Combining Two Local Searches with Crossover: An Efficient Hybrid Algorithm for the Traveling Salesman Problem

Weichen Liu, Thomas Weise, Yuezhong Wu and Qi Qi

University of Science and Technology of China (USTC) and Institute of Applied Optimization (IAO) of Hefei University

2017-07-17 [2]
Outline

1 Research Content
2 Local Search Algorithm
   - Lin-Kernighan Algorithm
   - Ejection Chain Method
   - Multi-Neighborhood Search
3 Crossover Operator
   - Heuristic Crossover Operator
   - Order Based Crossover Operator
4 Hybrid Algorithms for the TSP
   - LS-LS-X Hybrids
   - Hybrid Global Search with Local Search
5 Conclusion
Outline

1. Research Content
2. Local Search Algorithm
3. Crossover Operator
4. Hybrid Algorithms for the TSP
5. Conclusion
Traveling Salesman Problem

- Traveling Salesman Problem (TSP): Given a collection of $n$ cities and the travel distance between them, solving a TSP means to find the shortest round-trip tour through all cities and back to the starting point.
Traveling Salesman Problem

- Given: A cost matrix $D = (D_{i,j})$, where $D_{i,j}$ is the cost of traveling from city $i$ to $j$.
- Target: Find a permutation $t$ of the integers from 1 to $n$ minimizing the sum $D_{t[1],t[2]} + D_{t[2],t[3]} + \cdots + D_{t[n],t[1]}$.
- In this paper, we focus on symmetric TSPs, where $D_{i,j} = D_{j,i}$ holds.
- Prominent $\mathcal{NP}$-hard problem in Combinatorial Optimization.
Experimentation Environment

- **TSP Suite**: A holistic benchmark environment for algorithms solving the TSP written in Java. It offers integrated support for implementing, testing, benchmarking and comparing algorithms.
- **Benchmark**: *TSPLIB* contains 110 symmetric TSP instances whose city scale is range from 14 to 85900.
Outline

1. Research Content

2. Local Search Algorithm
   - Lin-Kernighan Algorithm
   - Ejection Chain Method
   - Multi-Neighborhood Search

3. Crossover Operator

4. Hybrid Algorithms for the TSP

5. Conclusion
Outline

1 Research Content

2 Local Search Algorithm
   - Lin-Kernighan Algorithm
   - Ejection Chain Method
   - Multi-Neighborhood Search

3 Crossover Operator

4 Hybrid Algorithms for the TSP

5 Conclusion
Lin-Kernighan Algorithm

The LK10 is an improved LK heuristic algorithm introduced in [5].
Outline

1. Research Content

2. Local Search Algorithm
   - Lin-Kernighan Algorithm
   - Ejection Chain Method
   - Multi-Neighborhood Search

3. Crossover Operator

4. Hybrid Algorithms for the TSP

5. Conclusion
Ejection Chain Method

FSM** is an improved Ejection Chain Method [1, 3].

- It iteratively improves a stem-and-cycle reference structure ($S&C$) by applying two rules.

\[ \text{(a) } S&C \quad \text{(b) Trial} \quad \text{(c) Rule 1} \quad \text{(d) Rule 2} \]
Outline

1 Research Content

2 Local Search Algorithm
   - Lin-Kernighan Algorithm
   - Ejection Chain Method
   - Multi-Neighborhood Search

3 Crossover Operator

4 Hybrid Algorithms for the TSP

5 Conclusion
Multi-Neighborhood Search

Multi-Neighborhood Search (MNS) is an efficient local search algorithm introduced in [4].
Outline

1. Research Content
2. Local Search Algorithm
3. Crossover Operator
   - Heuristic Crossover Operator
   - Order Based Crossover Operator
4. Hybrid Algorithms for the TSP
5. Conclusion
Heuristic Crossover Operator

1. It first selects a random city as the current (starting) city of the offspring tour.

2. Second, it considers the four (directed) edges incident to the current city. Over these edges, a probability distribution is defined based on their cost. The probability associated with an edge incident to a previously visited city is equal to zero.

3. An edge is selected based on this distribution. If none of the parental edges leads to an unvisited city, a random edge is selected.

4. The step 2 and 3 are repeated until a complete tour has been constructed.
Outline

1. Research Content
2. Local Search Algorithm
3. Crossover Operator
   - Heuristic Crossover Operator
   - Order Based Crossover Operator
4. Hybrid Algorithms for the TSP
5. Conclusion
Order Based Crossover Operator

The Order Based Crossover Operator (OX2) selects (at random) several positions in a parent tour and the order of the cities in the selected positions of this parent is imposed on the other parent.

Parent 1

\[ (1 \ 2 \ 3 \ [4 \ 5 \ 6] \ 7 \ 8) \]

\[ (1 \ 2 \ 3 \ [\star \ \star \ \star] \ 7 \ 8) \]

\[ (1 \ 2 \ 3 \ \star \ \star \ \star \ 7 \ 8) \]

\[ (1 \ 2 \ 3 \ \star \ \star \ \star \ 7 \ 8) \]

\[ (1 \ 2 \ 3 \ 4 \ 6 \ 5 \ 7 \ 8) \]

Parent 2

\[ (2 \ [4 \ 6] \ 8 \ 7 \ [5] \ 3 \ 1) \]

\[ (\star \ [4 \ 6] \ \star \ \star \ [5] \ \star \ \star) \]

\[ (\star \ [4 \ 6] \ \star \ \star \ [5] \ \star \ \star) \]

\[ (1 \ 2 \ 3 \ \star \ \star \ \star \ 7 \ 8) \]

\[ (1 \ 2 \ 3 \ 4 \ 6 \ 5 \ 7 \ 8) \]
Outline

1 Research Content
2 Local Search Algorithm
3 Crossover Operator
4 Hybrid Algorithms for the TSP
   - LS-LS-X Hybrids
   - Hybrid Global Search with Local Search
5 Conclusion
Outline

1. Research Content

2. Local Search Algorithm

3. Crossover Operator

4. Hybrid Algorithms for the TSP
   - LS-LS-X Hybrids
   - Hybrid Global Search with Local Search

5. Conclusion
Motivation

- Different local search algorithms have different features, use different data structures, and different search moves neighborhoods. Combining two different local searches means combining their different strengths.
- Crossover helps the search to escape from local optima, while retaining good building blocks.
LS-LS-X Hybrids

Begin

LS One

Improved

N
End

Y

LS Two

Improved

N

Improved

Y

Crossover
Performance Measure and Time

- Empirical cumulative distribution function (ECDF) returns the fraction of runs that have reached a given goal error $F_t$ (normally, $F_t = 0$) for a time measure such as $NT$ or $FE$. It is plotted over the runtime. The earlier and the higher the ECDF rises, the better is the algorithm.
LS algorithms Performance

FSM**, LK10 and MNS algorithms

(e) ECDF for NT and $F_t=0$
LS-LS Hybrids Performance

LS-LS hybrids: FSM**-LK10, LK10-MNS

(f) ECDF for NT and $F_t=0$
LS-LS-X Hybrids Performance

LS-LS-X hybrids: LK10-MNS-HX, FSM**-LK10-OX2

(g) ECDF for NT and $F_t=0$
LS-LS-X Hybrids Performance

Different LS-LS has different suitable Crossover Operator

(h) ECDF for NT and $F_t=0$
LS-LS-X Hybrids

All tested LS-LS-X hybrids

(i) ECDF for NT and $F_t=0$
LS-LS-X Hybrids

City Scale from 128 to 255 and 256 to 511:

(j) $F_t = 0$ for $128 \leq n \leq 255$

(k) $F_t = 0$ for $256 \leq n \leq 511$
LS-LS-X Hybrids

City Scale from 128 to 255 and 256 to 511:

(l) $F_t = 0$ for $128 \leq n \leq 255$

(m) $F_t = 0$ for $256 \leq n \leq 511$
LS-LS-X Hybrids

LS-LS-X Experiment Result:


图: LS-LS-X hybrid algorithms ranking from best to worst. The different algorithm types LS-LS hybrid and LS-LS-X hybrid are highlighted.
Outline

1. Research Content
2. Local Search Algorithm
3. Crossover Operator
4. Hybrid Algorithms for the TSP
   - LS-LS-X Hybrids
   - Hybrid Global Search with Local Search
5. Conclusion
Local Search (LS)

- Feature: Fast convergence, but may get trapped by local optima
Hybrid Evolutionary Algorithm with LS
Hybrid Evolutionary Algorithm with LS

EA-LS-LS-X experiment result:

(a) ECDF for NT and $F_t=0$
Hybrid Evolutionary Algorithm with LS

EA-LS-LS-X experiment result:

(b) ECDF for NT and $F_t=0$
Hybrid Evolutionary Algorithm with LS

EA-LS-LS-X experiment result:

MA(2+8)-LK10-MNS-HX (rank 1), MA(2+4)-LK10-MNS-HX (2),
MA(16+64)-LK10-MNS-HX (3), MA(16+64)-LK10-MNS (4),
MA(16+64)-LK10-MNS-OX2 (5), MA(16+64)-FSM**-LK10-HX (6),
MA(16+64)-FSM**-LK10 (7), MA(16+64)-LK10 (8),
MA(2+8)-FSM**-LK10-OX2 (9), MA(2+4)-FSM**-LK10-OX2 (10),
MA(2+8)-FSM**-LK10-HX (11), MA(2+4)-LK10-MNS-OX2 (12),
MA(2+8)-LK10-MNS-OX2 (13), MA(16+64)-FSM**-LK10-OX2 (14),
MA(2+8)-LK10-MNS (15), MA(2+4)-FSM**-LK10-HX (16),
MA(2+4)-LK10-MNS (17), MA(2+8)-FSM**-LK10 (18),
MA(16+64)-FSM** (19), MA(2+4)-LK10 (20), MA(2+8)-LK10 (21),
MA(2+4)-FSM**-LK10 (22), MA(16+64)-MNS (23), MA(2+8)-FSM** (24),
MA(2+4)-FSM** (25), MA(2+8)-MNS (26), MA(2+4)-MNS (27).

图: EA hybrid algorithms ranking from best to worst.
Outline

1. Research Content
2. Local Search Algorithm
3. Crossover Operator
4. Hybrid Algorithms for the TSP
5. Conclusion
Conclusion

- The new LS-LS-X hybrids are better than their pure LS algorithm and LS-LS hybrid components.

- The new EC-LS-LS-X hybrids outperform the LS-LS-X algorithms as well as EC-LS and EC-LS-LS hybrids. MA(2+4)-LK10-MNS-HX becomes the new most powerful hybrid EA algorithm in the huge collection of algorithms and experimental results of the popular TSP Suite.

- Different LS-LS hybrids have different suitable crossover operators.
Weichen Liu, Thomas Weise, Yuezhong Wu, and Raymond Chiong.
Hybrid ejection chain methods for the traveling salesman problem.

Weichen Liu, Thomas Weise, Yuezhong Wu, and Qi Qi.
Combining two local searches with crossover: An efficient hybrid algorithm for the traveling salesman problem.

Weichen Liu, Thomas Weise, Yuezhong Wu, Dan Xu, and Raymond Chiong.
An improved ejection chain method and its hybrid versions for solving the traveling salesman problem.

Benchmarking optimization algorithms: An open source framework for the traveling salesman problem.
*IEEE Computational Intelligence Magazine (CIM)*, 9(3):40–52, August 2014.

Yuezhong Wu, Thomas Weise, and Weichen Liu.
Hybridizing different local search algorithms with each other and evolutionary computation: Better performance on the traveling salesman problem.
Thanks!
Appendix

*TSPLIB* contains 110 symmetric TSP instances whose city scale is range from 14 to 85900.

<table>
<thead>
<tr>
<th>Instance Scale</th>
<th>Number of instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>$0 \sim 2^4 - 1$</td>
<td>1</td>
</tr>
<tr>
<td>$2^4 \sim 2^5 - 1$</td>
<td>8</td>
</tr>
<tr>
<td>$2^5 \sim 2^6 - 1$</td>
<td>8</td>
</tr>
<tr>
<td>$2^6 \sim 2^7 - 1$</td>
<td>17</td>
</tr>
<tr>
<td>$2^7 \sim 2^8 - 1$</td>
<td>20</td>
</tr>
<tr>
<td>$2^8 \sim 2^9 - 1$</td>
<td>11</td>
</tr>
<tr>
<td>$2^9 \sim 2^{10} - 1$</td>
<td>13</td>
</tr>
<tr>
<td>$2^{10} \sim 2^{11} - 1$</td>
<td>15</td>
</tr>
<tr>
<td>$2^{11} \sim 2^{12} - 1$</td>
<td>6</td>
</tr>
<tr>
<td>$2^{12} \sim 2^{13} - 1$</td>
<td>4</td>
</tr>
<tr>
<td>$2^{13} \sim 2^{14} - 1$</td>
<td>4</td>
</tr>
<tr>
<td>$2^{14} \sim 2^{15} - 1$</td>
<td>1</td>
</tr>
<tr>
<td>$2^{15} \sim 2^{16} - 1$</td>
<td>1</td>
</tr>
<tr>
<td>$2^{16} \sim 2^{17} - 1$</td>
<td>1</td>
</tr>
</tbody>
</table>

表: Distribution of Symmetric Instances in *TSPLIB*
Global Search VS Local Search

Global Search VS Local Search:

(a) Global Search VS Local Search