نقلیه در نظر گرفته نشده است.
- ✓ ورود جعبه های جدید (پر شده از زباله) در داروخانه با استفاده از متغیرهای تصادفی مدل شده است که وابسته به داروخانه و دوره زمانی است.
- ✓ سطح موجودی اولیه برای هر داروخانه دقیقاً مشخص است.
- ✓ هیچ محدودیتی به سطح موجودی پایان دوره برنامه ریزی تحمیل نشده است.
- ✓ تعیین روزهای بازدید و تورهای جمع آوری فقط یکبار و آن هم در آغاز افق برنامه ریزی انجام می شود.

❖ تابع هدف ترکیبی از دو نوع هزینه توزیع و موجودی است.

(1) هزینه های توزیع شامل هزینه های مسیریابی و هزینه های ثابتی است که هر بار یک تور برنامه ریزی می شود به وجود می آید.

(2) هزینه های موجودی هم مرتبط با تعداد جعبه هایی است که در بازدید از هر داروخانه برداشته شده است.

برای تعداد جعبه های درون وسیله حمل هر داروخانه در هر بازدید یک سطح حداقل و حداکثر تعریف شده و با اندیسهای به ترتیب $thrd$ و max مشخص می شوند که چنانچه تعداد جعبه ها از حداقل کمتر و از حداکثر بیشتر باشد در تابع هدف جریمه در نظر گرفته می شود.

اهمیت جعبه های مازاد از ظرفیت یک وسیله حمل فقط به تعداد آنهاست نه تعداد دوره هایی که در انتظار جمع آوری می مانند.

مدل ریاضی و شرح آن

این مسئله می تواند به عنوان یک مسئله بهینه سازی غیرقطعی دو مرحله ای با ارجاع فرموله شود.

در مرحله اول تصمیمات باید روی روزهای بازدید از هر داروخانه و تورهایی که باید توسط وسیله نقلیه انجام گیرد، اتخاذ شود.

در مرحله دوم تعداد جعبه هایی که باید از هر داروخانه جمع آوری گردد تعیین شده است.

تشریح عناصر تشکیل دهنده مدل:

$G=(V,A)$: گراف کامل

$\{1, \dots, T\}$: مجموعه ای از دوره های زمانی منفرد

$V=\{0, 1, \dots, n\}$: مجموعه گره ها

(گره 0 برای دپو در نظر گرفته شده است و گره های 1 تا n داروخانه ها رو مشخص می کند)

A: مجموعه کمان

مدل ریاضی و شرح آن

پارامترها:

- n : تعداد داروخانه ها
- T : تعداد دوره های زمانی
- ξ_i^t : تعداد جعبه هایی که در زمان t در داروخانه i قرار گرفته اند
(متغیر تصادفی با توزیع پواسون و میانگین λ_{it})
- I_i^0 : موجودی اولیه در داروخانه i
- I_i^{max} : حداکثر موجودی پذیرفته شده در داروخانه i
- I_i^{thrd} : سطح موجودی آستانه برای داروخانه i
- C_{ij} : هزینه سفر بین مکانهای i و j
- C_f : هزینه ثابت وسیله نقلیه
- C_i^{thrd} : هزینه جریمه برای داروخانه i (به علت نرسیدن سطح موجودی به آستانه)
- C_i^{max} : هزینه موجودی اضافی برای داروخانه i

مدل ریاضی و شرح آن

متغیرها:

w_i^t : برابر با یک اگر داروخانه i در زمان t بازدید شده است، در غیر اینصورت برابر صفر

z_i^t : تعداد جعبه های برداشته شده از داروخانه i در زمان t

x_{ij}^t : برابر با یک اگر داروخانه j بلافاصله پس از داروخانه i در زمان t بازدید شده است، در غیر اینصورت برابر صفر

I_i^t : موجودی در داروخانه i در زمان t
(بعد از ورود جعبه های جدید و قبل از جمع آوری جعبه های ذخیره شده)

v_i^t : تعداد جعبه های بیشتر از I_i^{max} در زمان t

y_i^t : برابر با یک اگر در هنگام بازدید از داروخانه i سطح موجودی I_i^{thrd} رد نشده باشد، در غیر اینصورت برابر صفر

مرحله اول

$$\min \sum_{i=0}^n \sum_{j=0}^n \sum_{t=1}^T c_{ij} x_{ij}^t + \sum_{j=1}^n \sum_{t=1}^T c_f x_{0j}^t + \mathbf{E}(Q(x, w, \xi)) \quad (1) \quad \text{تابع هدف:}$$

هزینه سفر

هزینه ثابت

هزینه موجودی انتظاری

هزینه موجودی انتظاری ناشی از اختلاف بین تعداد جعبه های برآورد شده از تعداد حقیقی آنها در هنگام بازدید است. یا به عبارت دیگر هزینه تصمیمات اخذ شده پس از تشخیص ξ

تابع ارجاع: $Q(x, w, \xi)$

مدل ریاضی و شرح آن

محدودیتها

$$\sum_{j=1}^n x_{0j}^t \leq 1 \quad \forall t = 1, \dots, T \quad (2)$$

محدودیت ۲: بیان می کند که حداکثر یک تورمی تواند در هر دوره زمانی مجزا انجام شود.

$$\sum_{i=0, i \neq j}^n x_{ij}^t = w_j^t \quad \forall j = 1, \dots, n; t = 1, \dots, T \quad (3)$$

$$\sum_{j=0, j \neq i}^n x_{ij}^t = w_i^t \quad \forall i = 1, \dots, n; t = 1, \dots, T \quad (4)$$

محدودیتهای ۳ و ۴: تضمین می کنند که یک داروخانه فقط وقتی بازدید شده است که یک کمان به آن داروخانه هدایت شده است. همچنین این دو اطمینان می دهند که اگر وسیله نقلیه در یک دوره زمانی مجزا از داروخانه بازدید کرد، حتماً در همان دوره زمانی داروخانه را ترک می کند.

مدل ریاضی و شرح آن

$$\sum_{i,j \in S} x_{ij}^t \leq |S| - 1 \quad \forall S \subseteq \{1, \dots, n\}; t = 1, \dots, T \quad (5)$$

محدودیت ۵: محدودیت حذف زیر تور است.

$$x_{ij}^t \in \{0, 1\} \quad \forall i = 0, \dots, n; j = 0, \dots, n; t = 1, \dots, T \quad (6)$$

$$w_i^t \in \{0, 1\} \quad \forall i = 1, \dots, n; t = 1, \dots, T \quad (7)$$

محدودیت ۶ و ۷: متغیرهای مرحله اول مسئله را تعریف می کنند.

مدل ریاضی و شرح آن

مرحله دوم

تابع هدف:

$$Q(x, w, \xi) = \min \sum_{i=1}^n \sum_{t=1}^T c_i^{\max} v_i^t + \sum_{i=1}^n \sum_{t=1}^T c_i^{\text{thrd}} y_i^t + \sum_{i=1}^n c_i^{\max} I_i^{T+1} \quad (8)$$

هزینه موجودی مازاد دوره T هزینه جریمه آستانه هزینه موجودی مازاد مجزا

مدل ریاضی و شرح آن

محدودیتها

$$I_i^t = I_i^0 + \sum_{s=1}^t \xi_i^s - \sum_{s=1}^{t-1} z_i^s \quad \forall i = 1, \dots, n; t = 1, \dots, T \quad (9)$$

$$I_i^{T+1} \geq I_i^T - z_i^T - I_i^{\max} \quad \forall i = 1, \dots, n \quad (10)$$

محدودیت‌های ۹ و ۱۰: به ترتیب سطح موجودی داروخانه i در دوره زمانی t و موجودی مازاد در پایان افق برنامه‌ای را نشان می‌دهند.

$$z_i^t \leq I_i^t + M(1 - w_i^t) \quad \forall i = 1, \dots, n; t = 1, \dots, T \quad (11)$$

$$z_i^t \geq I_i^t - M(1 - w_i^t) \quad \forall i = 1, \dots, n; t = 1, \dots, T \quad (12)$$

محدودیت‌های ۱۱ و ۱۲: این دو محدودیت اطمینان می‌دهند که کل موجودی یک داروخانه که مورد بازدید قرار گرفته است، برداشته شده است.

مدل ریاضی و شرح آن

$$z_i^t \leq Mw_i^t \quad \forall i = 1, \dots, n; t = 1, \dots, T \quad (13)$$

محدودیت ۱۳: این محدودیت تأکید می کند که تعداد جعبه های جمع آوری شده هر داروخانه که بازدید نشده است، صفر است.

$$v_i^t \geq z_i^t - I_i^{\max} \quad \forall i = 1, \dots, n; t = 1, \dots, T \quad (14)$$

$$v_i^t \geq 0 \quad \forall i = 1, \dots, n; t = 1, \dots, T \quad (18)$$

محدودیت ۱۴: این محدودیت به همراه محدودیت ۱۸، تعداد (غیر منفی) جعبه هایی که سطح موجودی ماکسیمم مربوط به داروخانه i را رد کرده اند، تعیین می کند.

$$z_i^t \geq I_i^{\text{thrd}}(w_i^t - y_i^t) \quad \forall i = 1, \dots, n; t = 1, \dots, T \quad (15)$$

محدودیت ۱۵: این محدودیت، y_i^t را برابر با یک تعریف می کند اگر تعداد جعبه های برداشته شده از داروخانه i در زمان t کمتر از موجودی آستانه باشد و در غیر این صورت برابر با صفر.

مدل ریاضی و شرح آن

$$I_i^t \geq 0 \quad \forall i = 1, \dots, n; t = 1, \dots, T + 1 \quad (16)$$

$$z_i^t \geq 0 \quad \forall i = 1, \dots, n; t = 1, \dots, T \quad (17)$$

$$v_i^t \geq 0 \quad \forall i = 1, \dots, n; t = 1, \dots, T \quad (18)$$

$$y_i^t \in \{0, 1\} \quad \forall i = 1, \dots, n; t = 1, \dots, T \quad (19)$$

محدودیت‌های ۱۶ تا ۱۹: این محدودیتها، متغیرهای مرحله دوم مسئله را تعریف می کنند.

روشهای حل استفاده شده

به دلیل پیچیدگی مسئله یک روش حل exact را نمی توان برای مسائل دنیای واقعی به ذهن متبادر کرد. به همین دلیل دو روش حل ابتکاری ارائه شده است.

روش اول- روش نمونه گیری:

در این رویکرد توزیع مفروض بردار تصادفی ξ بوسیله یک توزیع نمونه ای تجربی، بدست آمده از طراحی N سناریو ξ^v ($v=1, \dots, N$) از توزیع اصلی ξ ، تخمین زده شده است. سپس توزیع اصلی با یک توزیع مجزا جایگزین شده است. هر یک از N سناریو احتمال یکسان $\frac{1}{N}$ را دارا هستند. حال هزینه انتظاری موجودی مازاد و هزینه انتظاری بازدید از یک داروخانه بوسیله میانگین نمونه ای تخمین زده شده اند:

$$\frac{1}{N} \sum_{v=1}^N Q(x, w, \xi^{(v)}) \quad (20)$$

با تعریف $I_{ij}^{(v)}$ ، $Z_{ij}^{(v)}$ ، $v_{ij}^{(v)}$ و $y_{ij}^{(v)}$ به عنوان نمونه ای از متغیرهای مرحله دوم که در ارتباط با سناریو v است، می توانیم مدل مرحله اول ($v-1$) و مدل مرحله دوم ($19-8$) را در قالب یک برنامه ریزی خطی عدد صحیح مرکب بیاوریم.

تابع هدف:

$$\min \sum_{i=0}^n \sum_{j=0}^n \sum_{t=1}^T c_{ij} x_{ij}^t + \sum_{j=1}^n \sum_{t=1}^T c_f x_{0j}^t + \frac{1}{N} \sum_{v=1}^N \left(\sum_{i=1}^n \sum_{t=1}^T c_i^{\max} v_{it}^{(v)} + \sum_{i=1}^n \sum_{t=1}^T c_i^{\text{thrd}} y_{it}^{(v)} + \sum_{i=1}^n c_i^{\max} I_{i,T+1}^{(v)} \right) \quad (21)$$

روشهای حل استفاده شده

محدودیتها:

$$\sum_{j=1}^n x_{0j}^t \leq 1 \quad \forall t = 1, \dots, T \quad (22)$$

$$\sum_{i=0, i \neq j}^n x_{ij}^t = w_j^t \quad \forall j = 1, \dots, n; t = 1, \dots, T \quad (23)$$

$$\sum_{j=0, j \neq i}^n x_{ij}^t = w_i^t \quad \forall i = 1, \dots, n; t = 1, \dots, T \quad (24)$$

$$\sum_{i,j \in S} x_{ij}^t \leq |S| - 1 \quad \forall S \subseteq \{1, \dots, n\}; t = 1, \dots, T \quad (25)$$

$$I_{it}^{(v)} = I_{i0}^{(v)} + \sum_{s=1}^t \xi_{is}^{(v)} - \sum_{s=1}^{t-1} z_{is}^{(v)}$$

$$\forall i = 1, \dots, n; t = 1, \dots, T; v = 1, \dots, N \quad (26)$$

$$I_{i,T+1}^{(v)} \geq I_{iT}^{(v)} - z_{iT}^{(v)} - I_i^{\max} \quad \forall i = 1, \dots, n; v = 1, \dots, N \quad (27)$$

روشهای حل استفاده شده

محدودیتها:

$$z_{it}^{(v)} \leq I_{it}^{(v)} + M(1 - w_i^t)$$

$$\forall i = 1, \dots, n; t = 1, \dots, T; v = 1, \dots, N \quad (28)$$

$$z_{it}^{(v)} \geq I_{it}^{(v)} - M(1 - w_i^t)$$

$$\forall i = 1, \dots, n; t = 1, \dots, T; v = 1, \dots, N \quad (29)$$

$$z_{it}^{(v)} \leq Mw_i^t \quad \forall i = 1, \dots, n; t = 1, \dots, T; v = 1, \dots, N \quad (30)$$

$$v_{it}^{(v)} \geq z_{it}^{(v)} - I_i^{\max}$$

$$\forall i = 1, \dots, n; t = 1, \dots, T; v = 1, \dots, N \quad (31)$$

$$z_{it}^{(v)} \geq I_i^{\text{thrd}}(w_i^t - y_i^t)$$

$$\forall i = 1, \dots, n; t = 1, \dots, T; v = 1, \dots, N \quad (32)$$

$$x_{ij}^t \in \{0, 1\} \quad \forall i = 0, \dots, n; j = 0, \dots, n; t = 1, \dots, T \quad (33)$$

$$w_i^t \in \{0, 1\} \quad \forall i = 1, \dots, n; t = 1, \dots, T \quad (34)$$

روشهای حل استفاده شده

محدودیتها:

$$I_{it}^{(v)} \geq 0 \quad \forall i = 1, \dots, n; t = 1, \dots, T + 1; v = 1, \dots, N \quad (35)$$

$$z_{it}^{(v)} \geq 0 \quad \forall i = 1, \dots, n; t = 1, \dots, T; v = 1, \dots, N \quad (36)$$

$$v_{it}^{(v)} \geq 0 \quad \forall i = 1, \dots, n; t = 1, \dots, T; v = 1, \dots, N \quad (37)$$

$$y_{it}^{(v)} \in \{0, 1\} \quad \forall i = 1, \dots, n; t = 1, \dots, T; v = 1, \dots, N \quad (38)$$

روشهای حل استفاده شده

روش دوم-روش الگوریتم ALNS:

ALNS یک الگوریتم مبتنی بر جستجوی وسیع همسایگی است، به نحوی که یک پاسخ اولیه از طریق تخریب و تعمیر متناوب، رفته رفته بهبود داده می شود. شبه کد زیر نمای اصلی الگوریتم ALNS ما را نمایش می دهد که در یک فرآیند چند آغاز جاسازی شده است.

Algorithm 1 ALNS algorithm in a multi-start procedure

```
1:  $f(s^*) \leftarrow +\infty$ 
2: repeat
3:   Initialize probabilities for perturbation
4:    $s \leftarrow$  initial solution
5:    $s \leftarrow$  local search( $s$ )
6:   repeat
7:     STOP = false
8:     repeat
9:       Select destroy mechanism
10:       $s' \leftarrow$  perturbation( $s$ ) (destroy mechanism)
11:       $s' \leftarrow$  local search( $s'$ ) (repair mechanism)
12:    until Improvement or fail
13:    if  $f(s') < f(s)$  then
14:       $s \leftarrow s'$ 
15:      Update probabilities for perturbation
16:    else
17:      STOP = true
18:    end if
19:  until STOP
20:  if  $f(s) < f(s^*)$  then
21:     $s^* \leftarrow s$ 
22:  end if
23: until Time limit
```

جواب s^* بهترین جواب پیدا شده در طی اجرای الگوریتم را نشان می دهد. در ابتدا s^* به یک جواب موهوم ارجاع می دهد که مقدار $f(s^*)$ به صورت مصنوعی بر روی مقدار $+\infty$ تنظیم شده است. یک پاسخ تصادفی s ساخته شده است (خط ۴)، سپس با ALNS بهبود پیدا کرده است (خطهای ۵-۱۹). در هر بار بهبود s با s^* مقایسه شده اگر بهتر باشد نگه داشته می شود (خطهای ۲۰-۲۲) فرآیند چند آغاز تکرار می شود تا وقتی که محدودیت زمانی مفروض به پایان برسد.

روشهای حل استفاده شده

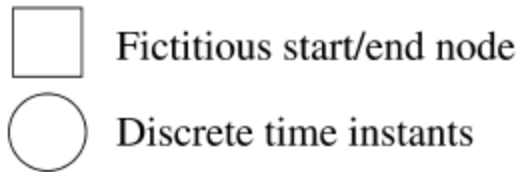
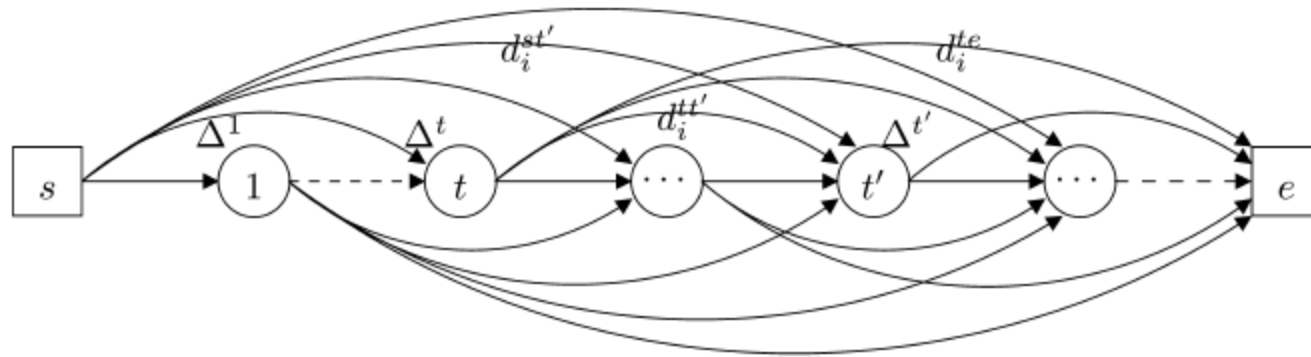
پاسخ های اولیه با تخصیص تصادفی روزهای جمع آوری به داروخانه ها تولید می شود. سپس برای هر روز یک مسئله TPS با روش ابتکاری Lin-Kernighan حل می شود. روش ابتکاری Lin-Kernighan یک عمومی سازی از 2-opt و 3-opt است که با جابجایی جفتی از زیر تورها یک تور جدید خلق می شود.

مکانیزم تخریب و تعمیر: الگوریتم ALNS بر چهار عملگر تخریب متفاوت و یک مکانیسم جستجوی محلی همسایگی وسیع مجزا استوار است.

این چهار عملگر تخریب به صورت چرخشی بر اساس موفقیتشان در طی جستجو انتخاب می شوند. در ابتدا هر چهار مکانیسم تخریب احتمال یکسانی دارند که چنانچه یک عملگر منجر به بهبود پاسخ فعلی شود احتمال آن تا $1/5$ درصد افزایش یافته و احتمال دیگر عملگرها تا $0/5$ درصد کاهش می یابد. هر انحرافی توسط جستجوی محلی تعقیب می شود. پاسخ انحراف یافته s' فقط وقتی که بر جواب s فعلی غلبه کند نگه داشته می شود.

در غیر اینصورت عملگر تخریب دیگری انتخاب می شود. وقتی که هر چهار عملگر تخریب مردود شدند الگوریتم متوقف می شود.

روشهای حل استفاده شده



Shortest path problem for pharmacy i .

برای به دست آوردن بهترین همسایگی (زمانهای بازدید) برای داروخانه i از روش کوتاهترین مسیر به صورت شکل بالا استفاده می شود. هر گره یک بازه زمانی مجزا را مدیریت می کند (گره ابتدا و انتها موهومی بوده و برای کامل شدن گراف ترسیم شده اند). عبور از گره t به این معناست که داروخانه i در زمان t بازدید شده است. هر گره از طریق کمانها به تمام گره های بعد از خودش متصل می شود. مسافت (هزینه ها) برای این مسئله کوتاهترین مسیر از دو نوع هزینه تشکیل شده است:

Δ^t : هزینه مسیریابی تحمیل شده بر اثر بازدید از یک گره (داروخانه) در زمان t

$d_i^{tt'}$: هزینه موجودی بر اثر برنامه ریزی بازدید در روز t' پس از بازدید در روز t از داروخانه i

روشهای حل استفاده شده

عدم قطعیت مسئله SCMIRP در هزینه موجودی لحاظ شده است به این ترتیب که از آنجایی که تعداد جعبه های جمع شده در داروخانه را نمی توان به صورت دقیق مشخص کرد از متغیر تصادفی ξ برای برآورد جعبه ها استفاده می شود. از این رو مقدار هزینه موجودی نیز که خود از دو هزینه موجودی مازاد مجزا و هزینه نرسیدن به سطح موجودی آستانه تشکیل شده است به صورت زیر برآورد می شود:

$$d_i^{tt'} = c_i^{\max} \sum_{I=I_i^{\max}+1}^{\infty} P(I_i^{tt'} = I) \times (I - I_i^{\max}) + c_i^{\text{thrd}} P(I_i^{tt'} \leq I_i^{\text{thrd}} - 1) \quad (39)$$

$I_i^{tt'}$: سطح موجودی داروخانه i در زمان بازدید در روز t' پس از انجام آخرین بازدید در روز t

$$I_i^{tt'} = \sum_{r=t+1}^{t'} \xi_i^r \quad (40)$$

چنانچه بازدید قبلی یک داروخانه گره موهومی آغازین باشد برای محاسبه سطح موجودی فعلی آن رابطه (41) جایگزین معادله (40) می شود.

$$I_i^{st'} = I_i^0 + \sum_{r=1}^{t'} \xi_i^r \quad (41)$$

روشهای حل استفاده شده

چنانچه هزینه موجودی برای گره موهومی پایانی مد نظر باشد فقط هزینه موجودی مازاد در نظر گرفته می شود و رابطه (42) جایگزین رابطه (39) می شود.

$$d_i^{te} = c_i^{\max} \sum_{I=I_i^{\max}+1}^{\infty} P(I_i^{tT} = I) \times (I - I_i^{\max}) \quad (42)$$

کوتاهترین مسیر با به کارگیری الگوریتم دایجسترا حاصل می شود و بهترین الگوی جمع آوری را برای داروخانه i با در نظر گرفتن الگوهای جمع آوری برای دیگر داروخانه ها مهیا می کند. چنانچه الگوی ارائه شده توسط کوتاهترین مسیر با الگوی فعلی متفاوت باشد جواب به روز می شود. در این هنگام همان طور که توضیح داده شد الگوریتم Lin-Kernighan برای بهبود هزینه های مسیریابی برای روزهای تعیین شده به کار می رود. سپس این روال برای داروخانه بعدی مورد استفاده قرار میگیرد.

نتایج عددی و کاربردی به دست آمده

برای انجام محاسبات آزمایشی ۵ نمونه واقعی از منطقه PACA که شامل ۵ داروخانه با افق زمانی ۱۰ روز و دوره زمانی ۱ روز است در نظر گرفته شده است. هزینه ثابت و وسیله نقلیه، هزینه جریمه نرسیدن به سطح آستانه و هزینه هر یک عدد موجودی مازاد، هر کدام برابر ۱۰۰۰ واحد در نظر گرفته شده است. نرخ ورود جعبه های پر شده توزیع پواسون با میانگین λ_i^t فرض شده است. بر آورد هزینه موجودی از طریق معادلات (۳۹) و (۴۰) و به صورت زیر محاسبه می شود:

$$\lambda = \sum_{r=t+1}^r \lambda_i^r \quad (43)$$

$$d_i^{t''} = c_i^{\max} \sum_{I_i^{t''}=I_i^{\max}+1}^{\infty} e^{-\lambda} \frac{\lambda^{I_i^{t''}}}{I_i^{t''}!} (I_i^{t''} - I_i^{\max}) + c_i^{\text{thrd}} \sum_{I_i^{t''}=0}^{I_i^{\text{thrd}}-1} e^{-\lambda} \frac{\lambda^{I_i^{t''}}}{I_i^{t''}!} \quad (44)$$

$$= c_i^{\max} (\lambda - I_i^{\max} + (I_i^{\max} - \lambda) \sum_{I_i^{t''}=0}^{I_i^{\max}-1} e^{-\lambda} \frac{\lambda^{I_i^{t''}}}{I_i^{t''}!}$$

$$+ I_i^{\max} e^{-\lambda} \frac{\lambda^{I_i^{\max}}}{I_i^{\max}!}) + c_i^{\text{thrd}} \sum_{I_i^{t''}=0}^{I_i^{\text{thrd}}-1} e^{-\lambda} \frac{\lambda^{I_i^{t''}}}{I_i^{t''}!}$$

نتایج عددی و کاربردی به دست آمده

برنامه ریزی خطی عدد صحیح مرکب با نرم افزار IBM CPLEX حل شده است.
در جدول زیر اطلاعات قطعی به هر دو روش اختصاص داده شده است.

Computational results for realizations of small instances (deterministic).

	Routing costs	Inventory costs	Total costs	Computing time
CPLEX realizations				
Instance 1	2863.2	2100.0	4963.2	18 min
Instance 2	2983.8	4200.0	7183.8	45 min
Instance 3	2346.9	700.0	3046.9	10 min
Instance 4	2147.1	1100.0	3247.1	7 min
instance 5	2966.9	2700.0	5666.9	27 min
ALNS deterministic				
Instance 1	2863.2	2100.0	4963.2	30 sec
Instance 2	2983.8	4200.0	7183.8	30 sec
Instance 3	2346.9	700.0	3046.9	30 sec
Instance 4	2147.1	1100.0	3247.1	30 sec
Instance 5	2966.9	2700.0	5666.9	30 sec

نتایج عددی و کاربردی به دست آمده

در جدول زیر اطلاعات غیر قطعی به هر دو روش اختصاص داده شده است.

Computational results for small instances (stochastic).

	Routing costs	Inventory costs	Total costs	Computing time	Gap to ALNS (%)
CPLEX average					
Instance 1	2497.5	6365.1	8862.6	5 min	17.3
Instance 2	3687.7	10,061.7	13,749.4	30 min	25.8
Instance 3	1269.6	4200.8	5470.4	15 min	0.0
Instance 4	1188.5	9600.4	10,788.9	2 min	43.5
Instance 5	2455.5	7644.9	10,100.4	35 min	3.6
CPLEX sampling					
Instance 1	3637.2	5344.5	8981.7	32 h	18.8
Instance 2	2551.9	8371.0	10,922.9	46 h	0.0
Instance 3	1269.6	4200.8	5470.4	11 h	0.0
Instance 4	2426.7	5945.6	8372.3	2 h	11.4
Instance 5	2455.5	7644.9	10,100.4	72 h	3.6
ALNS average					
Instance 1	2497.7	5059.7	7557.4	30 sec	
Instance 2	2551.9	8371.0	10,922.9	30 sec	
Instance 3	1269.6	4200.8	5470.4	30 sec	
Instance 4	2427.8	5088.7	7516.5	30 sec	
Instance 5	2455.5	7290.9	9746.4	30 sec	

مشخص است که الگوریتم ALNS در هر دو حالت قطعی و غیر قطعی بهتر است و زمان آن نیز بسیار سریعتر است.

نتایج عددی و کاربردی به دست آمده

در جدول زیر اطلاعات غیر قطعی بوده و به هزینه موجودی و هزینه مسیریابی ضرایب مختلف داده و تأثیر آنها را در زمان محاسبه و میزان فاصله تا مقدار ALNS مقایسه کرده است. چون عدم قطعیت مربوط به هزینه موجودی است، با افزایش ضریب هزینه مسیریابی، زمان محاسبه افزایش می یابد ولی میزان فاصله تا مقدار ALNS کاهش می یابد.

Cost analysis for small instances (stochastic).

Method	Rout. Factor	Inv. Factor	Total costs	Routing costs	Inventory costs	Computing time (sec.)	Gap to ALNS
ALNS average	1	10	4236.01	52,401.68	56,637.66	30	0.00%
CPLEX average	1	10	3351.52	75,385.50	78,737.08	8	44.44%
CPLEX sampling	1	10	4002.17	58,629.92	62,632.10	1749	11.77%
ALNS average	1	5	3804.21	26,467.18	30,271.36	30	0.00%
CPLEX average	1	5	3351.52	37,208.96	40,560.48	9	37.68%
CPLEX sampling	1	5	3787.52	29,722.70	33,510.18	2489	10.83%
ALNS average	5	1	10,014.27	6408.94	16,423.20	30	0.00%
CPLEX average	5	1	9962.86	7352.36	17,315.22	11483	5.71%
CPLEX sampling	5	1	10,049.91	6948.21	16,998.10	15,7452	3.99%
ALNS average	10	1	15,143.84	9400.23	24,544.08	30	0.00%
CPLEX average	10	1	15,017.28	12,844.78	27,862.04	24,384	12.98%
CPLEX sampling	10	1	12,589.62	12,172.13	24,761.76	132,457	0.98%

نتایج عددی و کاربردی به دست آمده

مقایسه ۳ سناریو دیگر با سناریو RFID

Computational results for real-world instances.

	Routing costs	Inventory costs	Total costs
Current scenario 85%			
Instance 1	98,060.0	156,870.0	254,930.0
Instance 2	97,535.4	177,960.0	275,495.4
Instance 3	98,779.4	185,810.0	284,589.4
Instance 4	96,575.0	140,610.0	237,185.0
Instance 5	98,475.8	179,700.0	278,175.8
Mean value	97,885.1	168,190.0	266,075.1 ✓
Current scenario threshold			
Instance 1	107,120.4	31,830.0	138,950.4
Instance 2	107,355.0	39,390.0	146,745.0
Instance 3	107,721.4	37,510.0	145,231.4
Instance 4	106,346.7	25,810.0	132,156.7
Instance 5	107,296.3	38,270.0	145,566.3
Mean value	107,167.9	34,562.0	141,729.9 ✓
RFID scenario			
Instance 1	47,868.8	91,700.0	141,148.4
Instance 2	53,114.8	100,980.0	155,121.7
Instance 3	42,700.7	106,420.0	150,839.2
Instance 4	42,630.6	88,860.0	131,879.4
Instance 5	47,939.6	101,190.0	150,764.0
Mean value	46,850.9	97,830.1	144,681.0 ✓
Rolling horizon scenario			
Instance 1	53,383.4	71,650.0	125,033.4
Instance 2	53,069.2	83,540.0	136,609.2
Instance 3	54,102.3	81,920.0	136,022.3
Instance 4	50,074.3	68,310.0	118,384.3
Instance 5	52,191.8	82,080.0	134,271.8
Mean value	52,564.2	77,500.0	130,064.2 ✓

ارائه پیشنهادهایی برای توسعه مسئله

- ۱- در مدل اصلی می توانیم ظرفیت وسیله نقلیه را نیز به عنوان یک محدودیت در نظر بگیریم.
- ۲- می توانیم مسافت طی شده در یک تور را به عنوان یک محدودیت به مسئله اضافه کنیم.
- ۳- می توانیم برای مراکز جمع آوری زباله محدودیت سطح موجودی در نظر بگیریم و توزیع جعبه ها در میان مراکز توزیع را بهینه سازی کنیم.
- ۴- می توانیم برای موجودی پایان افق برنامه ریزی سطح صفر را تعریف کنیم.

با سپاس از توجه شما