A metaheuristic for solving large instances of the School Bus Routing Problem

Patrick Schittekat  Marc Sevaux  Kenneth Sörensen  Johan Springael

ORTEC / University of Antwerp, Belgium
patrick.schittekat@ua.ac.be

University of Leuven, Belgium
kenneth.sorensen@cib.kuleuven.be

University of South-Brittany, Lorient, France
marc.sevaux@univ-ubs.fr

University of Antwerp, Belgium
johan.springael@ua.ac.be

MIC 2007
Motivations

Request of the Flemish region (Belgium)
- Being transported to school is a right
- Public transportation is organised by the Flemish transportation company
- Routes of buses are annually revised

Interesting features
- Students can walk to a nearby bus stop
- Potential bus stop locations are known

⇒ School Bus Routing problem
The School Bus Routing problem

- a school
The School Bus Routing problem

- a school
- a set of students
The School Bus Routing problem

- a school
- a set of students
- a set of potential bus stops
The School Bus Routing problem

- a school
- a set of students
- a set of potential bus stops
- a maximum walking distance (students → stops)
A possible solution
A possible solution

- students are assigned to bus stops
A possible solution

- students are assigned to bus stops
- two potential bus stops are not visited
A possible solution

- students are assigned to bus stops
- two potential bus stops are not visited
- two bus tours are created
Difference with Basic VRP

Decisions
- How many routes?
- Allocate stops to route
- Order stops within a route
- Allocate items to stops

Objective: Minimize total distance

Restrictions
- Vehicle capacity restrictions
- Unit-stop restrictions
- etc.
Aims of the study

Company related requests
- help the Flemish region in this decision problem
- reducing their costs / keep a high service level

Research related aims
- solve this two-level problem at once
- implement dedicated heuristics and metaheuristics
- take advantages from similar problems and vice versa

Aim of this presentation
- propose a metaheuristic approach
- compare results to optimal solutions of medium-sized instances
- give preliminary results on larger instances
Overview

- Metaheuristics and Exact methods
- Iterated fashion → multiple solutions
- Construction phase (GRASP, stochastic)
  - Clark-Wright savings heuristic
  - $s_{ij} = c_{i0} + c_{0j} - c_{ij}$
  - Three selection types
- Improvement phase (VNS, deterministic)
  - Change two stops within one route
  - Change two stops between routes
  - Replace one stop
  - Add/Remove stops
Student allocation problem

- Special case of the Transportation problem
- Students → supply points
  Routes → demand points

\[
\min \sum_{i \in S} \sum_{j \in R} c_{ij} x_{ij} \quad (1)
\]

\[
\sum_{j \in R} x_{ij} = 1 \quad \forall i \in S \quad (2)
\]

\[
\sum_{i \in S} x_{ij} \leq K \quad \forall j \in R \quad (3)
\]

\[
x_{ij} \in 0, 1 \quad (4)
\]

→ Out-of-kilter method of Ford and Fulkerson \(^1\)

\(^1\) Ford and Fulkerson, Solving the Transportation Problem, Management Science, Vol. 3, No. 1, 24-32
Selection type

- **best**
  - 100
  - 88
  - 65
  - 40
  - 23
  - 15

- **random**
  - 8
  - 65
  - 100
  - 40
  - 15
  - 88

- **roulette**
  - 65
  - 100
  - 40
  - 88

- **total savings**

**Aims**

**Solution technique**

**Calculating bounds**

**Results**

**Conclusion and future work**
Pseudocode - construction heuristic

1: Construct initial solution
2: Allocate students to stops
3: Calculate savings for stops $i$ and $j$
4: for max number of moves do
5: Selection: move($s_1$, $s_2$)
6: if $s_1$ and $s_2$ are in different routes($r_1$, $r_2$) and not interior then
7: Connect routes $r_1$ and $r_2$
8: end if
9: end for
10: Remove non-visited stops
Pseudocode - Variable Neighbourhood Search

1: $i \leftarrow 0$
2: $BestSolution \leftarrow CurrentSolution$
3: while stopping criteria are not met do
4:  Generation: $N(i)$ moves
5:  Selection: move($s_1$, $s_2$) out of $N(i)$
6:  Implement move($s_1$, $s_2$)
7:  if $f_{cost}(CurrentSolution) < f_{cost}(BestSolution)$
     and $f_{feas}(CurrentSolution)$ then
8:     $i \leftarrow 0$
9:     $BestSolution \leftarrow CurrentSolution$
10:    else
11:       $CurrentSolution \leftarrow PreviousSolution$
12:      $i \leftarrow i + 1$
13:  end if
14: end while
Solving medium-sized instances to optimality

Implementation with XPress-MP © Dash Associates (Mosel modeling language)

Algorithm:

1. Implement formulation without subtour elimination constraints
2. Solve the formulation
3. Detect violated constraints
   if no violated constraints GOTO 6
4. Add specific capacity cut constraints
5. GOTO 2
6. END: the solution is optimal

Example with 50 students, 10 potential bus stops provided by a real-life style instance generator
Solving medium-sized instances to optimality

<table>
<thead>
<tr>
<th>It.</th>
<th>Subtours</th>
<th>Cost</th>
<th>CPU (s)</th>
<th>ILP it.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(5,9) (2,10) (4,6)</td>
<td>245.74</td>
<td>1424</td>
<td>129427</td>
</tr>
<tr>
<td>2</td>
<td>(2,8) (4,10)</td>
<td>321.23</td>
<td>2966</td>
<td>229393</td>
</tr>
<tr>
<td>3</td>
<td>(4,7)</td>
<td>327.26</td>
<td>2303</td>
<td>134987</td>
</tr>
<tr>
<td>4</td>
<td>–</td>
<td>330.87</td>
<td>5676</td>
<td>313358</td>
</tr>
<tr>
<td></td>
<td><strong>Optimal solution</strong></td>
<td><strong>330.87</strong></td>
<td><strong>12369</strong></td>
<td><strong>–</strong></td>
</tr>
</tbody>
</table>

**Table:** Instance 1: details of the iterations
### Solving medium-sized instances to optimality

<table>
<thead>
<tr>
<th>It.</th>
<th>Subtours</th>
<th>Cost</th>
<th>CPU (s)</th>
<th>ILP it.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(5,8) (2,6) (4,9)</td>
<td>223.90</td>
<td>10686</td>
<td>639683</td>
</tr>
<tr>
<td>2</td>
<td>(2,9)</td>
<td>235.17</td>
<td>19790</td>
<td>1324610</td>
</tr>
<tr>
<td>3</td>
<td>(2,4,9)</td>
<td>236.24</td>
<td>10900</td>
<td>590679</td>
</tr>
<tr>
<td>4</td>
<td>(2,9,4)</td>
<td>236.25</td>
<td>10722</td>
<td>547891</td>
</tr>
<tr>
<td>5</td>
<td>(3,9) (2,4)</td>
<td>243.81</td>
<td>6858</td>
<td>482269</td>
</tr>
<tr>
<td>6</td>
<td>(6,9) (2,7)</td>
<td>252.57</td>
<td>10406</td>
<td>628056</td>
</tr>
<tr>
<td>7</td>
<td>(3,6)</td>
<td>253.75</td>
<td>7892</td>
<td>426228</td>
</tr>
<tr>
<td>8</td>
<td>(4,6)</td>
<td>254.33</td>
<td>9961</td>
<td>563522</td>
</tr>
<tr>
<td>9</td>
<td>(3,9,4)</td>
<td>255.90</td>
<td>45045</td>
<td>2173264</td>
</tr>
<tr>
<td>10</td>
<td>(3,4,9)</td>
<td>255.90</td>
<td>27594</td>
<td>1314457</td>
</tr>
<tr>
<td>11</td>
<td>(2,6,4)</td>
<td>257.10</td>
<td>26829</td>
<td>1311801</td>
</tr>
<tr>
<td>12</td>
<td>(2,4,6)</td>
<td>257.10</td>
<td>5712</td>
<td>278011</td>
</tr>
<tr>
<td>13</td>
<td>Manually stopped after 37369s</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Unfinished after about 64 hours of CPU time

**Table:** Instance 2: details of the iterations
Solving medium-sized instances to optimality

<table>
<thead>
<tr>
<th>It.</th>
<th>Subtours</th>
<th>Cost</th>
<th>CPU (s)</th>
<th>ILP it.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(6,10) (1,9) (5,8)</td>
<td>294.67</td>
<td>84</td>
<td>9665</td>
</tr>
<tr>
<td>2</td>
<td>–</td>
<td>307.44</td>
<td>143</td>
<td>7575</td>
</tr>
<tr>
<td></td>
<td>Optimal solution</td>
<td>307.44</td>
<td>227</td>
<td>–</td>
</tr>
</tbody>
</table>

**Table:** Instance 3: details of the iterations
## Results on medium-sized instances

<table>
<thead>
<tr>
<th>Instance</th>
<th>Best</th>
<th>Roulette (10)</th>
<th>Roulette (25)</th>
<th>Random (10)</th>
<th>Random (25)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>350.45</td>
<td>334.65</td>
<td>331.63</td>
<td>334.65</td>
<td>330.87</td>
</tr>
<tr>
<td>2</td>
<td>263.29</td>
<td>262.55</td>
<td>262.21</td>
<td>262.82</td>
<td>262.49</td>
</tr>
<tr>
<td>3</td>
<td>309.23</td>
<td>308.51</td>
<td>307.80</td>
<td>308.51</td>
<td>307.98</td>
</tr>
</tbody>
</table>

*Table: solution quality*
Results on medium-sized instances

<table>
<thead>
<tr>
<th>Instance</th>
<th>Best</th>
<th>Roulette</th>
<th>Random</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>10</td>
<td>25</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>25</td>
<td>10</td>
</tr>
<tr>
<td>1</td>
<td>5.92</td>
<td>1.14</td>
<td>0.23</td>
</tr>
<tr>
<td>2</td>
<td>2.41</td>
<td>2.12</td>
<td>1.99</td>
</tr>
<tr>
<td>3</td>
<td>0.58</td>
<td>0.35</td>
<td>0.12</td>
</tr>
</tbody>
</table>

Table: comparison solution quality to bounds in %
Results on medium-sized instances

<table>
<thead>
<tr>
<th>Instance</th>
<th>Best</th>
<th>Roulette</th>
<th>Random</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.09</td>
<td>1.12</td>
<td>2.47</td>
</tr>
<tr>
<td>2</td>
<td>0.15</td>
<td>0.92</td>
<td>2.21</td>
</tr>
<tr>
<td>3</td>
<td>0.04</td>
<td>0.49</td>
<td>1.13</td>
</tr>
</tbody>
</table>

Table: solution speed (s)
Preliminary results on large-sized instances

- 100 stops, 1000 students
- Single construction procedure duration increases till $\approx 240s$
- Possible directions of research:
  1. **Improve coding to increase speed**
  2. **Adaptation/choose better allocation method**
  3. **Use information of previous allocation(s)**
  4. Smart reduction of problem size
  5. Perform student allocation less often
Preliminary results on large-sized instances

Construction heuristic (savings procedure):

<table>
<thead>
<tr>
<th>Method</th>
<th>C = 30</th>
<th>C = 45</th>
<th>C = 60</th>
</tr>
</thead>
<tbody>
<tr>
<td>Out-of-kilter</td>
<td></td>
<td>166.17</td>
<td>97.33</td>
</tr>
<tr>
<td>RelaxIV</td>
<td>&gt; 300</td>
<td>&gt; 300</td>
<td>&gt; 300</td>
</tr>
<tr>
<td>CS2</td>
<td></td>
<td>139.21</td>
<td>156.61</td>
</tr>
<tr>
<td>MCFZIB</td>
<td>19.86</td>
<td>12.45</td>
<td>13.45</td>
</tr>
</tbody>
</table>

**Table:** Preliminary Results (s)

Frangioni and Manca, A Computational Study of Cost Reoptimization for Min-Cost Flow Problems, INFORMS Journal on Computing 18(1), 6170
Conclusion and Future Work

Conclusion

▸ Indication method performs very well for medium-sized instances
▸ We embedded an exact algorithm within an metaheuristic
▸ Tested it on larger instances and found some further research directions
▸ Performed some preliminary tests with other allocation methods and re-optimization

Future work

▸ Speeding up student allocation
▸ Include different schools
▸ Include other objectives
▸ Find strong lower bounds
A metaheuristic for solving large instances of
the School Bus Routing Problem

Patrick Schittekat  Marc Sevaux  Kenneth Sørensen
Johan Springael

ORTEC / University of Antwerp, Belgium
patrick.schittekat@ua.ac.be

University of Leuven, Belgium
kenneth.sorensen@cib.kuleuven.be

University of South-Brittany, Lorient, France
marc.sevaux@univ-ubs.fr

University of Antwerp, Belgium
johan.springael@ua.ac.be

MIC 2007