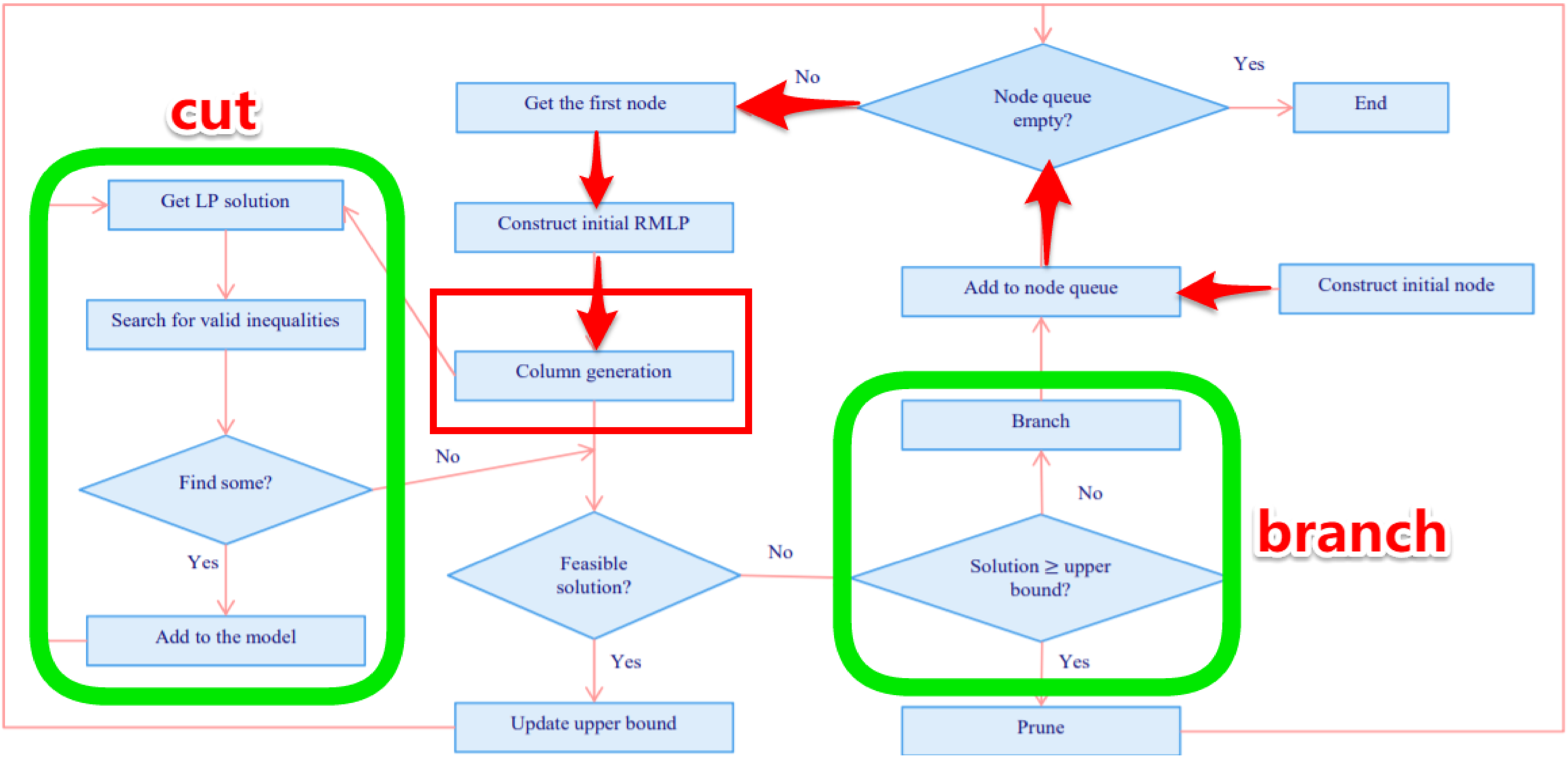
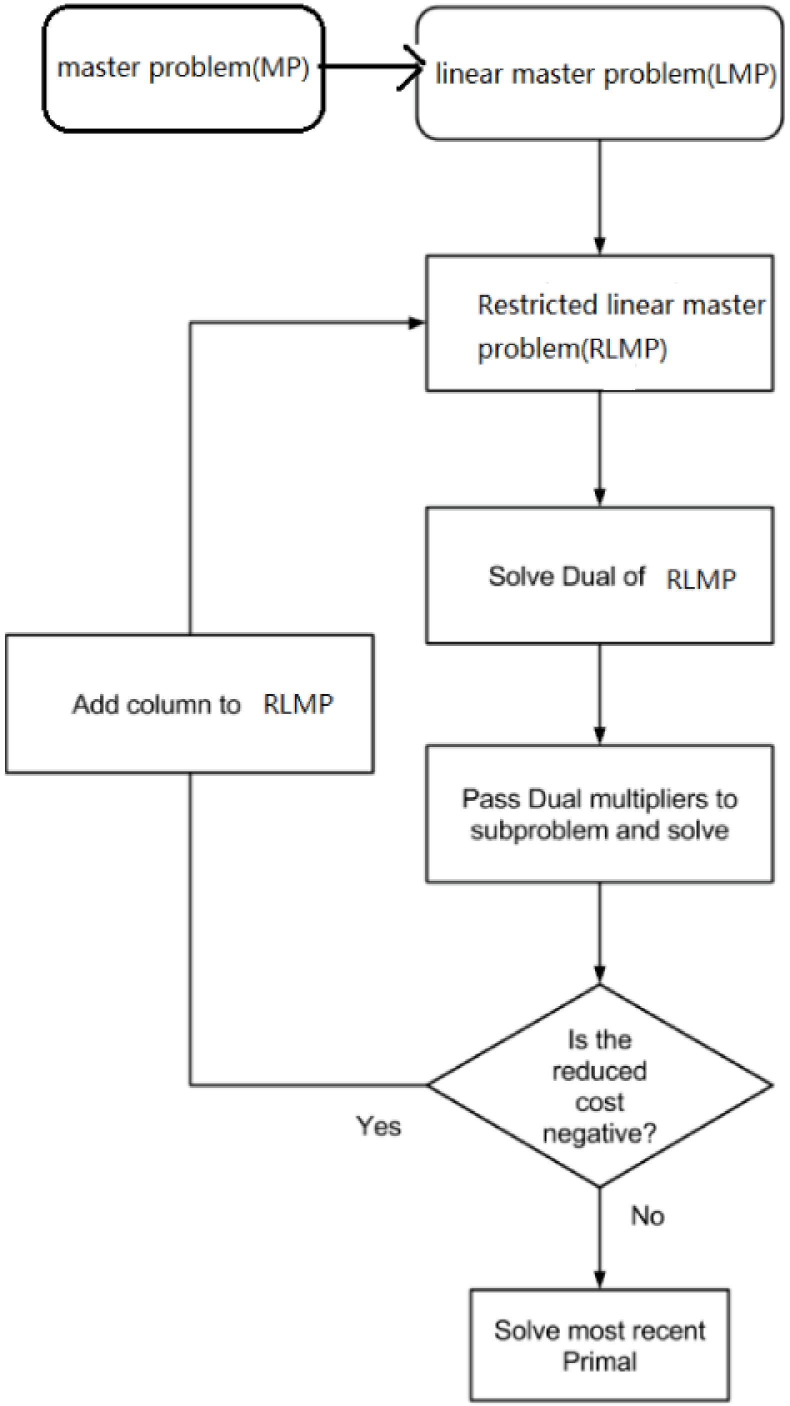
Cuts in Column Generation

汇报人：庄浩城

Branch-and-Price-and-Cut框架





$$(MP(\Omega_1)) \qquad \text{minimize} \sum_{r_k \in \Omega_1} c_k \theta_k \tag{14}$$

subject to

$$\sum_{r_k \in \Omega_1} a_{ik}\theta_k \geq 1 \quad (v_i \in V \setminus \{v_0\}), \tag{15}$$

$$\sum_{r_k \in \Omega_1} \theta_k \leq U, \tag{16}$$

$$\theta_k \geq 0 \quad (r_k \in \Omega_1). \tag{17}$$

$MP(\Omega_1)$ is called the Restricted Master Problem. Let also $D(\Omega_1)$ be the dual program of $MP(\Omega_1)$:

$$(D(\Omega_1)) \qquad \text{maximize} \sum_{v_i \in V \setminus \{v_0\}} \lambda_i + U\lambda_0 \tag{18}$$

subject to

$$\sum_{v_i \in V \setminus \{v_0\}} a_{ik}\lambda_i + \lambda_0 \leq c_k \quad (r_k \in \Omega_1), \tag{19}$$

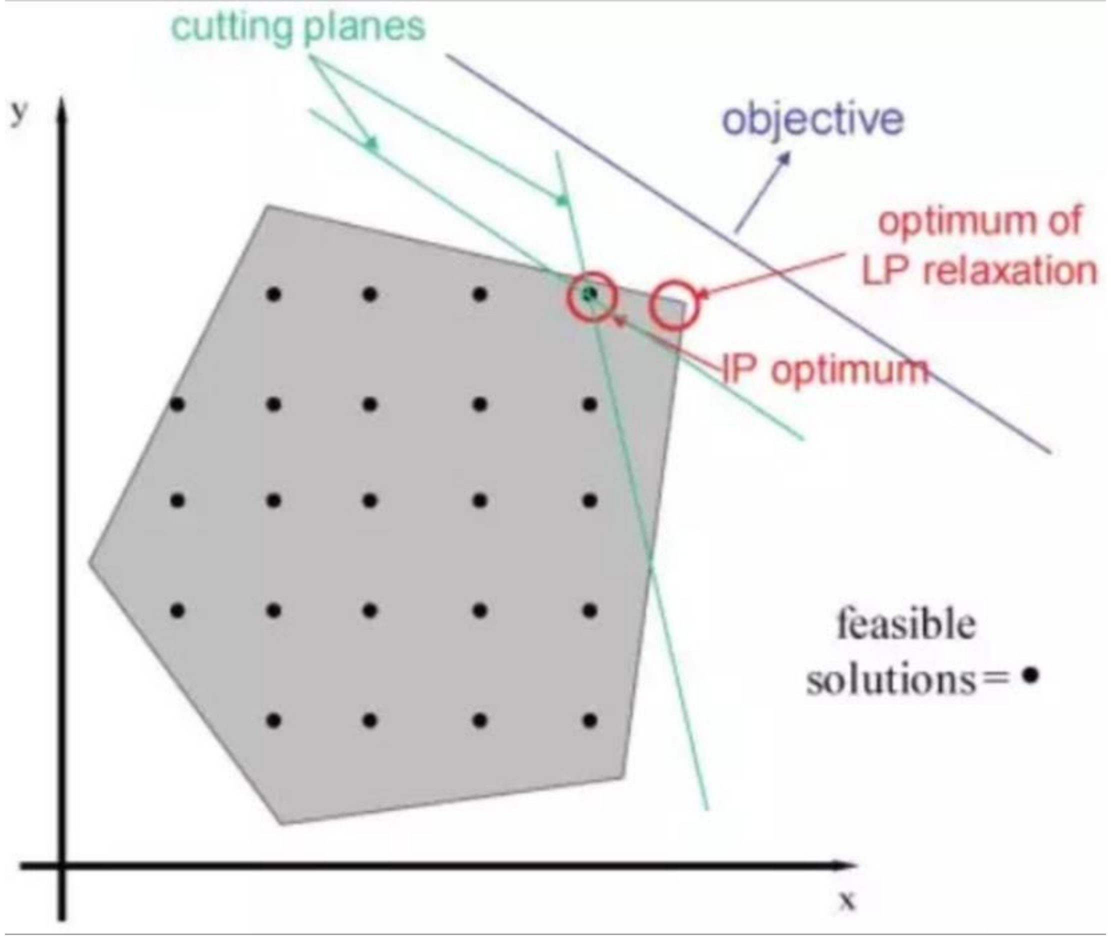
$$\lambda_i \geq 0 \quad (v_i \in V \setminus \{v_0\}), \tag{20}$$

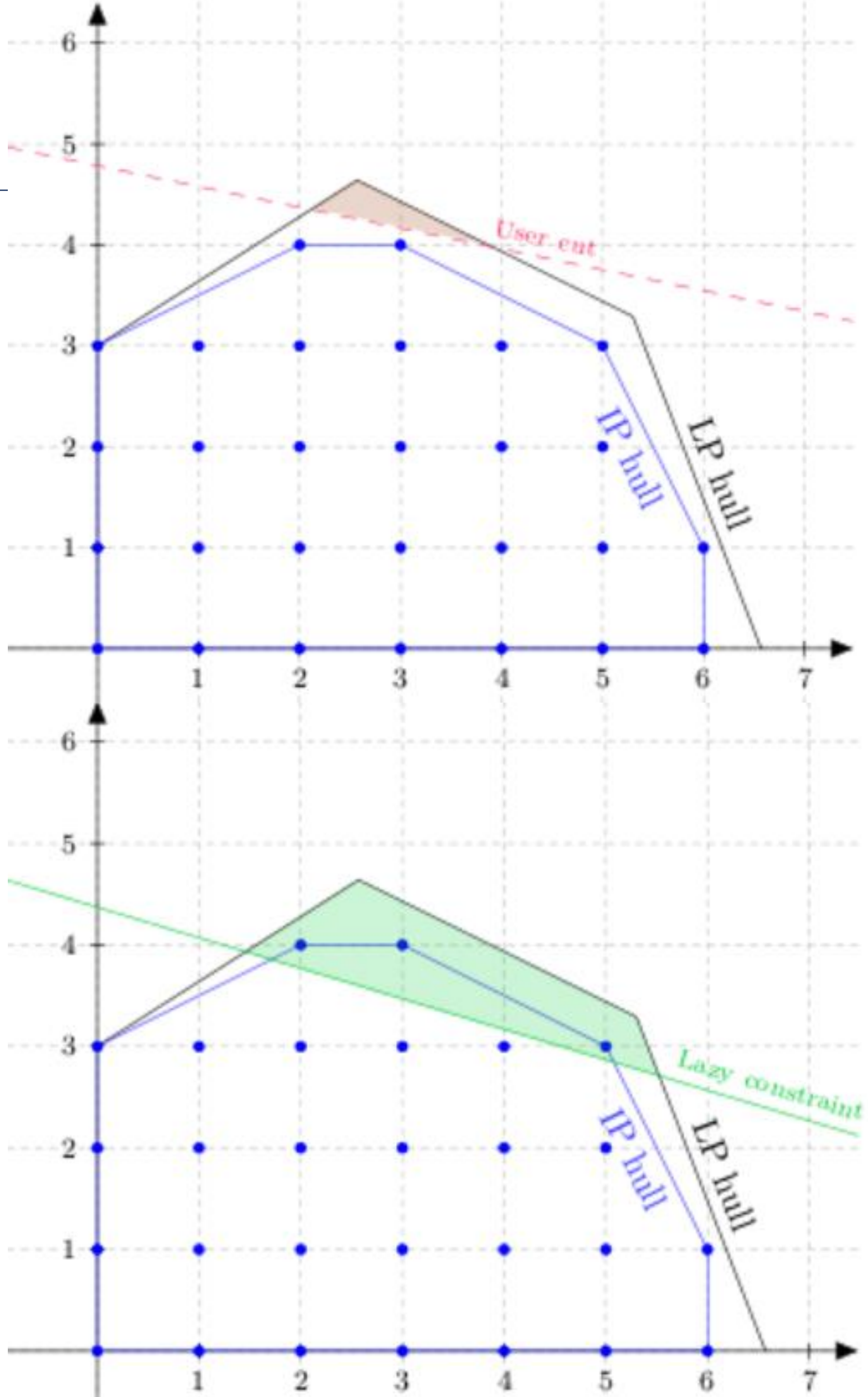
$$\lambda_0 \leq 0. \tag{21}$$

$$c_k - \sum_{v_i \in V \setminus \{v_0\}} a_{ik}\lambda_i^* - \lambda_0^* < 0.$$

$$\sigma_k = c_k - c_B B^{-} a_k$$

$$\sum_{(v_i, v_j) \in A} b_{ijk}(c_{ij} - \lambda_i^*) < 0.$$


cutting planes
objective
optimum of
LP relaxation
IP optimum
y
x
feasible
solutions = •


User cut
IP hull
LP hull
Lazy constraint
IP hull
LP hull
1
2
3
4
5
6
7

Cut type

Robust cuts

- Can be expressed using arc-flow variables
- Duals can be transferred directly on the modified arc costs $\bar{c}_{ij}$
- Structure of PP is not altered

$$\sum_{r\in\Omega}\sum_{(i,j)\in A}\beta_{ij}b^r_{ij}\lambda_r \leq \beta_0$$

$$\bar{c}_r = c_r - \sum_{i\in V'} a^r_i \pi_i - \rho \sum_{(i,j)\in A}\beta_{ij}b^r_{ij}$$

$$= \sum_{(i,j)\in A}(c_{ij} - \pi_j - \rho\beta_{ij})b^r_{ij} = \sum_{(i,j)\in A}\bar{c}_{ij}b^r_{ij}$$

Non-robust cuts

- Defined directly on the MP variables θ_p
- Duals cannot be transferred directly on the modified arc costs $\bar{c}_{ij}$
- Structure of PP is altered (typically, additional resources, modified dominance rule)

$$x_{ij} = \sum_{r\in\Omega} b^r_{ij}\lambda_r,$$

$$\sum_{r\in\Omega}\beta_r\lambda_r \leq \beta_0$$

$$\bar{c}_r = \sum_{(i,j)\in A}\bar{c}_{ij}b^r_{ij} - \beta_r\sigma$$

Robust Cut

K-path inequalities

- S∈N: S是顾客节点N的子集
- k(S): 服务集合S中的节点所需要的车辆数量
- $\delta^-(S) = \{(\boldsymbol{i}, \boldsymbol{j}) \in \boldsymbol{A} | \boldsymbol{i} \in \boldsymbol{N} \backslash \mathrm{S}, \boldsymbol{j} \in \boldsymbol{S}\} \subset \boldsymbol{A}$,进入S的弧的集合，也是弧集的子集
- $X(S) = \sum_{(i,j)\in\delta^-(S)} x_{ij}$，总的进入S的流

转化 $$\boldsymbol{X(S)} = \sum_{(i,j)\in\delta^-(S)} \boldsymbol{x_{ij}} = \sum_{r\in\Omega} \sum_{(i,j)\in\delta^-(S)} \boldsymbol{b_{ijr}\lambda_r} \geq \boldsymbol{k(S)}$$

$$\sum_{r\in\Omega} \sum_{(i,j)\in A} \beta_{ij} b_{ij}^r \lambda_r \leq \beta_0.$$

k(S):Depends on the problem

CVRP

- 转化成装箱问题，箱子容量为车辆容量，货物尺寸为顾客节点需求量(NP-Hard)
- $k(S) = \lceil \sum_{i \in S} q_i / Q \rceil$

VRPTW

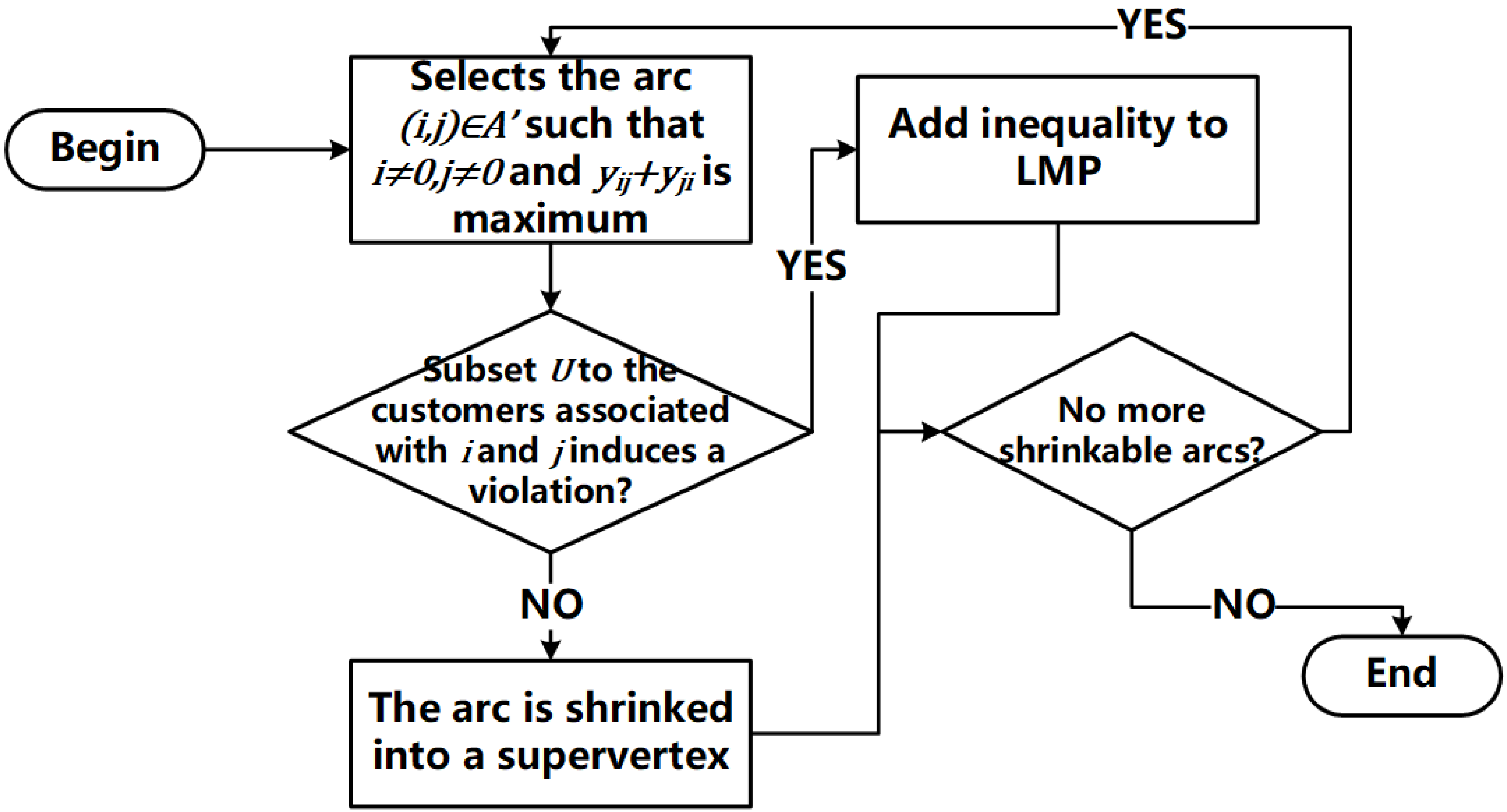
- 解一个小规模的VRPTW(NP-Hard)
- 解一个TSPTW，如果不可行令k(S)=2,否则等于1

Its dual value is subtracted from the modified cost of all arcs in $\delta^{-}(S)$

$$\bar{c}_r = c_r - \sum_{i \in V'} a_i^r \pi_i - \rho \sum_{(i,j) \in A} \beta_{ij} b_{ij}^r$$
$$= \sum_{(i,j) \in A} (c_{ij} - \pi_j - \rho \beta_{ij}) b_{ij}^r = \sum_{(i,j) \in A} \bar{c}_{ij} b_{ij}^r$$

Heuristic Separation Algorithms for k-path Inequalities

Define a support graph $G' = (V', A')$, $\bar{V}$ is the vertex set and $A' = \{(i,j) \in A, \alpha_{ij} > 0\}$



A Integer Programming Formulation

Let x_{ij} represent the number of times a vehicle travels between vertices i and j. (Because the problem is undirected, x_{ij} and x_{ji} represent the same variable.) Let $V_c = V \setminus \{0\}$ denote the set of customers. Given a set of customers $S \subseteq V_c$, let $q(S)$ denote $\sum_{i \in S} q_i$, $\delta(S)$ denote the set of edges in G with exactly one end-vertex in S, $E(S)$ denote the set of edges in G with both end-vertices in S, and $r(S)$ denote the minimum number of vehicles required to serve the customers in S. That is, $r(S)$ is the optimal solution to the *Bin Packing Problem* (BPP) with bin capacity Q and item sizes given by the demands of the customers in S. Finally, given an arbitrary $F \subseteq E$, $x(F)$ will denote $\sum_{e \in F} x_e$. The integer programming formulation is then:

$$\begin{aligned}
&\textit{Minimize} && \textstyle\sum_{e \in E} c_e x_e && \\
&\textit{Subject to:} && x(\delta(\{i\})) = 2 && (i = 1, \ldots, n) && (1)\\
& && x(\delta(S)) \geq 2r(S) && (S \subseteq V_c, |S| \geq 2) && (2)\\
& && x_{ij} \in \{0, 1\} && (1 \leq i < j \leq n) && (3)\\
& && x_{ij} \in \{0, 1, 2\} && (i = 0, j = 1, \ldots, n). && (4)
\end{aligned}$$

Framed capacity inequalities

- For some $S \in V_c$, let $\Omega = \{S_1, \dots, S_p\}$ be a partition of S
- Let $r(S, \Omega)$ equal the minimum number of vehicles needed to service S given that the capacity inequality for each S_i holds with equality

$$x(\delta(S)) + \sum_{i=1}^{p} x(\delta(S_i)) \geq 2r(S, \Omega) + 2\sum_{i=1}^{p} r(S_i).$$

$$\bar{c}_r = c_r - \sum_{i \in V'} a_i^r \pi_i - \rho \sum_{(i,j) \in A} \beta_{ij} b_{ij}^r$$
$$= \sum_{(i,j) \in A} (c_{ij} - \pi_j - \rho\beta_{ij}) b_{ij}^r = \sum_{(i,j) \in A} \bar{c}_{ij} b_{ij}^r$$

$$\sum_{(i,j) \in \delta(S)} x_{ij} = x(\delta(S))$$

this inequality works as follows: if all of the capacity inequalities for the sets S_i are tight, then the summation on the left hand side equals the summation on the right hand side, and therefore $x(\delta(S))$ must be at least $2r(S, \Omega)$, as required

Its dual value times coefficient is subtracted from the modified cost of all arcs in $\delta(S)$

Multistar inequalities

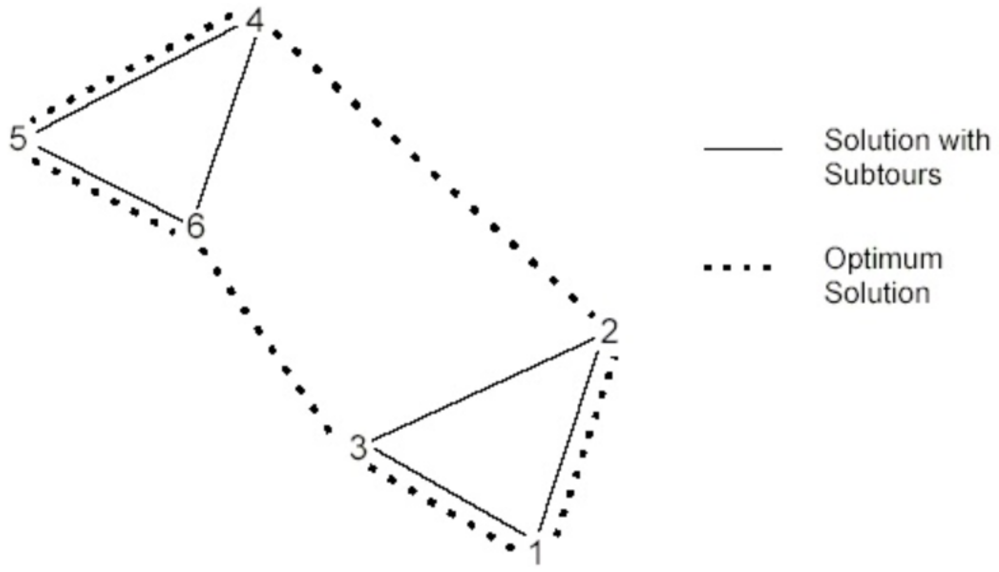
- Let $\rho = 2\frac{d(N)}{Q}, \sigma = 2\sum_{i\in S}\frac{d_i}{Q}$, vehicle visiting the customers in N and using an edge $e \in E(N{:}S)$ must have sufficient free capacity for $S \cup \{i\}, i \in S$
- E(N:S)为连接集合N与集合S的边的集合，$N \cap S = \emptyset$

$$x(\delta(N)) \geq \rho + \sigma x(E(N:S))$$

$$\sum_{(i,j)\in\delta(N)} x_{ij} \geq= 2\frac{d(N)}{Q} + 2\sum_{(i\in S)}\frac{d_i}{Q}\sum_{(i,j)\in E(N:S)} x_{ij}$$

Its dual value times coefficient is subtracted from the modified cost of all arcs in $\delta(S)$

Subtour Elimination Inequalities



$$\sum_{r\in\Omega}\sum_{(i,j)\in S} b_{ijr}\lambda_r = \sum_{i,j\in S} x_{ij} \le |S|-1 \ \ (S\subset V, 2\le |S| \le n-2)$$

在pricing problem中，对应的弧的reduced cost直接减去不等式的对偶变量值

Hypotour Inequalities

Let $\boldsymbol{F} \subset \boldsymbol{E}$ be such that any feasible CVRP solution uses at least one edge From $\boldsymbol{F}$

$$\sum_{r\in\Omega}\sum_{(i,j)\in F} b_{ijr}\lambda_r = \sum_{(i,j)\in F} x_{ij} \geq 1$$

在pricing problem中，对应的弧的reduced cost直接减去不等式的对偶变量值

$$\begin{aligned}\bar{c}_r &= c_r - \sum_{i\in V'} a_i^r \pi_i - \rho \sum_{(i,j)\in A} \beta_{ij} b_{ij}^r \\ &= \sum_{(i,j)\in A} (c_{ij} - \pi_j - \rho\beta_{ij}) b_{ij}^r = \sum_{(i,j)\in A} \bar{c}_{ij} b_{ij}^r.\end{aligned}$$

Non-Robust Cut

Subset Row Inequalities

- S$\subseteq V'$：顾客集合的子集，对应一个行的集合
- 对于路线对应的变量λ_p，只有当$\sum_{i\in S} a_{ip} \geq k$，也就是路线$p$访问了超过k个集合S中的顾客的时候，该变量对应的系数才不为零

$$\sum_{p\in P} \left\lfloor \frac{1}{k} \sum_{i\in S} a_{ip} \right\rfloor \lambda_p \leq \left\lfloor \frac{|S|}{k} \right\rfloor$$

同时访问集合S中的k个节点，且访问的k个节点没有交集的路径最多有$\lfloor \frac{|S|}{k} \rfloor$条

Subset Row Inequalities

- 大多数应用都考虑$|S| \leq 5$的情况
- 令σ为不等式对应的对偶变量，那么对应的主问题中的路线r的reduced cost为$\bar{c}_p = \sum_{(i,j)\in A}(c_{ij} - \pi_j)\alpha_{ijp} - \sigma v_s^r$, $v_s^r = \lfloor \frac{1}{k} \sum_{i \in C_s} \alpha_{ip} \rfloor$，$C_s$ 是定义SRC s的节点的集合

$$\sum_{r\in\Omega} \beta_r \lambda_r \leq \beta_0 \longrightarrow \bar{c}_r = \sum_{(i,j)\in A} \bar{c}_{ij} b_{ij}^r - \beta_r \sigma.$$

Solving the Modified Pricing Problem

- 令V(L)为标签L所访问过的节点，S为子集，则标签L的cost可以表达为
- $\ell(L) = |S \cap V(L)| \bmod k$

$$\hat{c}_p(L) = \bar{c}_p - \sigma \left\lfloor \frac{\sum_{i \in S} \sum_{(i,j) \in \delta^+(i)} \alpha_{ijp}}{k} \right\rfloor = \sum_{(i,j) \in E} \bar{c}_{ij} \alpha_{ijp} - \sigma \left\lfloor \frac{\sum_{i \in S} \sum_{(i,j) \in \delta^+(i)} \alpha_{ijp}}{k} \right\rfloor$$

令$Q = \{q: \sigma_q < 0 \wedge \ell_q(L_i) > \ell_q(L_j)\}$当满足以下条件的时候标签$L_i$ dominates 标签L_j

- $\bar{v}(L_i) = \bar{v}(L_j)$ (部分路径的当前端点)
- $\hat{c}(L_i) - \sum_{q \in Q} \sigma_q \leq \hat{c}(L_j)$
- $r(L_i) \leq r(L_j), \ \forall r \in R$ (R为资源的集合)

Solving the Modified Pricing Problem

- $\ell(L_i) > \ell(L_j)$
- $\overline{v}(L_i) = \overline{v}(L_j)$
- $\hat{c}(L_i) - \sigma \leq \hat{c}(L_j)$
- $r(L_i) \leq r(L_j),\ \forall r \in R$

→ 标签L_i dominates 标签L_j

证明：令ϵ为两标签共同的一个可扩展节点，则对于惩罚项σ的系数有以下关系

$$\left\lfloor \frac{|S \cap \epsilon| + \ell(L_i)}{\mathrm{k}} \right\rfloor \geq \left\lfloor \frac{|S \cap \epsilon| + \ell(L_j)}{\mathrm{k}} \right\rfloor$$

又已知$0 < \ell(L_j) < \ell(L_i) < k$，所以两者在$\sigma$的系数上最多相差1

$$\left\lfloor \frac{|S \cap \epsilon| + \ell(L_i)}{\mathrm{k}} \right\rfloor - 1 \leq \left\lfloor \frac{|S \cap \epsilon| + \ell(L_j)}{\mathrm{k}} \right\rfloor$$

$$\begin{aligned}
\hat{c}(L_i + \epsilon) &= \hat{c}(L_i) + \overline{c}(\epsilon) - \sigma \left\lfloor \frac{|S \cap \epsilon| + \ell(L_i)}{\mathrm{k}} \right\rfloor \\
&= \hat{c}(L_i) - \sigma + \overline{c}(\epsilon) - \sigma \left(\left\lfloor \frac{|S \cap \epsilon| + \ell(L_i)}{\mathrm{k}} \right\rfloor - 1 \right) \\
&\leq \hat{c}(L_j) + \overline{c}(\epsilon) - \sigma \left\lfloor \frac{|S \cap \epsilon| + \ell(L_j)}{\mathrm{k}} \right\rfloor \\
&= \hat{c}(L_j + \epsilon)
\end{aligned}$$

Solving the Modified Pricing Problem

- $\ell(L_i) \le \ell(L_j)$
- $\overline{v}(L_i) = \overline{v}(L_j)$
- $\hat{c}(L_i) \le \hat{c}(L_j)$
- $r(L_i) \le r\left(L_j\right),\ \forall r \in R$

$\longrightarrow$ **标签L_i dominates 标签L_j**

证明：令ϵ为两标签共同的一个可扩展节点，则对于惩罚项σ的系数有以下关系

$$\left\lfloor \frac{|S \cap \epsilon| + \ell(L_i)}{\mathrm{k}} \right\rfloor \le \left\lfloor \frac{|S \cap \epsilon| + \ell\left(L_j\right)}{\mathrm{k}} \right\rfloor$$

从而有

$$\begin{aligned} \hat{c}(L_i + \epsilon) &= \hat{c}(L_i) + \bar{c}(\epsilon) - \sigma \left\lfloor \frac{|S \cap \epsilon| + \ell(L_i)}{\mathrm{k}} \right\rfloor \\ &\le \hat{c}\left(L_j\right) + \bar{c}(\epsilon) - \sigma \left\lfloor \frac{|S \cap \epsilon| + \ell\left(L_j\right)}{\mathrm{k}} \right\rfloor \\ &= \hat{c}(L_j + \epsilon) \end{aligned}$$

Separation of Subset Row Inequalities

$$\boldsymbol{max} \sum_{r\in\Omega} \lfloor \frac{\mathbf{1}}{\boldsymbol{k}} \sum_{i\in C} \boldsymbol{a}_i^r \rfloor \lambda_r - \left\lfloor \frac{\mathbf{1}}{\boldsymbol{k}} |\boldsymbol{C}| \right\rfloor$$

$$\boldsymbol{s.t.} \sum_{i\in V'} \boldsymbol{x}_i = |\boldsymbol{C}|$$

$$\boldsymbol{x}_i \in \{\mathbf{0}, \mathbf{1}\} \ \forall \boldsymbol{i} \in \boldsymbol{V}'$$

- $1 \leq k \leq |C|$
- $x_i = 1$ 如果i $\in C$否则$x_i = 0$
- NP-Hard,所以主要还是通过Enumeration解决

Limited-Memory Subset-Row Cut

- M是Memory Set，且$S \subseteq M \subseteq V'$

$$\sum_{p \in P} \lfloor \frac{1}{k} \sum_{i \in S} a_{ip} \rfloor \lambda_p \leq \lfloor \frac{|S|}{k} \rfloor \xleftarrow{M = V'} \sum_{p \in P} \alpha(S, M, \gamma, p) \lambda_p \leq \lfloor \frac{|S|}{k} \rfloor$$

Algorithm 1 Procedure that calculates the coefficient of a route r in a lm-SRC

```
function α(C, M, p, r)
    coeff ← 0, state ← 0
    for every vertex i in route r (in order) do
        if i ∉ M then
            state ← 0
        else if i ∈ C then
            state ← state + p
            if state ≥ 1 then
                coeff ← coeff + 1, state ← state − 1
    return coeff
```

p对应$\frac{1}{k}$，c对应S

Limited-Memory Subset-Row Cut in Labeling Algorithm

Algorithm 1 Procedure that calculates the coefficient of a route r in a lm-SRC

function $\alpha(C, M, p, r)$
$coeff \leftarrow 0, state \leftarrow 0$
for every vertex i in route r (in order) **do**
    if $i \notin M$ **then**
        $state \leftarrow 0$
    else if $i \in C$ **then**
        $state \leftarrow state + p$
        if $state \geq 1$ **then**
            $coeff \leftarrow coeff + 1, state \leftarrow state - 1$
return $coeff$

- **每一个标签都多一个state属性**
- **不用额外存储coeff，当系数为1的时候直接在reduced cost中减去对偶变量的值**
- **记忆集越小Labeling Algorithm越快**

calculate the memory set of a separated lm-SRC

Algorithm 2 Procedure to calculate the memory set of a separated lm-SRC

function Calculate $M(C, p, \lambda)$
$M \leftarrow C$
for each route r such that $\lambda_r > 0$ and $\lfloor p \sum_{i \in C} a_i^r \rfloor > 0$ **do**
  $state \leftarrow 0$, $Aux \leftarrow \emptyset$
  for every vertex i in route r (in order) **do**
    if $i \in C$ **then**
      $state \leftarrow state + p$
      if $state \geq 1$ **then**
        $M \leftarrow M \cup Aux$, $Aux \leftarrow \emptyset$, $state \leftarrow state - 1$
    else if $state > 0$ **then**
      $Aux \leftarrow Aux \cup \{i\}$
return M

- 令subset C={1,2,3}
- 有route $r_1 = (0-1-4-5-3-6-2-7-1-0), \lambda_1 = 0.2$
  $r_2 = (0-7-2-8-3-0), \lambda_2 = 0.3$
  $r_3 = (0-5-3-4-1-7-9-2-0), \lambda_3 = 0.4$
- 当$\frac{1}{k} = \frac{1}{2}$的时候违反了cut $2\lambda_1 + \lambda_2 + \lambda_3 \leq 1$
- $M = C \cup \{4,5\} \cup \{7\} \cup \{8\} \cup \{4\}$

Elementary Inequalities(Balas et al. 1997)

- 给定路线$r \in \Omega$，令$V^+(r)$为路线r访问的节点的集合
- 给定一个顾客的子集$C \subset V^+$以及一个顾客节点$i \in V^+ \backslash \mathrm{C}$
- 令$\Omega^+(\mathrm{i},\mathrm{C}) = \{\mathrm{r} \in \Omega | \mathrm{a_{ir}} > 0, \mathrm{a_{jr}} = 0, \forall j \in C\}$，为访问过节点$i$但是没有访问过集合$C$中的任意一个节点的路线的集合

$$\lambda_r \leq \sum_{q \in \Omega^+(i, V^+(r))} \lambda_q \quad (1)$$

If route r is used, then the route visiting customer i cannot visit any customer in $V^+(r)$, as at least one customer in $V^+(r)$ would be visited twice

Elementary Inequalities(Pecin et al. 2017)

- **Define multipliers** $p_i^C = (|C| - 1)/|C|$ **and** $p_j^C = 1/|C|$
- **给定一个顾客的子集**$C \subset V^+$**以及一个顾客节点**$i \in V^+ \backslash C$

$$\sum_{r \in \Omega} \lfloor p_i^C a_{ir} + \sum_{j \in C} p_j^C a_{jr} \rfloor \lambda_r \leq 1 \; (2)$$

同时访问节点i**和集合**C**中的任意一个节点的route最多只有一条**

Elementary Inequalities(Pecin et al. 2017)

Proposition 1. For a given route $r \in \Omega$ and a customer $i \in V^+ \backslash V^+(r)$, let $v(i,r) = \lambda_r - \sum_{q \in \Omega^+(\mathrm{i}, \mathrm{V}^+(\mathrm{r}))} \lambda_q$ be the violation of an elementary cut of (1). Similarly, $\mu(i,r) = \sum_{r \in \Omega} \lfloor p_i^C a_{ir} + \sum_{j \in C} p_j^C a_{jr} \rfloor \lambda_r - 1$ be the violation of (2). If Ω contains only elementary routes, then $\mu(i,r) \geq v(i,r)$ and this inequalities might be strict

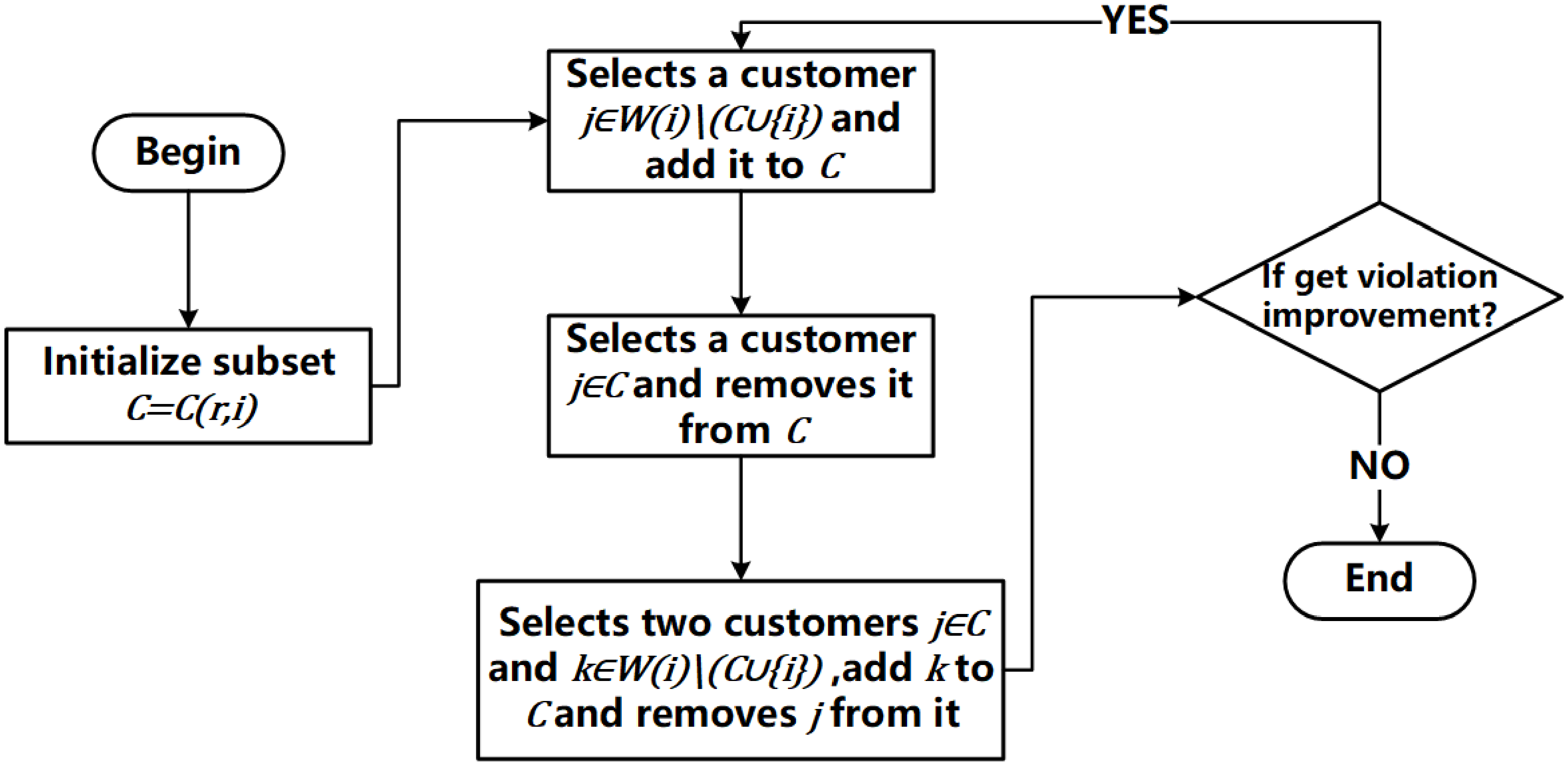
Proof. Let $\Omega(\mathrm{i}) \subseteq \Omega$ be the subset of feasible routes that visit i and let $\Omega(\mathrm{i}, \mathrm{V}^+(\mathrm{r})) = \Omega(\mathrm{i}) \backslash \Omega^+(\mathrm{i}, \mathrm{V}^+(\mathrm{r}))$

$$\sum_{q \in \Omega(i)} \lambda_q = \sum_{q \in \Omega^+(\mathrm{i}, \mathrm{V}^+(\mathrm{r}))} \lambda_q + \sum_{q \in \Omega(\mathrm{i}, \mathrm{V}^+(\mathrm{r}))} \lambda_q = 1 \longrightarrow v(i,r) = \lambda_r + \sum_{q \in \Omega(\mathrm{i}, \mathrm{V}^+(\mathrm{r}))} \lambda_q - 1$$

$$\lfloor p_i^C a_{ir} + \sum_{j \in C} p_j^C a_{jr} \rfloor \geq 1 . \forall q \in \Omega(\mathrm{i}, \mathrm{V}^+(\mathrm{r})) \cup \{r\} \longrightarrow \mu(i,r) \geq v(i,r)$$

Separation

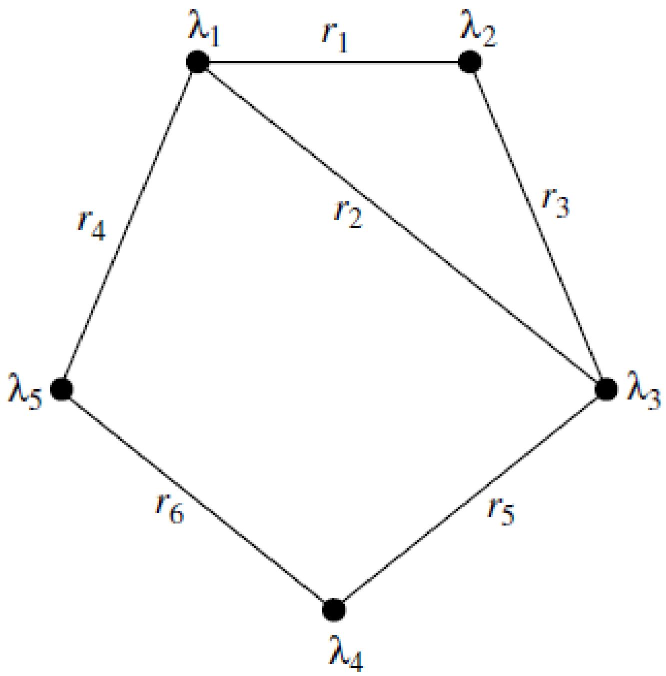
- **Let $\Omega(\mathrm{i})$ be the set of all routes visiting i, $\Omega^+(\mathrm{i})$ be the set of all routes that do not visit i**
- **For each route $r \in \Omega^+(\mathrm{i})$, define $C(r,i) = V^+(r) \cap (\bigcup_{q\in\Omega(\mathrm{i})} V^+(q))$ (the subset of customers visited in r and by at least one route visiting i)**
- **Define $W(i) = \bigcup_{q\in\Omega(\mathrm{i})} V^+(q)$ (the set of all customers that are visited by at least one route visiting customer i)**



Clique Inequalities

Define a undirected graph $G' = (\Omega, E')$, an edge between two vertices exists if they both visit a customer.

A Clique W is a maximal subset of vertices that is conflicting pairwise,



$$\sum_{p \in W} \lambda_p \leq 1$$

Separation

Exact: The separation problem is the Maximum Weighted Clique Problem which is NP-Hard

Heuristic: Starting from any row greedily build cliques by adding columns that are non-orthogonal

Pricing problem and Labeling Algorithm

Set $\chi(W) \subseteq V'$ is conflicting rows of W, row $i \in V'$ belongs to this set if i is conflicting for at least one pair of distinct routes p and q in W, $\chi_{min}(W)$ is the minimal subset of $\chi(W)$

Set $v_p(\chi_{min}(W))$ is the sub-vector of the column restricted to the rows in $\chi_{min}(W)$

$$\begin{bmatrix} 1 & 1 & & & & 1 \\ & & 1 & & & \\ 1 & & 1 & & & 1 \\ 1 & & & 1 & 1 & 1 \\ & 1 & 1 & 1 & & \\ & & & 1 & 1 & 1 \\ & 1 & & & 1 & \\ & & 1 & & 1 & \\ 1 & & & & & \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & 1 & & & \\ 1 & & 1 & & \\ 1 & & & 1 & 1 \\ & 1 & 1 & 1 & \\ & 1 & & & 1 \\ & & 1 & & 1 \end{bmatrix}$$

Pricing problem and Labeling Algorithm

$$\bar{c}_{p} = \sum_{(i,j)\in p} (c_{ij} - \pi_j) - \sum_{W\in\Omega : p\in W} \sigma_W$$

For each clique $W \in \Omega$, $|\chi_{min}(W)| + 1$ binary resources are defined and corresponding components are added to each label

The first $|\chi_{min}(W)|$ of these resource values indicate whether or not each customer of the minimal subset $\chi_{min}(W)$ has been visited, until it is proven that route can enlarge W

Proposition 2. Let L and L' be two labels representing partial paths ending at the same node. Label L dominates label L' (which can be discarded) if

$$T_{cost}(L) - \sum_{W\in\Omega_{LL'}} \zeta_W \leq T_{cost}(L') \tag{7}$$

$$T_r(L) \leq T_r(L') \qquad \forall r \in \mathcal{R}, \tag{8}$$

where $\Omega_{LL'} = \{W \in \Omega : T^W_{inadm}(L) = 1 \text{ and } (T^W_{inadm}(L') = 0 \text{ or } \exists \ell \in \chi_{min}(W) \text{ such that } T^W_{cust_\ell}(L) > T^W_{cust_\ell}(L'))\}$ is the set of cliques W for which the penalty ζ_W could be paid in a feasible extension of L along an arc sequence, while it would not be paid when extending similarly L' (especially if the penalty was already paid, i.e., if $T^W_{inadm}(L') = 0$).

Clique Admissibility Rule

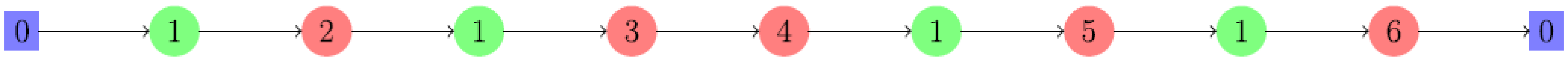
Only the routes p for which $v_p\left(\chi_{min}(W)\right) \geq v_q(\chi_{min}(W))$ for al least one route $q \in \Gamma(\mathrm{W})$ (the resulting set of routes used in clique)

$p =$	1	2	3	4	5	a	b	c
	1	1				1		
	1		1			1		1
	1			1	1	1	1	1
		1	1	1			1	
		1			1			1
			1		1		1	

- a is in the clique since $a =$ column 1
- b is in the clique since $b \geq$ column 4
- c is not in the clique although it conflicts with all columns

K-cycle Elimination Cuts

Given a route $r \in \Omega$ and a vertex $j \in V'$, let α_j^{kr} be a parameter indication the number of times that route r visits vertex j either for the first or after at least k vertices since last visit to j.



Route r with $\alpha_1^r = 4, \alpha_1^{1r} = 4, \alpha_1^{2r} = 2, \alpha_1^{3r} = 1$

$$\sum_{r \in \Omega} \alpha_j^{kr} \lambda_r \geq 1$$

K-cycle Elimination Cuts

All routes $r \in \Omega$ visiting node j twice or more and such that $\alpha_j^{kr} < \alpha_j^r$ will be non-basic in the linear relaxation of master problem.

$$\sum_{r\in\Omega} \alpha_j^r \lambda_r = 1$$ Degree constraint

$$\sum_{r\in\Omega} \alpha_j^{kr} \lambda_r \geq 1 \longrightarrow \sum_{r\in\Omega} (\alpha_j^{kr} - \alpha_j^r) \lambda_r \geq 0$$

Labeling Algorithm

The resource for each generated k-CEC takes value k until reaching j for the first time It is reset to 0 at every visit to j. Then, it counts the number of vertices different from j that are visited consecutively. The dual value associated with this cut is subtracted from the reduced cost every time that vertex j is visited and the value of this resource is greater than or equal to k.These cuts are separated by inspection

$$\bar{c}(L') = \bar{c}(L) + \overline{c_{vw}} - \sum_{k \in R_k(L,L')} \sigma(k)$$

L'对应的点是k-CEC中的一个cut的点且自从上一次访问该点已访问了超过k个不同的点

$$\bar{c}(L) \leq \bar{c}(L') - \sum_{k \in R_k^{L>L'}} \sigma(k)$$

对于所有的cut，L访问的不同的点的数量比L'多

参考

[1] Costa, L., Contardo, C., & Desaulniers, G. (2019). Exact Branch-Price-and-Cut Algorithms for Vehicle Routing. Transportation Science, 53(4), 946–985.
[2] Archetti, C., Bouchard, M., & Desaulniers, G. (2011). Enhanced branch and price and cut for vehicle routing with split deliveries and time windows. Transportation Science, 45(3).(k-path)
[3] Jepsen, M., Petersen, B., Spoorendonk, S., & Pisinger, D. (2008). Subset-Row Inequalities Applied to the Vehicle-Routing Problem with Time Windows. Operations Research, 56(2), 497–511.
[4] Pecin, D., Pessoa, A., Poggi, M., & Uchoa, E. (2017). Improved branch-cut-and-price for capacitated vehicle routing. Mathematical Programming Computation, 9(1), 61–100. (LM-SRc)
[5] Pecin, D., Contardo, C., Desaulniers, G., & Uchoa, E. (2017). New Enhancements for the Exact Solution of the Vehicle Routing Problem with Time Windows. INFORMS Journal on Computing, 29(3), 489–502. (elemantary)
[6] Spoorendonk, S., & Desaulniers, G. (2017). Clique Inequalities Applied to the Vehicle Routing Problem with Time Windows. INFOR: Information Systems and Operational Research.
[7] Contardo, C., & Martinelli, R. (2014). A new exact algorithm for the multi-depot vehicle routing problem under capacity and route length constraints. Discrete Optimization, 12, 129–146. (kcycle)
[8] Lysgaard, J., Letchford, A. N., & Eglese, R. W. (2004). A new branch-and-cut algorithm for the capacitated vehicle routing problem. Mathematical Programming, 100(2), 423–445.