Limited Memory Rank-1 Cuts for the Set Partitioning Formulation of Vehicle Routing Problems

Diego Pecin ¹
Artur Pessoa ²
Marcus Poggi ¹
Haroldo Santos ³
Eduardo Uchoa ²

PUC - Rio de Janeiro ¹
Universidade Federal Fluminense ²
Universidade Federal de Ouro Preto ³

January, 2015
Vehicle Routing Problem (VRP)

**Instance:** Complete graph $G = (V, A)$ with $V = \{0, \ldots, n\}$; vertex 0 is the *depot*, the other vertices are *customers*. Each arc $a \in A$ has a *cost* $c_a$. Customers have *demands*. There is a fleet of *vehicles* in the depot.

**Solution:** A set of routes starting and ending at the depot, attending all customers, and respecting the given operational constraints, with minimal total cost.

Dozens of variants:
- CVRP: Most classical, routes limited only by vehicle capacity
- VRPTW: Customers must also be attended within time windows
- HFVRP: Heterogeneous fleet
Set Partitioning Formulation (Balinski and Quandt [1964])

\[(SPF) \quad \min \sum_{r \in \Omega} c_r \lambda_r \quad (1)\]

S.t.

\[\sum_{r \in \Omega} a^r_i \lambda_r = 1, \quad \forall i \in V_+, \quad (2)\]

\[\lambda_r \in \{0, 1\} \quad \forall r \in \Omega. \quad (3)\]

- \(\Omega\) is the set of routes, \(a^r_i\) is the number of times that customer \(i\) appears in route \(r\).
- Must be solved by column generation. The set \(\Omega\) is often relaxed (allowing some non-elementary routes) in order to make the pricing subproblem more tractable.
Even if $\Omega$ only contains elementary routes, the linear relaxation of SPF is **not** strong enough for efficient branch-and-price. 

- Except when routes are very constrained (e.g., very narrow time windows).

SPF should be combined with cutting, yielding branch-cut-and-price algorithms.
Cuts over Edge/Arc Formulations

Depend of the specific VRP variant:

- **CVRP**: Rounded Capacity, Strengthened Combs
- **VRPTW**: 2-Path
- **HFVRP**: Extended Capacity Cuts

Improve significantly the relaxations. They are robust, their dual variables are translated into edge/arc costs in the pricing. Lead to efficient algorithms.

Seems to be exhausted. Really good new cuts not found in the last years.
Valid for most VRP variants. Several cuts known from the SPP literature: Cliques, Odd holes, ...

Potential for big improvements in the relaxations. However, they are non-robust, each added cut makes the pricing subproblem harder, quickly making it intractable.
Subset Row Cuts (SRCs)

Given $C \subseteq V_+$ and a scalar multiplier $p$, the $(C, p)$-Subset Row Cut is:

$$
\sum_{r \in \Omega} \left( p \sum_{i \in C} a^{r}_{i} \right) \lambda_{r} \leq \lfloor p |C| \rfloor 
$$

(4)

Non-robust cut obtained by a Chvátal-Gomory rounding of $|C|$ constraints in the SPF, less harmful to pricing structure than clique or odd hole cuts.

Interesting SRCs

Given an SRC with base set \( C \), for each integer \( d \), define \( y^d_C \) as the sum of all variables \( \lambda_r \) such that \( \sum_{i \in C} a_{ir} = d \).

The cuts where \( |C| = 3 \) and \( p = 1/2 \) are called **3-Subset Row Cuts (3SRCs)**, expressed as:

\[
y^2_C + y^3_C \leq 1.
\]

- Used in Baldacci et al. [2011] and Contardo and Martinelli [2014]
- Potentially very effective
Interesting SRCs

$|C| = 4$ and $p = 2/3$, 4SRCs:

\[ y_C^2 + 2y_C^3 + 2y_C^4 \leq 2. \]

$|C| = 5$ and $p = 1/3$, 5,1SRCs:

\[ y_C^3 + y_C^4 + y_C^5 \leq 1. \]

$|C| = 5$ and $p = 1/2$, 5,2SRCs:

\[ y_C^2 + y_C^3 + 2y_C^4 + 2y_C^5 \leq 2. \]
Due to their impact in the pricing, not many SRCs could be effectively added to SPF and the potential gains were not achieved.

Pecin et al. [2014] proposed a new technique for greatly reducing the impact of SRCs in the pricing and could obtain the full benefit of those cuts.

- In CVRP, the size of the largest solved instance increased from 150 to 360 customers (improvements in other algorithmic elements also contributed to the advance).

Given $C \subseteq V_+$, a memory set $M$, $C \subseteq M \subseteq V_+$, and a scalar multiplier $p$, the limited memory $(C, M, p)$-Subset Row Cut is:

$$\sum_{r \in \Omega} \alpha(C, M, p, r) \lambda_r \leq \lfloor p|C| \rfloor,$$  \hspace{1cm} (5)

where the coefficient of a route $r$ is computed as:

1: function $\alpha(C, M, p, r)$
2: $\text{coeff} \leftarrow 0$, $\text{state} \leftarrow 0$
3: for every vertex $i \in r$ (in order) do
4:     if $i \not\in M$ then
5:       $\text{state} \leftarrow 0$
6:     else if $i \in C$ then
7:       $\text{state} \leftarrow \text{state} + p$
8:     if $\text{state} \geq 1$ then
9:       $\text{coeff} \leftarrow \text{coeff} + 1$, $\text{state} \leftarrow \text{state} - 1$
10: return coeff
Limited Memory Subset Row Cuts (lm-SRCs)

1: \textbf{function} \ \alpha(C, M, p, r)
2: \ \ \ \textbf{coeff} \leftarrow 0, \ \textbf{state} \leftarrow 0
3: \ \textbf{for every vertex} \ i \in r \ (\text{in order}) \ \textbf{do}
4: \ \ \ \ \textbf{if} \ i \notin M \ \textbf{then}
5: \ \ \ \ \ \ \ \textbf{state} \leftarrow 0
6: \ \ \ \ \textbf{else if} \ i \in C \ \textbf{then}
7: \ \ \ \ \ \ \ \textbf{state} \leftarrow \textbf{state} + p
8: \ \ \ \ \ \textbf{if} \ \textbf{state} \geq 1 \ \textbf{then}
9: \ \ \ \ \ \ \ \textbf{coeff} \leftarrow \textbf{coeff} + 1, \ \textbf{state} \leftarrow \textbf{state} - 1
10: \ \ \ \textbf{return} \ \textbf{coeff}

- If \( M = V_+ \), the function returns \( \lfloor p \sum_{i \in C} a_i \rfloor \)
- Otherwise, the lm-SRC may be a weakening of the corresponding SRC
\( \lambda_{r_1} \) has coefficient 1 in the SRC with \( C = \{1, 2, 3\} \)
Separation of Im-SRCs

Minimum memory for $\lambda_{r_1}$ to have coefficient 1 in the Im 3-SRC
Separation of Im-SRCs

\( \lambda_{r_2} \) has coefficient 1 in the SRC with \( C = \{1, 2, 3\} \)
Separation of Im-SRCs

Minimum memory for $\lambda_{r2}$ to have coefficient 1 in the Im 3-SRC
\( \lambda_{r_3} \) has coefficient 1 in the SRC with \( C = \{1, 2, 3\} \)
Minimum memory for $\lambda_{r3}$ to have coefficient 1 in the Im 3-SRC
The set $M$ of the added Im 3-SRC is the union of the memories those $\lambda$ variables...
The next route of pricings is likely to produce routes that avoid $M$ to have coefficient zero in the Im 3-SRC
The set $M$ may be adjusted in the next round of separation.
Separation of lm-SRCs

If a violated \((C, p)\)-SRC exists, it finds a minimal set \(M\) such that the lm-\((C, M, p)\)-SRC has the same violation.

- Eventually (perhaps in more iterations), the lower bounds obtained with the lm-SRCs will be the same that would be obtained with the SRCs.

The odd algorithmic definition of the lm-SRCs makes sense when considering the labeling dynamic programming algorithm used in the pricing.

- A lm-\((C, M, p)\)-SRC only increases the space of states associated to vertices in \(M\). In practice, there are exponential gains with respect to ordinary SRCs.
Given $C \subseteq V_+$ and a vector of multipliers $p$ of dimension $|C|$, the $(C, p)$-Rank 1 Cut is:

$$\sum_{r \in \Omega} \left[ \sum_{i \in C} p_i a_i^r \right] \lambda_r \leq \left[ \sum_{i \in C} p_i \right]$$ (6)
Given $C \subseteq V_+$, a vector of multipliers $p$ of dimension $|C|$, a memory set $M$, $C \subseteq M \subseteq V_+$, the limited memory $(C, M, p)$-Rank 1 Cut is:

$$\sum_{r \in \Omega} \alpha(C, M, p, r) \lambda_r \leq \left\lfloor \sum_{i \in C} p_i \right\rfloor,$$

(7)

where the coefficient of a route $r$ is computed as:

1: function $\alpha(C, M, p, r)$
2: $\text{coeff} \leftarrow 0$, $\text{state} \leftarrow 0$
3: for every vertex $i \in r$ (in order) do
4:  if $i \notin M$ then
5:  $\text{state} \leftarrow 0$
6:  else if $i \in C$ then
7:  $\text{state} \leftarrow \text{state} + p_i$
8:  if $\text{state} \geq 1$ then
9:  $\text{coeff} \leftarrow \text{coeff} + 1$, $\text{state} \leftarrow \text{state} - 1$
10: return $\text{coeff}$
What are the interesting multipliers?

In a column generation context, cuts must be valid for all possible variable coefficients, not only those in the current restricted problem.
What are the interesting multipliers?

The **Master Set Partitioning** of order $n$ is defined as:

$$
2^n - 1 \sum_{j=1}^{2^n-1} b_j x_j = e_n, \ x \text{ binary},
$$

where $b_j$ is a vector of dimension $n$ with coefficients corresponding to the binary representation of number $j$ and $e_n$ is a unitary vector. For example, if $n = 3$ we have:

$$
\begin{bmatrix}
1 & 0 & 1 & 0 & 1 & 0 & 1 \\
0 & 1 & 1 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 1 & 1 & 1 & 1 \\
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5 \\
x_6 \\
x_7 \\
\end{bmatrix}
= \begin{bmatrix}
1 \\
1 \\
\end{bmatrix}
$$
What are the interesting multipliers?

The **Master Set Partitioning Polyhedron** of order $n$ is defined as:

$$MSPP(n) = \text{Conv}\left\{ \sum_{j=1}^{2^n-1} b_j x_j = e_n, \; x \text{ binary} \right\}.$$ 

We performed a computational study of $MSPP(n)$ for $n \leq 5$ to find the best possible inequalities that can be obtained from up to 5 rows of a SPP.

In particular, we found the multipliers corresponding to all facets of rank 1 and the multipliers that better approximate the facets of higher rank.
MSPPP(3) has a single non-trivial facet:

\[ x_3 + x_5 + x_6 + x_7 \leq 1. \]

This facet has rank 1 and corresponds to multipliers 
\(1/2, 1/2, 1/2\), being equivalent to \(y_C^2 + y_C^3 \leq 1\)

Therefore, the 3SRCs (a subfamily of the clique cuts) are already 
the best possible cuts that can be obtained by considering up to 3 
rows of a SPP.
Analysis of $MSPP(4)$

$MSPPP(4)$ has 8 non-trivial facets, all of rank 1:

- Multipliers $(1/2, 1/2, 1/2, 0)$ and its permutations (3SRCs)
- Multipliers $(2/3, 1/3, 1/3, 1/3)$ and its permutations (New family)

- The original 4SRCs are quite weak, they are the sum of those 8 facets.
- The new cuts have RHS 1 and are another subfamily of cliques.
Analysis of $MSPP(5)$

$MSPPP(5)$ has 294 non-trivial facets, 103 of then have rank 1. The interesting multipliers (along with their permutations) are:

- $(1/3, 1/3, 1/3, 1/3, 1/3)$ (5,1 SRCs)
- $(2/4, 2/4, 1/4, 1/4, 1/4)$ (New family)
- $(3/4, 1/4, 1/4, 1/4, 1/4)$ (New family)
- $(3/5, 2/5, 2/5, 1/5, 1/5)$ (New family)
- $(1/2, 1/2, 1/2, 1/2, 1/2)$ (5,2 SRCs)
- $(2/3, 2/3, 1/3, 1/3, 1/3)$ (New family)
- $(3/4, 3/4, 2/4, 2/4, 1/4)$ (New family)

- The first 4 families have RHS 1 and are subfamilies of clique cuts.
- the last 3 families have RHS 2 and are subfamilies of lifted odd holes.

Remark that we do not know how to separate general cliques or odd holes without destroying the pricing!
Average gaps over a set of hard instances ranging from 36 to 199 customers. Full separation until convergence:

<table>
<thead>
<tr>
<th>Only CG (elementary routes)</th>
<th>2.63</th>
</tr>
</thead>
<tbody>
<tr>
<td>+ robust cuts</td>
<td>0.98</td>
</tr>
<tr>
<td>+ 3SRCs</td>
<td>0.35</td>
</tr>
<tr>
<td>+ 4SRCs + 5SRCs</td>
<td>0.24</td>
</tr>
<tr>
<td>Rank 1 Cuts up to 5 rows</td>
<td>0.17</td>
</tr>
</tbody>
</table>

The new cuts removed 30% of the residual gap. They can help to solve some larger open instances.
Golden_20 (420 customers)

<table>
<thead>
<tr>
<th>Authors</th>
<th>BKS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vidal et al. [2012]</td>
<td>1818.32</td>
</tr>
<tr>
<td>Groër et al. [2011]</td>
<td>1818.25</td>
</tr>
<tr>
<td>Jin et al. [2014]</td>
<td>1817.89</td>
</tr>
<tr>
<td>Liu and Li [2014]</td>
<td>1817.86</td>
</tr>
</tbody>
</table>

- Optimal solution: **1817.59**
- Root LB: 1815.0 (1200 active Rank 1 cuts!)
- B&B Nodes: 370
- Total Time: 7 days (single core i7-3960X 3.30GHz)
Golden_20 (420 customers)

<table>
<thead>
<tr>
<th>Authors</th>
<th>BKS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vidal et al. [2012]</td>
<td>1818.32</td>
</tr>
<tr>
<td>Groër et al. [2011]</td>
<td>1818.25</td>
</tr>
<tr>
<td>Jin et al. [2014]</td>
<td>1817.89</td>
</tr>
<tr>
<td>Liu and Li [2014]</td>
<td>1817.86</td>
</tr>
</tbody>
</table>

- Optimal solution: **1817.59**
- Root LB: 1815.0 (1200 active Rank 1 cuts!)
- B&B Nodes: 370
- Total Time: 7 days (single core i7-3960X 3.30GHz)
Optimal solution of Golden_20, cost 1817.59
Preliminary Results on VRPTW

Being implemented with C. Contardo and G. Desaulniers.

55 out of 56 Solomon instances (100 customers) have gap zero.
Possible lessons from our VRP experience for general BCP construction

Aggressive non-robust cutting may pay

However, cutting and pricing should be fully integrated:

- Besides polyhedral considerations, the non-robust cuts should be designed in order to minimize their impact on the specific algorithm used in the pricing.
- The Im Rank 1 Cuts are good for the labeling algorithm. In an alternative BCP where the pricing is solved, say, by MIP, they would be terrible!
Possible lessons from our VRP experience for general BCP construction

Aggressive non-robust cutting may pay

However, cutting and pricing should be fully integrated:

- Besides polyhedral considerations, the non-robust cuts should be **designed** in order to minimize their impact on the specific algorithm used in the pricing.
  - The Im Rank 1 Cuts are good for the labeling algorithm. In an alternative BCP where the pricing is solved, say, by MIP, they would be terrible!
Possible lessons from our VRP experience for general BCP construction

Aggressive non-robust cutting may pay

However, when designing non-robust cuts, it is desirable to have a parameter that allows a smooth control on cut strength vs impact in the pricing:

- The $M$ parameter has that role in the Im Rank 1 Cuts.
- In our separation we always add the weakest possible cut that does the job.
Possible lessons from our VRP experience for general BCP construction

Aggressive non-robust cutting may pay

However, when designing non-robust cuts, it is desirable to have a parameter that allows a smooth control on cut strength vs impact in the pricing:

- The $M$ parameter has that role in the Im Rank 1 Cuts.
- In our separation we always add the weakest possible cut that does the job
Possible lessons from our VRP experience for general BCP construction

Aggressive non-robust cutting may pay

However, even with all care, the addition of too many non-robust cuts can still break the pricing:

- There must be escape mechanisms.
- In our BCP, when a round of separation makes the solution of a node too slow, it rolls back to a previous state (i.e., it removes the offending cuts).
Possible lessons from our VRP experience for general BCP construction

Aggressive non-robust cutting may pay

However, even with all care, the addition of too many non-robust cuts can still break the pricing:

- There must be escape mechanisms.
- In our BCP, when a round of separation makes the solution of a node too slow, it rolls back to a previous state (i.e., it removes the offending cuts).
A web page containing all CVRP instances from the literature and their optimal/best known solutions:

- Designed as a database, the visible pieces of information are queries
  - Easier maintenance
  - Solutions are automatically checked
  - Instances and solutions can be depicted graphically
- Contains the 100 new instances (X series) between 100 and 1000 customers created by the authors.

http://vrp.galgos.inf.puc-rio.br/
Instance: M-n200-k16

- Number of Customers (n): 199
- Minimum Number of Vehicles (K): 16
- Capacity (Q): 200
- Tigh: 1
- Upper Bound (UB): 1274
- Benchmark: Set M (Christofides, Mingozzi and Toth, 1979) [M]
- Demand: [0, 41]
- Distance: EUC 2D
- Files: 2D

1 Generated by adding customers from M-n151-k12 and the first 49 customers from E-n76-k10 and using the depot and capacity from M-n151-k12

2 Optimal

Customer 66, Demand: 20
Instance: M-n200-k16

- Number of Customers (n): 199
- Minimum Number of Vehicles (K): 16
- Capacity (Q): 200
- Tigh: 1
- Upper Bound (UB): 1274
- Benchmark: Set M (Christofides, Mingozzi, and Toth, 1979) [M]
- Demand: [0.41]
- Distance: EUC 2D
- Files: 

1 Generated by adding customers from M-n151-k12 and the first 49 customers from E-n76-k10 and using the depot and capacity from M-n151-k12
2 Optimal
500-Customer Prize - 300 USD for solving the 68 instances in the X series with up to 500 customers.

- 29 instances in that range still open

1000-Customer Prize - 500 USD for solving all the 100 instances in the X series.

- Only 2 out of the 32 instances with more than 500 customers already solved

Full rules in CVRPLIB, large scale parallelism allowed.
Thank you for your attention!


Aussois-2015 Pecin, Pessoa, Poggi, Santos, and Uchoa Limited Memory Rank-1 Cuts for VRP

