Vehicle Routing Problem With Time Window

مسئله مسیریابی وسیله نقلیه با پنجره زمانی

استاد : آقای دکتر ستاک

ارائه دهنده: امیدزارعی

بهار ۹۴

فهرست مطالب

- مقدمه
- انوع پنجره زمانی
- مدلسازی VRPTW با پنجره زمانی سخت
- مدلسازی VRPTWبا پنجره زمانی نرم
- تعیین یک جواب اولیه
- الگوریتم جستجو ممنوع

تعریف مسئله

- مسئله مسیریابی وسایل حمل ونقل با پنجره زمانی (VRPTW) مسئله تعمیم یافته ای از مسئله مسیریابی وسایل حمل ونقل با محدودیت ظرفیت (CVRP) است که در آن سرویس به هر مشتری باید در یک بازه **زمانی معین** (این بازه زمانی به پنجره زمانی معروف است) صورت گیرد.

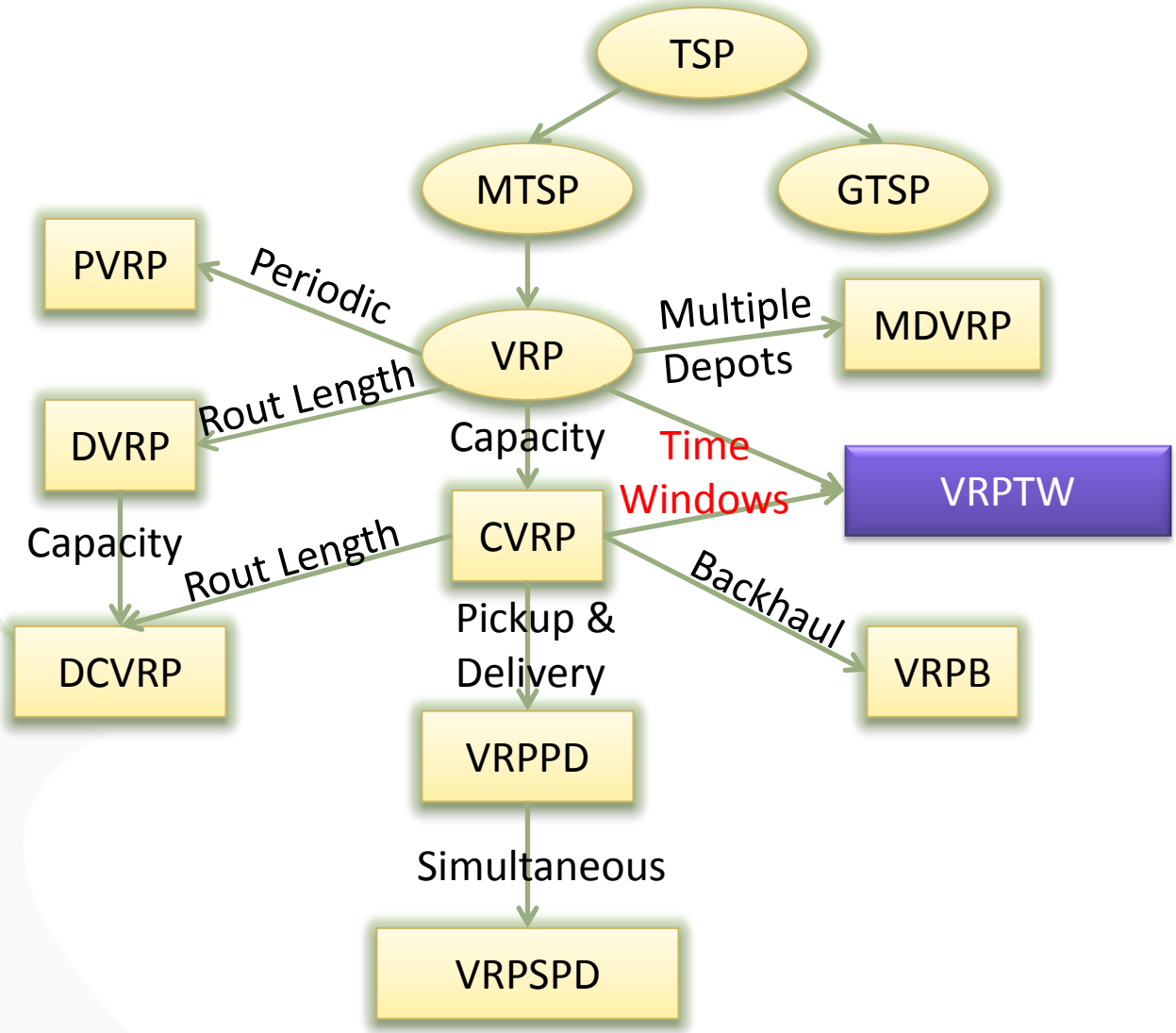
- مسئله مسیریابی وسیله نقلیه با توجه به صورت گرفتن سرویس در یک بازه زمانی خاص پیچیدگی زیادی داشته که در صورت اضافه شدن محدودیت **طول مسیر** و **هزینه پنجره زمانی** در عین پیچیدگی بسیار بالا به یک مبحث کاربردی درعمل بسیار نزدیک می گردد.

- در هر VRPTW با دو فاز **مسیریابی** و **برنامه ریزی** وسایل نقلیه مواجه هستیم. مسیریابی مربوط به پیدا کردن یک مسیر ایده آل است که اهداف مدل را تامین کرده و از مشتری تبعیت کند در حالی که برنامه ریزی وسایل نقلیه ، زمانی که باید به هر مشتری خدمت داده شود را معین می کند.

- سولومون در سال ۱۹۸۲ یکی از مهم‌ترین مشتقات VRP، یعنی ***پنجره زمانی*** را معرفی کرد.

مثال هایی از VRPTW

- تقسیم پول نقد به شعب بانک ها
- جمع آوری زباله ها و ضایعات صنعتی
- تقسیم سوخت به جایگاه های پخش
- سرویس مدارس
- تحویل نامه و روزنامه
- مراکز توزیع
- .....


TSP
MTSP
GTSP
PVRP
Periodic
VRP
Multiple
Depots
MDVRP
DVRP
Rout Length
Capacity
Time
Windows
VRPTW
Capacity
Rout Length
CVRP
Backhaul
Pickup &
Delivery
DCVRP
VRPB
VRPPD
Simultaneous
VRPSPD

ویژگی ها مسئله

- مسئله TSP با پنجره زمانی شکل ساده شده VRPTW می باشد که در آن تعداد وسایل نقلیه یک و مجموع تقاضاها کمتر از ظرفیت می باشد.
- جزء مسائل NP_hard است.
- VRPTW تعمیم یافته مسئله مسیریابی وسیله نقلیه با محدودیت $[e_i , l_i]$ ی است.
- پنجره زمانی در VPRTW می تواند یک طرفه یا دو طرفه باشد.

توابع هدف مسئله

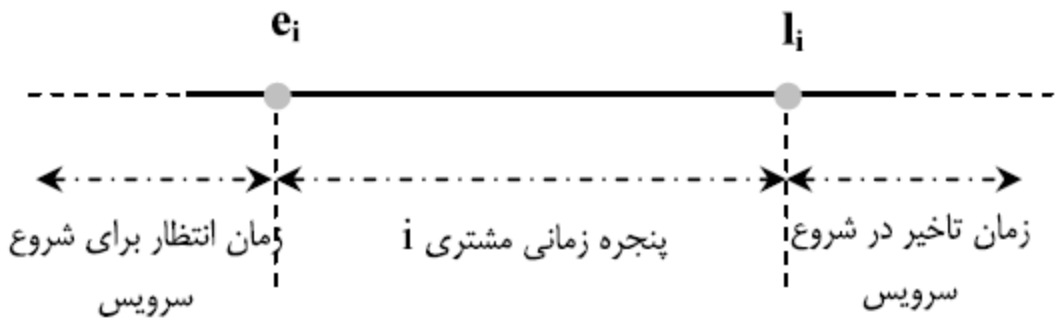
- بهینه سازی کل هزینه های مسیریابی
- حداقل کردن مجموع مدت زمان انتظار وسایل نقلیه
- حداقل نمودن مجموع زمان تاخیرات در ارائه خدمت به مشتریان
- حداقل نمودن کل زمان سفر
- حداقل نمودن تعداد وسایل نقلیه

مفروضات مسئله

- ۱-بیشتر از وسایل حمل ونقل در دسترس نمی تواند استفاده کرد.
- ۲-تقاضای مشتری باید پاسخ داده می شود.
- ۳-هر مشتری دقیقا یک بار ملاقات شود.
- ۴-مسیرشروع وسیله نقلیه از انبار بود و به همان انبار ختم می شود.
- ۵-وسایل حمل ونقل محدودیت ظرفیت دارند.

شرط توجیه پذیری

- رسیدن وسیله نقلیه به یک مشتری قبل از حد پایین پنجره زمانی آن منجر به اضافه زمان انتظار مسیر می شود.
- پنجره زمانی تخصیص داده شده به مشتری در حالت نرم ، ارائه سرویس دیرتر و یا داشتن زمان تاخیر تاثیری در موجه و یا ناموجه بودن مسئله ندارد، ولی در حالت سخت اگر یک مشتری بعد از حد بالای پنجره زمانی خودش سرویس ببیند، جواب غیر موجه می شود.

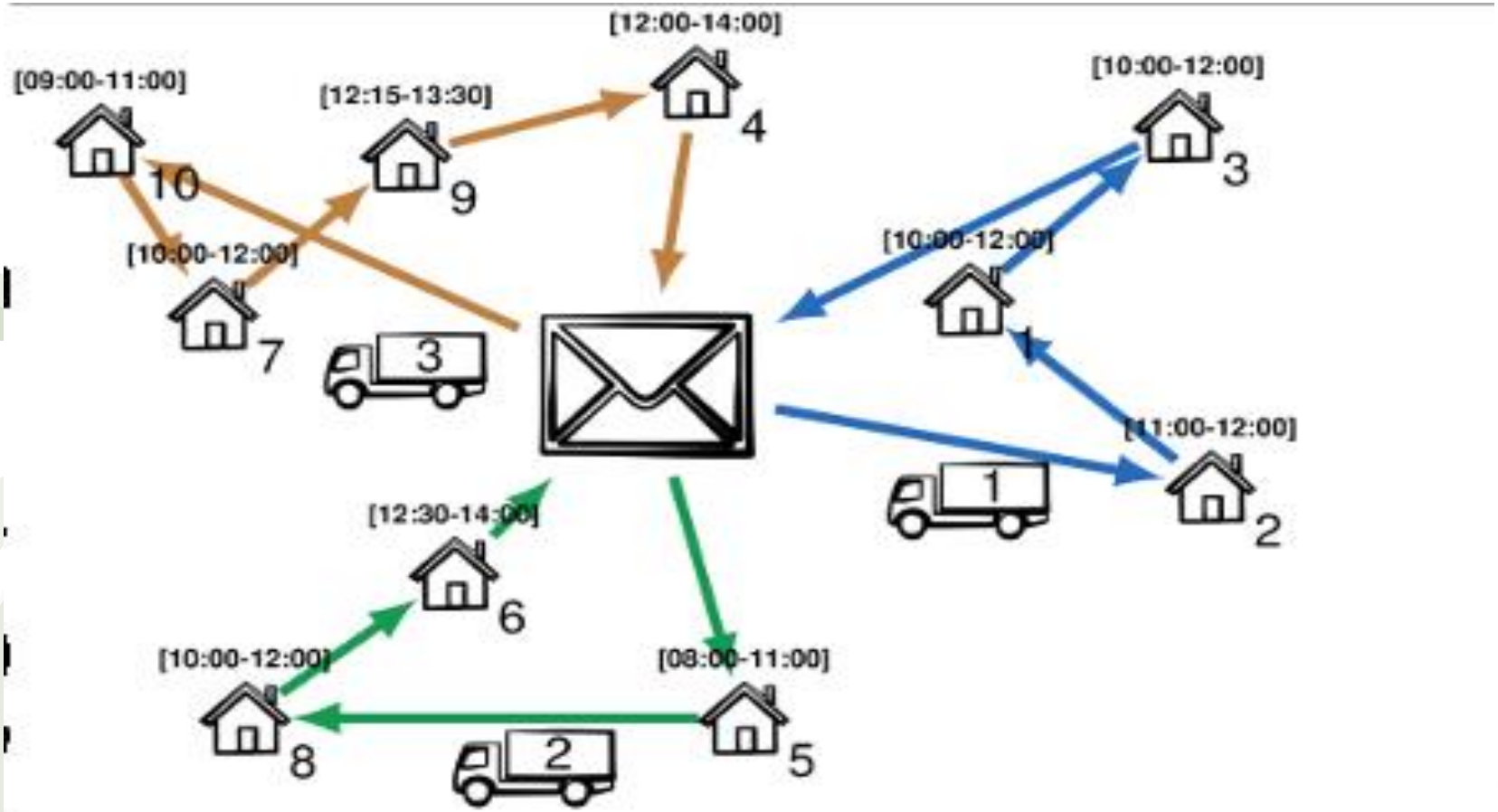


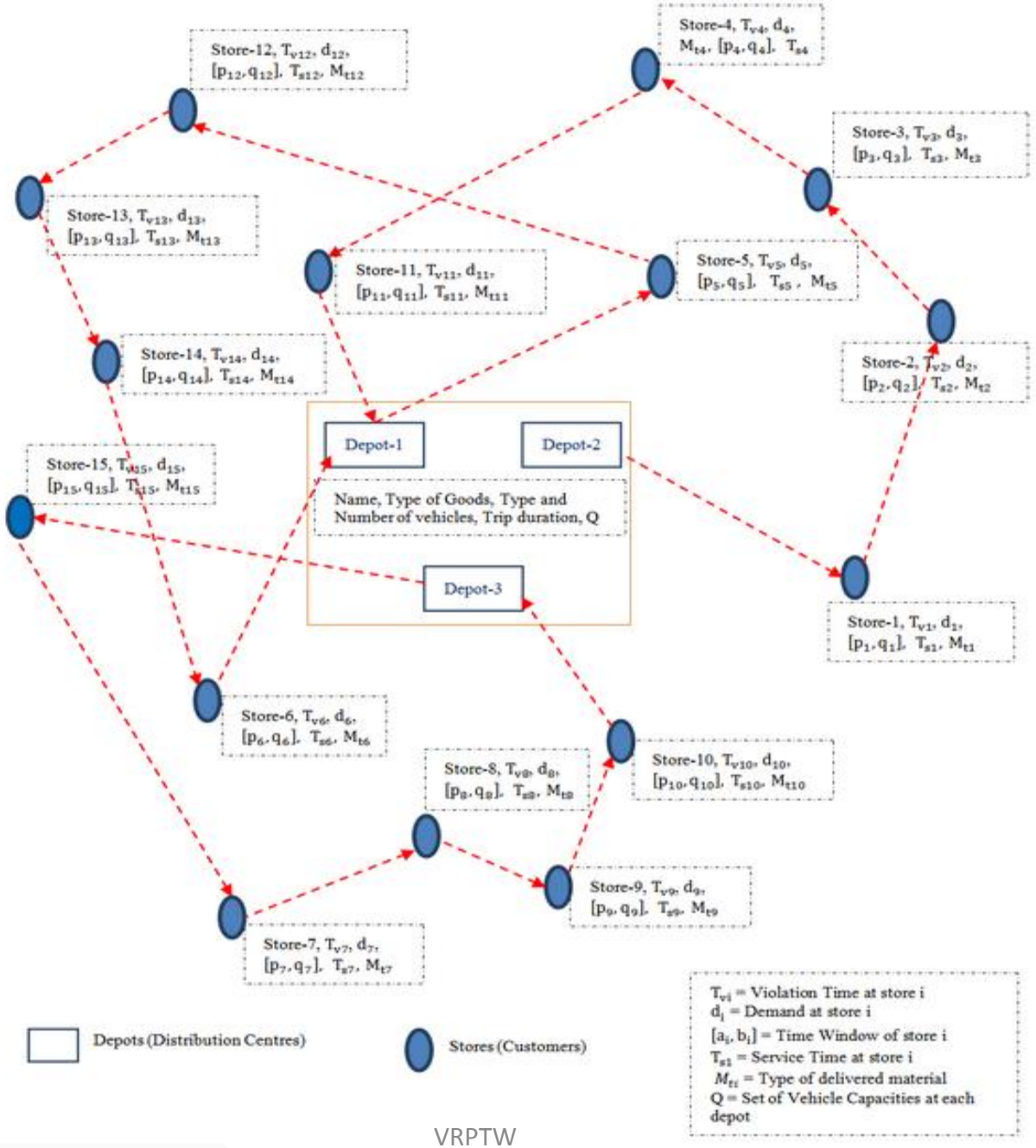
پنجره ی زمانی مربوط به مشتری i

مدل های TW به سه دسته کلی بصورت زیر قابل تقسیم می باشند:

- ۱- **مدل های سخت (VRPHTW)** : در این نوع مدل های TW وسایل نقلیه ملزم به انجام سرویس به یک مشتری در بازه زمانی معین می باشند.
- ۲- **مدل های نرم (VRPSTW)** : در حالت TW نرم سرویس دهی تا حد معینی خارج از بازه زمانی تعیین شده نیز مجاز می باشند. به این صورت که برای هر مشتری علاوه بر یک پنجره زمانی معین یک بازه زمانی سرویس دیگر نیز به نام تعریف می شود و به سرویس هایی که خارج از بازه مجاز صورت می گیرد جریمه تعلق خواهد گرفت.
- ۳- **مدل های سخت و نرم (VRPHTTW)**: حالت TW سخت و نرم ترکیبی از دو حالت فوق است. بدین صورت هر پنجره زمانی شامل یک محدوده نرم و یک محدوده سخت می باشد.

مسئله مسریابی وسیله نقلیه به صورت شماتیک




Store-12, T_{v12}, d_{12}, $[p_{12}, q_{12}]$, T_{s12}, M_{t12}
Store-4, T_{v4}, d_4, M_{t4}, $[p_4, q_4]$, T_{s4}
Store-3, T_{v3}, d_3, $[p_3, q_3]$, T_{s3}, M_{t3}
Store-13, T_{v13}, d_{13}, $[p_{13}, q_{13}]$, T_{s13}, M_{t13}
Store-11, T_{v11}, d_{11}, $[p_{11}, q_{11}]$, T_{s11}, M_{t11}
Store-5, T_{v5}, d_5, $[p_5, q_5]$, T_{s5}, M_{t5}
Store-14, T_{v14}, d_{14}, $[p_{14}, q_{14}]$, T_{s14}, M_{t14}
Store-2, T_{v2}, d_2, $[p_2, q_2]$, T_{s2}, M_{t2}
Depot-1
Depot-2
Name, Type of Goods, Type and Number of vehicles, Trip duration, Q
Store-15, T_{v15}, d_{15}, $[p_{15}, q_{15}]$, T_{s15}, M_{t15}
Depot-3
Store-1, T_{v1}, d_1, $[p_1, q_1]$, T_{s1}, M_{t1}
Store-6, T_{v6}, d_6, $[p_6, q_6]$, T_{s6}, M_{t6}
Store-8, T_{v8}, d_8, $[p_8, q_8]$, T_{s8}, M_{t8}
Store-10, T_{v10}, d_{10}, $[p_{10}, q_{10}]$, T_{s10}, M_{t10}
Store-9, T_{v9}, d_9, $[p_9, q_9]$, T_{s9}, M_{t9}
Store-7, T_{v7}, d_7, $[p_7, q_7]$, T_{s7}, M_{t7}
Depots (Distribution Centres)
Stores (Customers)
T_{vi} = Violation Time at store i
d_i = Demand at store i
$[a_i, b_i]$ = Time Window of store i
T_{si} = Service Time at store i
M_{ti} = Type of delivered material
Q = Set of Vehicle Capacities at each depot

تعریف نمادهای مدل پنجره زمانی سخت

- N: مجموعه مشتریان؛N=V\{0,n+1}
- A: مجموعه یال ها
- x_{ijk} :اگر وسیله نقلیه k یال (i ,j) را ملاقات کند مقدار یک را می گیرد و در غیر اینصورت مقدار o.
- $\Delta^{+}(i)$:مجموعه گره هایی که $(i,j)\epsilon A$. گره هایی که مستقیما از i قابل دسترسی اند.
- $\Delta^{-}(i)$: مجموعه گره هایی که $(j,i)\epsilon A$. گره هایی که از آنها مستقیما به i دسترسی داریم.
- s_i : مدت زمان سرویس دهی گره i.
- w_{ik} : متغیر زمانی مشخص کننده زمان شروع سرویس در گره i توسط وسیله k .
- t_{ij} : زمان سفر از گره i تا گره j

$$(\text{VRPTW})\ \min \sum_{k\epsilon K} \sum_{(i,j)\in A} c_{ij} x_{ijk} \qquad (۱-۱)$$

$$\sum_{k\epsilon K} \sum_{j\in\Delta^+(i)} x_{ijk} = 1 \quad \forall i \in N \qquad (۱-۲)$$

$$\sum_{j\in\Delta^+(0)} x_{0jk} = 1 \quad \forall k \in K \qquad (۱-۳)$$

$$\sum_{i\in\Delta^-(n+1)} x_{i,n+1,k} = 1 \quad \forall k \in K \qquad (۱-۴)$$

$$\sum_{i\in\Delta^-(j)} x_{ijk} - \sum_{i\in\Delta^+(j)} x_{jik} = 0 \quad \forall k \in K, j \in N \qquad (۱-۵)$$

محدودیت های جریان

$$x_{ijk}\left(w_{ik} + s_i + t_{ij} - w_{jk}\right) \leq 0 \quad \forall k \in K, (i,j) \in A \qquad (۶-۱)$$

$$a_i \sum_{j \in \Delta^+(i)} x_{ijk} \leq w_{ik} \leq b_i \sum_{j \in \Delta^+(i)} x_{ijk} \quad \forall k \in K, i \in N \qquad (۷-۱)$$

$$E \leq w_{ik} \leq L \quad \forall k \in K, i \in \{0, n+1\} \qquad (۸-۱)$$

محدودیت های زمانی

$$\sum_{i \in N} d_i \sum_{j \in \Delta^+(i)} x_{ijk} \leq C \quad \forall k \in K \qquad (۹-۱)$$

محدودیت ظرفیت

$$x_{ijk} \geq 0 \qquad \forall k \in K,\ (i,j) \in A, \qquad (۱۰-۱)$$

$$x_{ijk} \in \{0,1\} \qquad \forall k \in K,\ (i,j) \in A. \qquad (۱۱-۱)$$

خطی شدن محدودیت

$$x_{ijk}\left(w_{ik} + s_i + t_{ij} - w_{jk}\right) \leq 0 \quad \forall k \in K, (i,j) \in A \qquad (۶-۱)$$

$$x_{ijk} \in \{0,1\} \quad \forall k \in K, (i,j) \in A \qquad (۱۱-۱)$$

$$w_{ik} + s_i + t_{ij} - w_{jk} \leq \left(1 - x_{ijk}\right) M_{ij}$$

M_{ij} می تواند با $\max\{b_i + s_i + t_{ij} - a_j, 0\}$ جایگزین شود

تعریف نمادهای مسئله VRPSTW

$i=1$: قرارگاه مرکزی در گره

NV : مجموعه خودروهای در دسترس

K_v : ظرفیت وسیله نقلیه v ام

q_i : تقاضای گره i ام

a_v : میانگین سرعت حرکت هر وسیله نقلیه

d_{ij} : فاصله بین دو گره i , j.

t^v_{ij} : زمان سرویس‌دهی و زمان طی مسیر

g^v_{ij} : هزینه یک واحد سفر از گره i به j

S^v_i : زمان شروع سرویس

ye_{vi} : مقدار زمان زود کرد شروع سرویس

yl_{vi} : مقدار زمان دیر کرد شروع سرویس

P_e : جریمه یک واحد زود کرد سرویس

P_l : جریمه یک واحد دیر کرد سرویس

e_i : حد پایین پنجره زمانی سخت

l_i : حد بالای پنجره زمانی سخت

LB_i : حد پایین پنجره زمانی نرم

UB_i : حد بالای پنجره زمانی نرم

$$\text{Min } Z = \lambda_1 \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{v=1}^{NV} g_{ij}^{v} d_{ij} X_{ij}^{v} + \lambda_2 \sum_{i=1}^{n} \sum_{v=1}^{NV} (Pe \times Ye_{vi} + P_l \times Yl_{vi}) \qquad (۱-۲)$$

$$\sum_{j=1}^{n} X_{1j}^{v} = 1 \qquad \forall v \qquad (۲-۲)$$

$$\sum_{i=1}^{n} X_{i1}^{v} = 1 \qquad \forall v \qquad (۳-۲)$$

$$\sum_{i=1}^{n} \sum_{V=1}^{NV} X_{ij}^{v} = 1 \qquad \forall j > 0 \qquad (۴-۲)$$

$$\sum_{i=1}^{n} \sum_{V=1}^{NV} X_{ij}^{v} = 1 \qquad \forall i > 0 \qquad (۵-۲)$$

$$\sum_{i=1}^{n} X_{ip}^{v} - \sum_{j=1}^{n} X_{pj}^{v} = 0 \qquad p : \{1,....,n\}, \forall v \qquad (۶-۲)$$

$$\sum_{v=1}^{V} \sum_{i \in S} \sum_{j \notin S} x_{ij}^{v} \le |S| - r(S) \ \forall S \subseteq A - \{1\}, S \neq \varnothing \qquad (۷-۲)$$

محدودیت های جریان

$$S_j^v + M(1 - X_{ij}^v) - S_i^v - d_{ij}^v / a_v \geq 0 \qquad \forall v, i, j \qquad (۸-۲)$$

$$S_i^v \geq LB_i \qquad \forall v, i \qquad (۹-۲)$$

$$S_i^v \leq UB_i \qquad \forall v, i \qquad (۱۰-۲)$$

$$Ye_{vi} \geq e_i - S_i^v \qquad \forall v, i \qquad (۱۱-۲)$$

$$YL_{vi} \geq S_i^v - L_i \qquad \forall v, i \qquad (۱۲-۲)$$

محدودیت های زمانی

$$\sum_{i=1}^{n} q_i \left[\sum_{j=1}^{n} X_{ij}^v \right] \leq K_v \qquad \forall v \qquad (۱۳-۲)$$

$$X_{ij}^v \in \{0,1\} \qquad \forall v, i, j \qquad (۱۴-۲)$$

تعیین یک جواب اولیه

- از الگوریتم PFIH یا Push-Forward Insertion Heuristic که برای اولین بار توسط سالومون در سال ۱۹۸۷ مطرح گردیده است برای یافتن جواب اولیه استفاده می شود.

- این روش یک متد موثر برای وارد کردن مشتریان به مسیرهای جدید می باشد.

- **توجیه پذیری:**

۱-کل بار(ظرفیت)

۲- کل زمان سفر

۳- مقدار زمانی که زمان ورود مشتری بعدی به جلو برده می شود.

❖ تابع هزینه اولیه برای وارد کردن یک مشتری به عنوان ***گره اول*** به مسیر جدید به صورت زیر تعریف می شود:

$$Cost(C_i) = -\alpha\, t_{0i} + \beta\, l_i + \gamma\,((\frac{|\,p_i - p_j\,|}{360})\, t_{0i})$$

p_i :زاویه قطبی مشتری مورد سوال

p_j :زاویه قطبی آخرین مشتری ملاقات شده از مسیر قبلی می باشد.

t_{0i} :زمان سفر بین گره صفر تا i می باشد.

l_i :دیرترین زمان شروع سرویس در گره i می باشد.

مقادیر زیر بصورت تجربی بدست آمده اند:

- ❖ α =۰.۷
- ❖ β =۰.۲
- ❖ γ =۰.۱

❖ هزینه وارد کردن **گره های دیگر** به یک مسیر با استفاده از رابطه زیر:

$$Cost(C_i) = D_k + \phi W_k + \eta O_k + \kappa T_k$$

❖ D_K : کل مسافت طی شده توسط وسیله نقلیه k

❖ W_K : کل زمان سفر مصرف شده توسط وسیله نقلیه k

❖ O_K : تجاوز از ظرفیت وسیله نقلیه k

❖ T_K : مجموع زمان تاخیراتی می باشد که وسیله نقلیه k در هر مشتری داشته است.

ϕ: برابر با یک درصد D_K در نظر گرفته شده است

η: برابر با ده درصد D_K در نظر گرفته شده است.

κ : برابر با یک درصد D_K در نظر گرفته شده است

الگوریتم PFIH

۱-از یک مسیر خالی با آغاز از دپو شروع نمایید. و قرار دهید r=1 .

۲-اگر تمامی مشتریان مسیردهی شده اند به قدم ۸ بروید، در غیر اینصورت برای همه مشتریان بدون مسیر j هزینه وارد کردن مشتری را محاسبه نمایید و مقادیر را به صورت صعودی مرتب نمایید.

۳-از لیست بدست آمده در قدم قبل مشتری j^* را با حداقل هزینه انتخاب نمایید در حالی که موجه نیز باشد.

۴-مشتری j^* را به مسیر r ضمیمه نمایید و مسیر را به روز رسانی نمایید.

۵- برای همه مشتریان بدون مسیر j و برای همه کناره های k , l در مسیر فعلی، هزینه وارد کردن مشتریان را بین هر k وl محاسبه نمایید.

۶-مشتری j^* و کناره k,l بهینه را انتخاب نمایید و اگر وارد شدن این مشتری بین کناره مفروض موجه می باشد مشتری j بهینه را بین گره k,l بهینه قرار دهید و به قدم ۵ بروید، در غیر اینصورت به قدم ۷ بروید.

۷- مسیر جدیدی را از دپو با r=r+1 آغاز نمایید و به قدم ۲ بروید.

۸- اگر همه مشتریان مسیر دهی شده اند الگوریتم را متوقف نمایید.

روش جستجوی موضعی λ-interchange

روش برای جستجوی همسایگی و بهبود جوابهای حاصله استفاده می شود که بر مبنای تبادل مشتریان بین یک مجموعه از مسیرهای وسایل نقلیه و تولید همسایگی عمل می کند. فرض می نماییم که یک جواب موجه برای مساله VRPTW به فرم $S=\{R_1,\ldots,R_p,\ldots,R_q,\ldots,R_k\}$ باشد که در آن R_p مجموعه مشتریانی می باشند که توسط مسیر وسیله نقلیه p ام سرویس می بینند. روش λ-interchange بین دو مسیر دلخواه مانند R_p و R_q، زیرمجموعه $S_1 \subseteq R_p$ که در آن $|S_1| \leq \lambda$ می باشد با زیرمجوعه ی دیگر $S_2 \subseteq R_q$ که در آن $|S_2| \leq \lambda$ می باشد، جابجا نموده و مجموعه ی مسیر جدیدی به فرم زیر را به وجود می آورد :

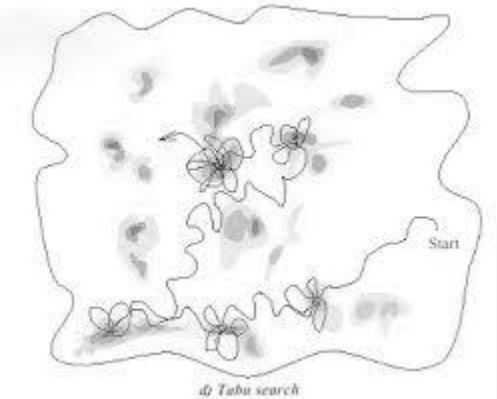
$$R'_p = (R_p - S_1) \cup S_2, R'_q = (R_q - S_2) \cup S_1$$

که باعث ایجاد همسایگی جدیدی به شکل $S' = \{R_1,\ldots,R'_p,\ldots,R'_q,\ldots,R_k\}$ می شود. بنابراین همسایگی $N\lambda(S)$ برای جواب معین S برابر است با همه همسایگی های $\{S'\}$ که به ازای یک λ مشخص حاصل شده اند.

این الگوریتم دارای دو نوع استراتژی اولین بهبود (FB) و بهترین بهبود (GB) می باشد که استراتژی اول، اولین جواب S' را در همسایگی $N\lambda(S)$ باعث کاهش در هزینه مساله می شود را می پذیرد و استراتژی دوم کلیه جوابهای S' را در همسایگی $N\lambda(S)$ مورد بررسی قرار می دهد و جوابی که باعث بهترین بهبود شده است انتخاب می کند.

الگوریتم جستجوی ممنوع (TS)

- الگوریتم جستجوی ممنوع یک استراتژی حافظه ای می باشد که برای اولین بار توسط گلاور مطرح گردیده است.
- در سال ۱۹۹۶ پوتوین و همکارانش کاربرد الگوریتم TS را در مسئله VRPTW توضیح داده اند که هدف آن در این کار حداقل نمودن تعداد وسایل نقلیه به جای مسافت و یا هزینه کل بوده است.
- این الگوریتم لیستی از حرکات یا جواب هایی را که در قبل ملاقات کرده است را نگهداری می کند که تحت عنوان لیست ممنوع از آن یاد می شود و دارای ابعادی متغیر یا ثابت است.



حافظه لیست ممنوع از دو نوع حافظه کوتاه مدت و حافظه بلند مدت می باشد.

لیست ممنوع

این لیست دو نوع اطلاعات *کوتاه مدت* را در خود ذخیره می کند:

۱- حرکات انجام شده را تحت ساختار زیر نگهداری می کند:

(R1, node1, position1, R2, node2, position2)

۲- اطلاعات دیگری که در این لیست ذخیره می شوند جواب هایی هستند که اخیرا ملاحظه شده اند و به صورت یک رشته عدد صحیح کد می شوند.

(3 2 4 5 0 10 6 1 12 11 0 9 8 7 0)

- مدت زمانی که یک جواب و یا یک حرکت در لیست می ماند بستگی به اندازه لیست ممنوع دارد.

نکته: البته داشتن یک لیست بزرگتر فرآیند جستجو را محدود کرده و خیلی سریع آن را پایان می دهد.

لیست کاندید

- دیدگاه *حافظه بلند مدت* در این است که جواب های ممتاز را که در خلال فرآیند جستجو کشف شده اند، در خود ذخیره می کند.

- این ذخیره یابی توسط رشته های صحیح به فرم قبل می باشد ولی رتبه بندی آنها در لیست براساس کل هزینه ایست که به این رشته ها ضمیمه شده است.

- مدت زمان ماندن یک جواب در لیست کاندید بیشتر از لیست ممنوع می باشد.

گوناگونی و تقویت سازی

- جواب اولیه پس از بهبود به عنوان یک جواب به روز شده در نظر گرفته می شود و این جواب و حرکاتی که در این راستا صورت می گیرد در لیست ممنوع ذخیره می شود.
- فرآیند تقویت سازی زمانی پایان می یابد که کل همسایگی مورد جستجو قرار گرفته باشد و هیچ جواب بهتری یافت نشود.
- بعد از یک چرخه از اجرای فرآیند تقویت سازی، جواب فعلی به عنوان جواب بهینه در همسایگی خودش در نظر گرفته می شود و سپس نیاز به یک گوناگونی در فرآیند جستجو می باشد تا مناطق دیگری نیز جستجو شوند.

روند اجرای الگوریتم

۱- یک جواب اولیه با استفاده از PFIH بیابید و بهترین جواب سراسری را با جواب حاضر به روز رسانی نمایید.

۲- لیست حرکات ممنوع، لیست جواب های ممنوع، لیست کاندید را تشکیل داده و جواب فعلی را وارد لیست ممنوع کنید.

۳-فرآیند تقویت سازی را با 2-interchange(GB) انجام دهید.

۴- فرآیند گوناگونی را انجام دهید و بعضی از جواب های رندوم را که در خلال hop های رندوم مواجه شده اند را در لیست کاندید ذخیره کنید.

۵- اگر گوناگونی صورت نگرفته است و یا تعداد کل تکرارها کمتر از ماکزیمم تکرارها بود به قدم ۳ بروید.

۶-در غیر اینصورت بهترین جواب سراسری بدست آمده تاکنون را توسط رویه 2-interchange(GB) بهبود بخشید و جواب بهبود یافته را برگردانید.

با تشکر از توجه شما