University of Nottingham
UK | CHINA | MALAYSIA

Automated Design of Local Search Algorithms for Vehicle Routing Problems

Weiyao Meng
10 June, 2024

University of Nottingham
Weiyao.meng2@Nottingham.ac.uk
https://weiyaomeng.github.io/

Content

- Background
- Introduction of Automated Algorithm Design
- AutoGCOP: A General Framework for Automated Composition
- Machine Learning for Automated Algorithm Composition
- Future Work Directions

Background

Combinatorial Optimisation Problems (COPs)

- Vehicle Routing
- Nurse Rostering
- Bin Packing

Some widely investigated COPs

Metaheuristics

Single solution-based algorithms
E.g., Tabu Search, Simulated Annealing, Iterated Local Search.

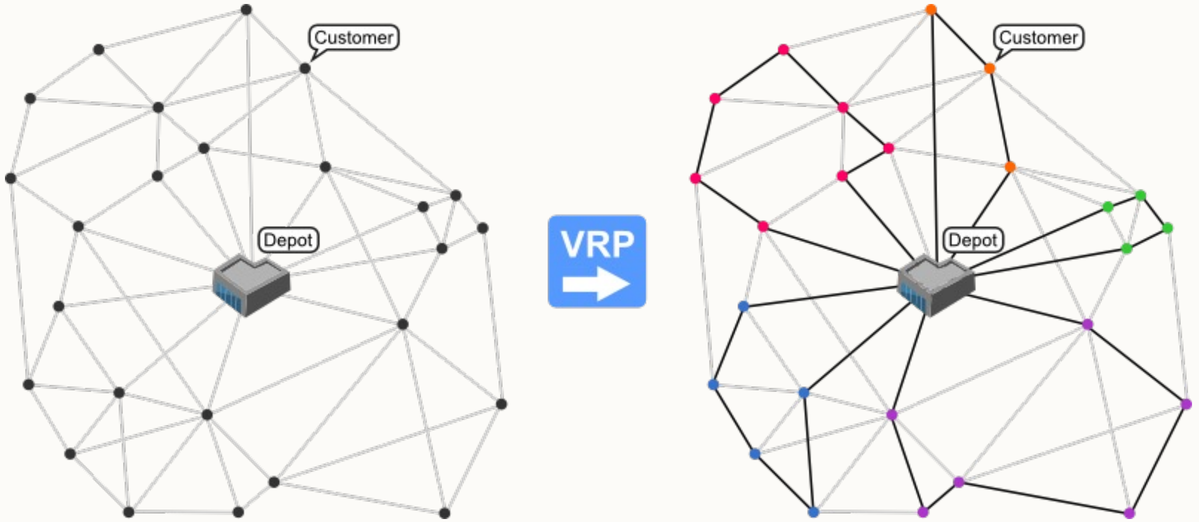
Population-based algorithms
E.g., Genetic Algorithms, Particle Swarm Optimisation, Ant Colony Optimisation.

Elementary components:

- Operators o_i,
- Acceptance Criteria a_j

Background

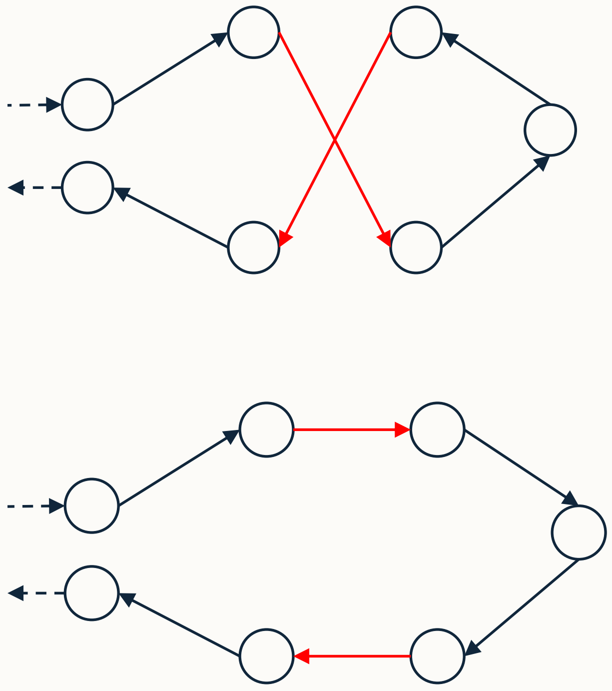
Vehicle routing problems (VRPs)



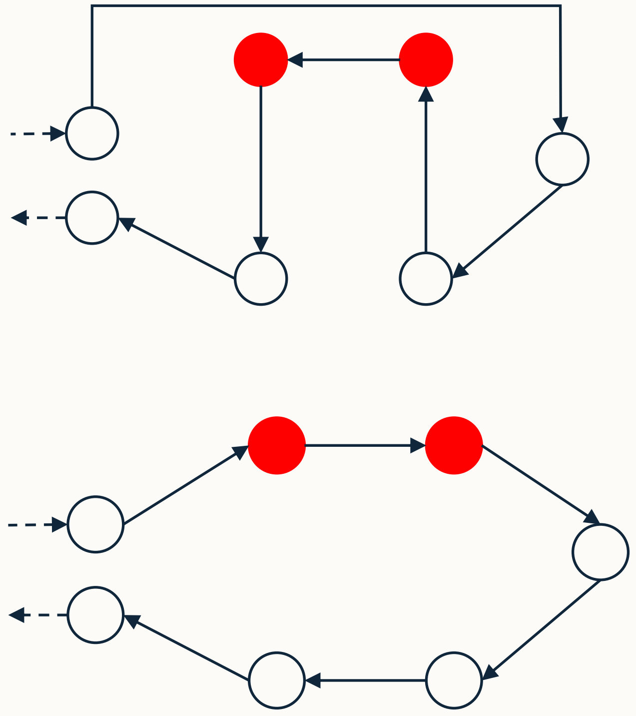
✓Total distance
✓Number of vehicles
✓Carbon emissions
✓Time constraints
✓…

Background

Vehicle routing problems (VRPs)



2-opt



Or-opt

Motivation

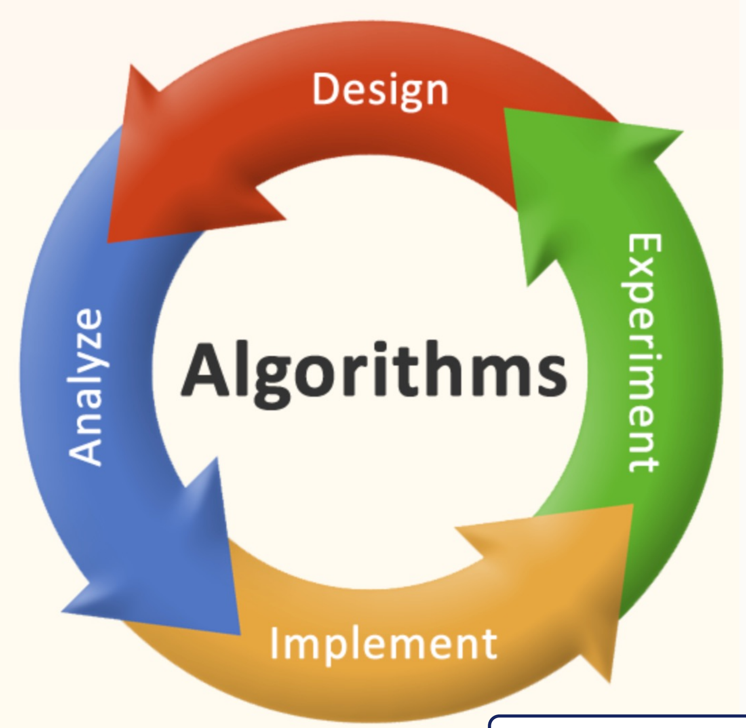
High demand of effective search algorithms

Designing effective algorithms: expertise & efforts

Automation

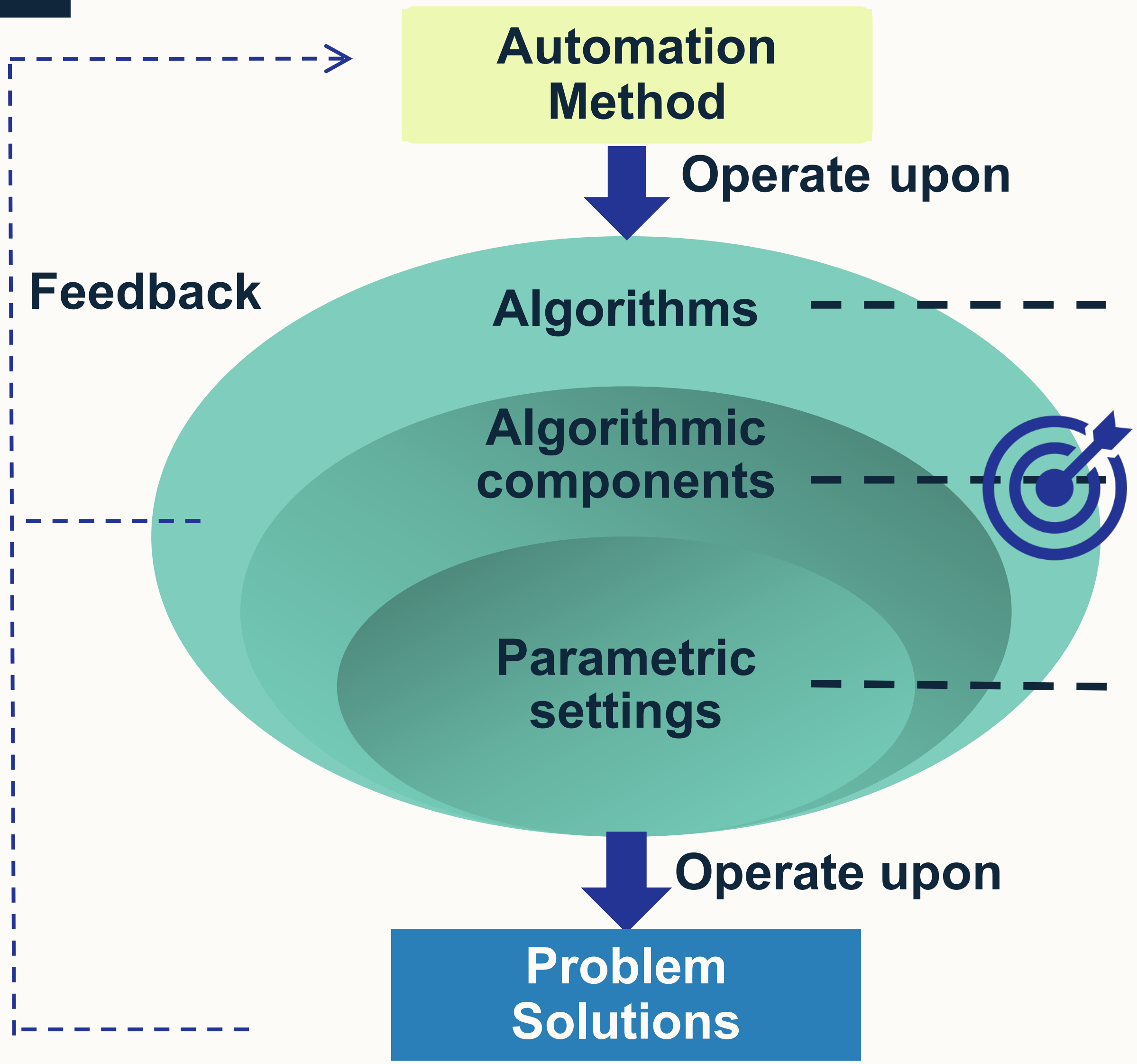
Explore a larger scope of candidate algorithms

Remove heavy reliance on human experts



- Decisions in algorithm design
  - Algorithm-level decisions
    - Existing algorithms
    - Existing algorithmic components
    - Parameter settings
  - Problem-level decisions
    - Representation
    - Objective function
    - Constraint handling

Introduction



Auto selection
to automatically select from a portfolio of chosen algorithms

Auto composition
to automatically compose heuristics or components of some algorithms

Auto configuration
to automatically configure the parameters of pre-defined target algorithm(s)

[1] Qu, R., Kendall, G. and Pillay, N., 2020. The general combinatorial optimization problem: Towards automated algorithm design. *IEEE Computational Intelligence Magazine*, *15*(2), pp.14-23.

Introduction

The General COP (GCOP) for AutoAD

- *Algorithms – **compositions** of elementary algorithmic components*

$$c_i = (o_2, a_1, o_3, o_1, a_4, \ldots)$$

A composition of elementary algorithmic components (e.g., basic operators o_i, acceptance criteria a_j, etc.).

- *Basic operators o_i in GCOP - an example*

$$o_{chg}(k, h1_w, h1_b)$$

use $h1_b$ to change the values of k decision variables selected by $h1_w$.

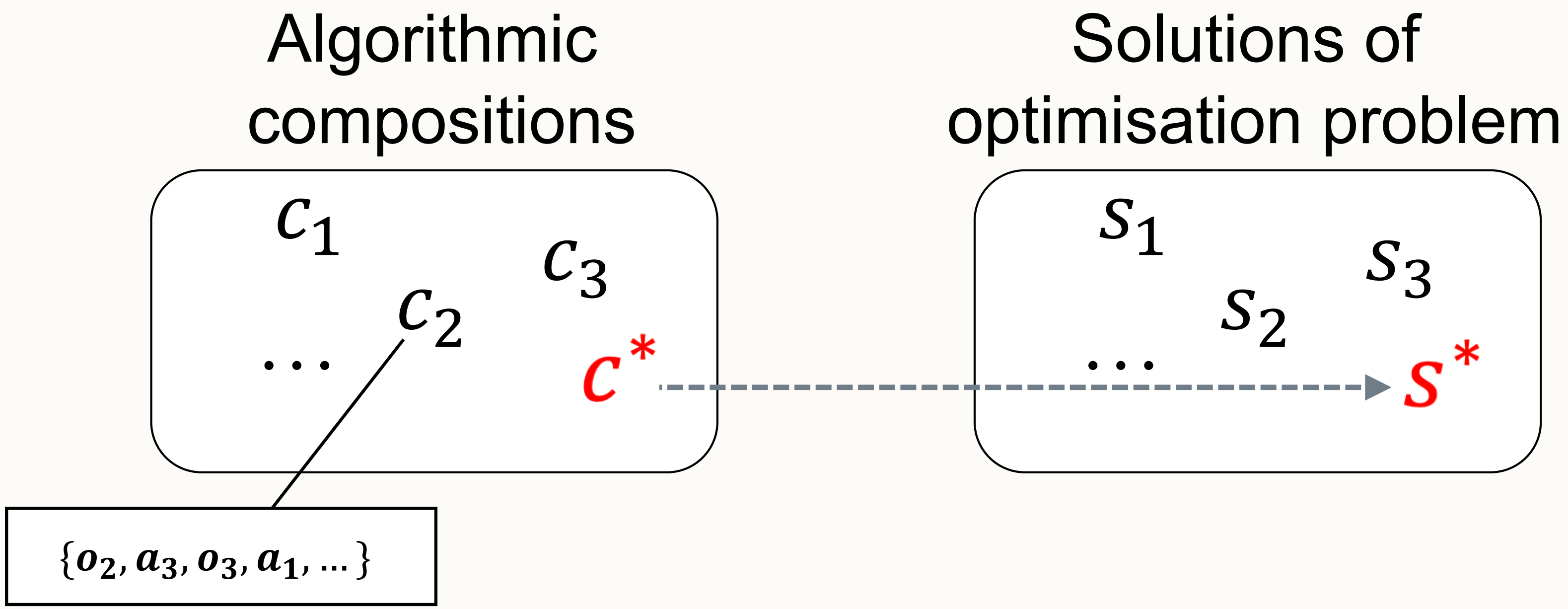
For solving NRP ***- change shift type of k nurses***

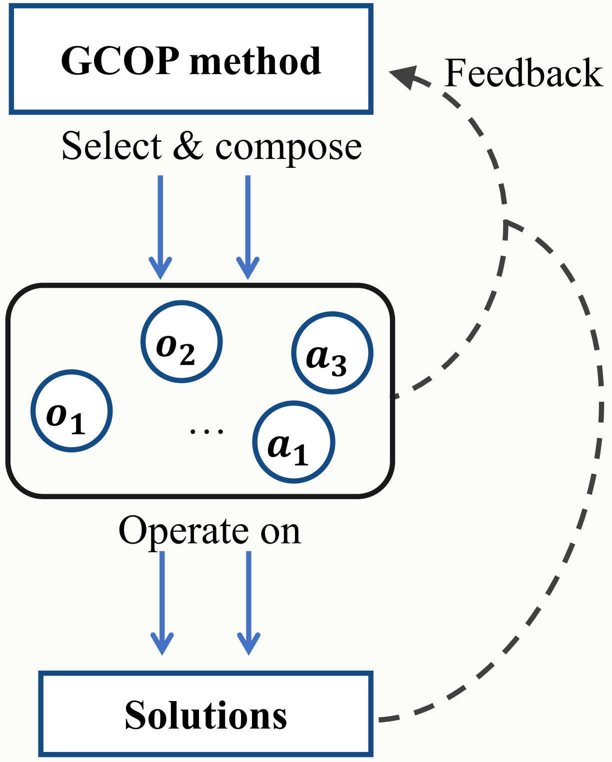
For solving VRP ***– shift k customers***

[1] Qu, R., Kendall, G. and Pillay, N., 2020. The general combinatorial optimization problem: Towards automated algorithm design. *IEEE Computational Intelligence Magazine*, *15*(2), pp.14-23.

Introduction

General Combinatorial Optimisation Problem (GCOP)

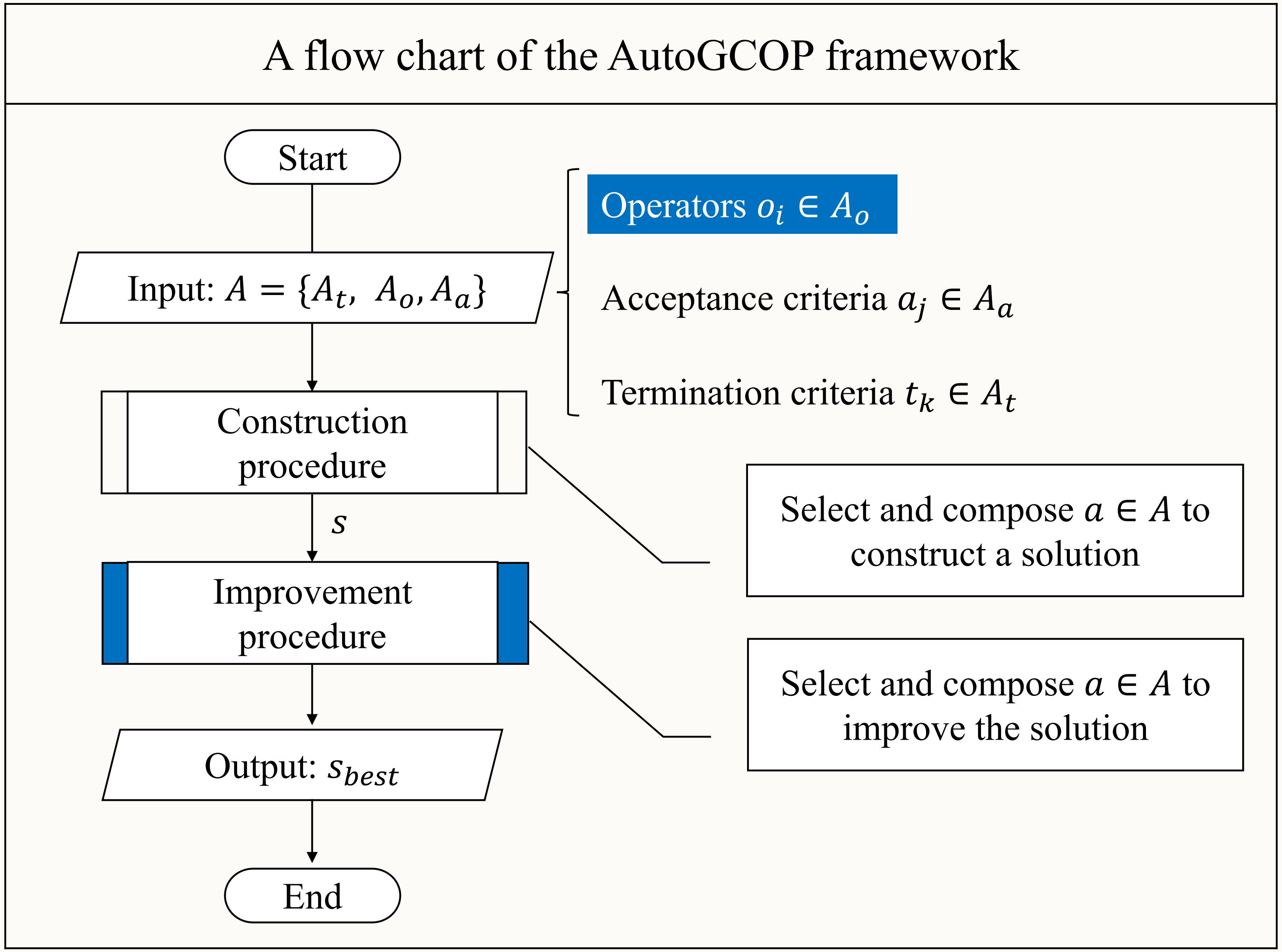


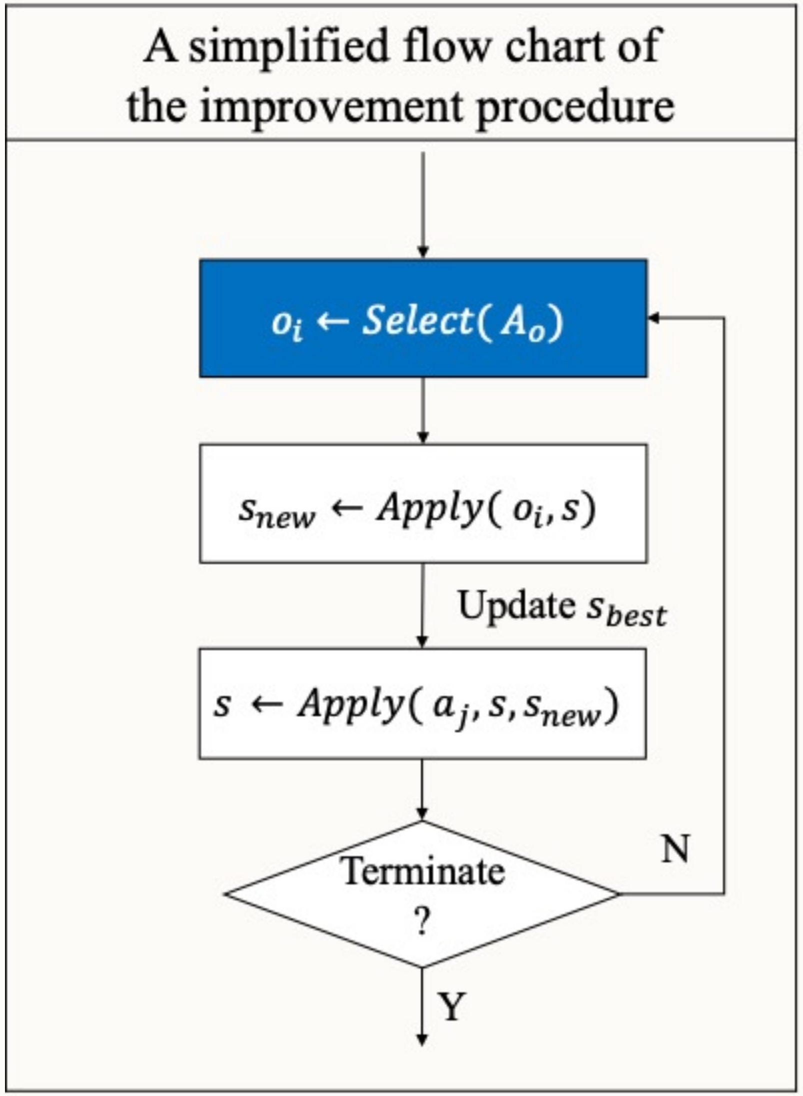


A composition of elementary algorithmic components (e.g., basic operators o_i, acceptance criteria a_j, etc. [1]).

[1] Qu, R., Kendall, G. and Pillay, N., 2020. The general combinatorial optimization problem: Towards automated algorithm design. *IEEE Computational Intelligence Magazine*, *15*(2), pp.14-23.

AutoGCOP Framework

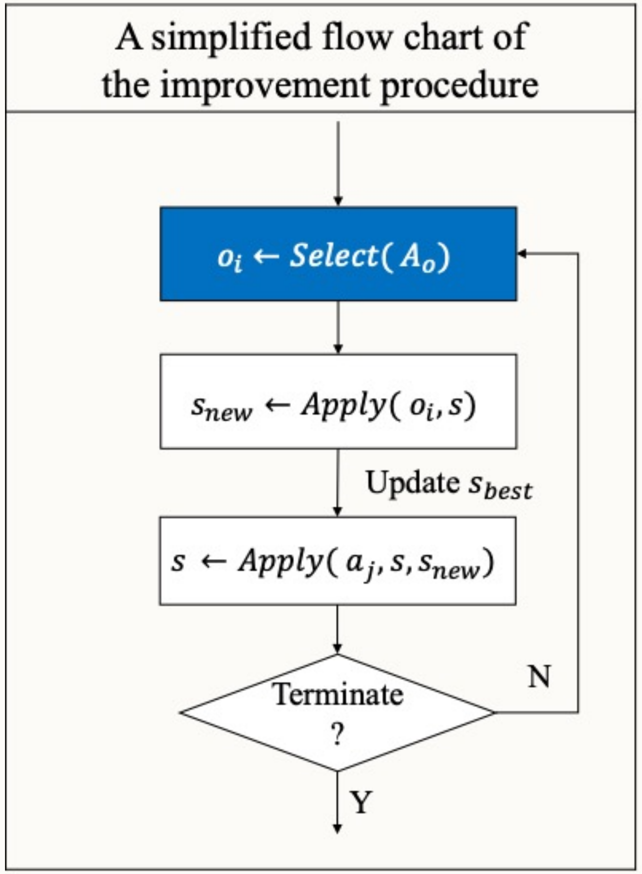


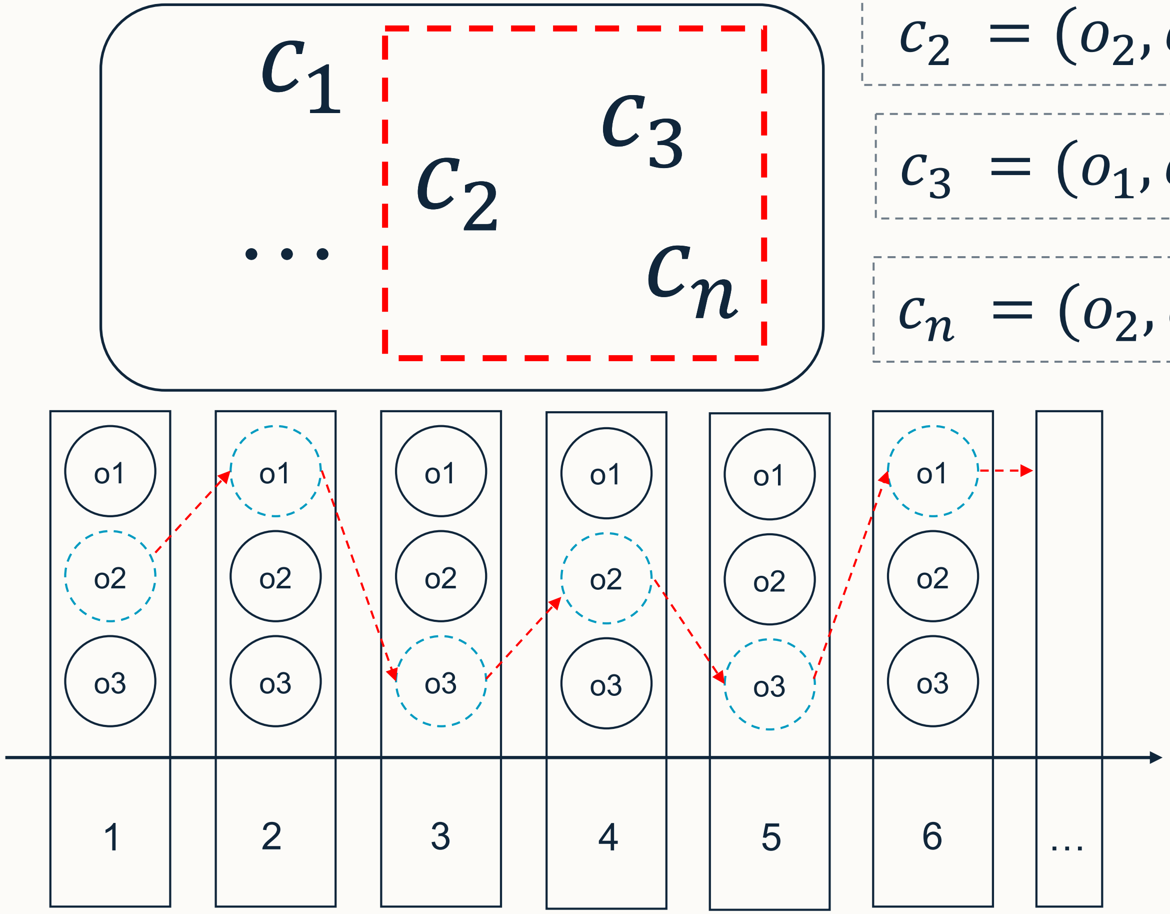


[2] Meng, W., & Qu, R. (2021). Automated design of search algorithms: Learning on algorithmic components. *Expert Systems with Applications*, *185*, 115493.

Automated Composition with AutoGCOP

Automated composition based on GCOP





$c_2 = (o_2, o_1, o_3, o_2, o_3, \dots)$

$c_3 = (o_1, o_1, o_2, o_2, o_3, \dots)$

$c_n = (o_2, o_1, o_3, o_1, o_4, \dots)$

Sequential Relations

How to learn?
What to learn?

ML for Automated Algorithm Composition

Different learning perspectives – to select basic o_i

Online learning

Learn individual performance vs. transition performance of basic o_i

→ **Task: Reinforcement learning (RL)**

Method: RL supported by Markov Chain (MC)

Offline learning

Learn good transitions of basic o_i to make predictions

→ **Task: Sequence classification**

Method: Long short-term memory (LSTM) and Transformer

Offline learning

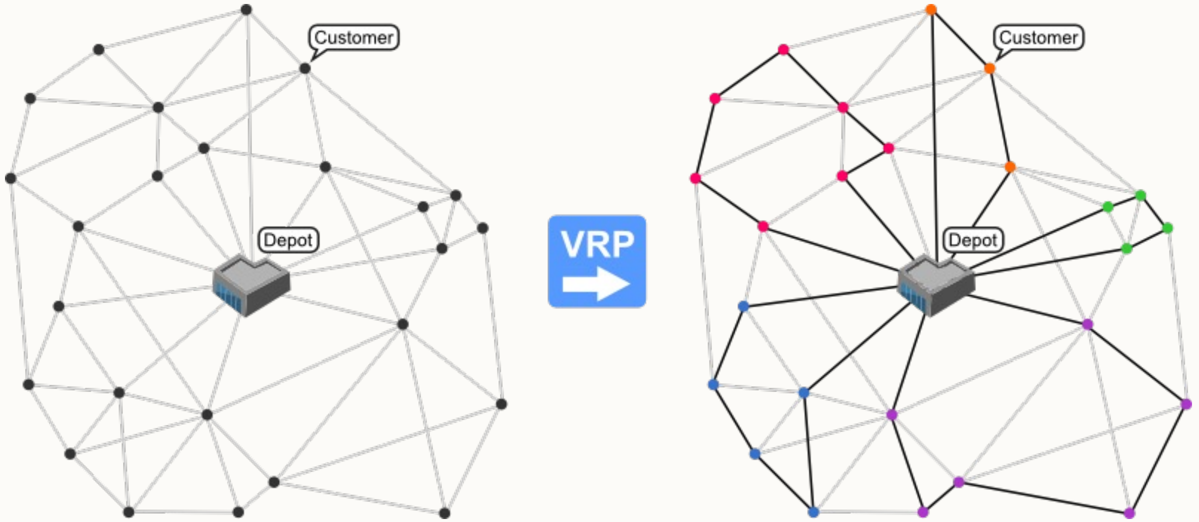
Extract rules from good transitions of basic o_i

→ **Task: Rule mining**

Method: Sequential rule mining

Research Testbed

Vehicle routing problems (VRPs)

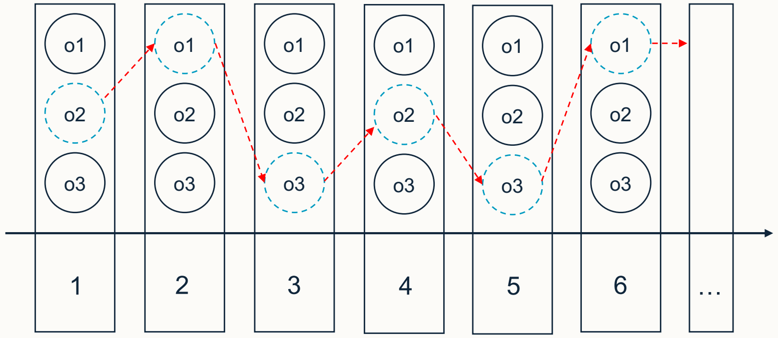


Operator	Description
o_{xchg}^{in}	Swap two customers in one route
o_{xchg}^{bw}	Swap two customers from different routes
o_{ins}^{in}	Move one customer to other position within the same route
o_{ins}^{bw}	Move one customer to other position of another route
o_{rr}	Remove 10% customers and reinsert them

Basic operators instantiated for VRPs [1].

[1] Qu, R., Kendall, G. and Pillay, N., 2020. The general combinatorial optimization problem: Towards automated algorithm design. *IEEE Computational Intelligence Magazine*, *15*(2), pp.14-23.

Method 1: Markov Chain

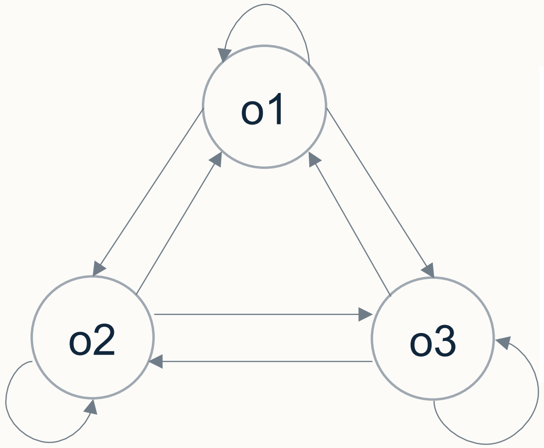


Online learning

Learn individual performance (IP) vs. **transition performance (TP)** of basic o_i

Task: Reinforcement learning (RL)

Method: Markov Chain (MC) enhanced by RL



	o_1	o_2	o_3
o_1	1	1	1
o_2	1	1	1
o_3	1	1	1

$o_2 \rightarrow o_1$ selected

s_2 is not better than s_{best}

	o_1	o_2	o_3
o_1	1	1	1
o_2	1	1	1
o_3	1	1	1

$o_1 \rightarrow o_3$ selected

s_3 is better than s_{best}

	o_1	o_2	o_3
o_1	1	1	2
o_2	1	1	1
o_3	1	1	1

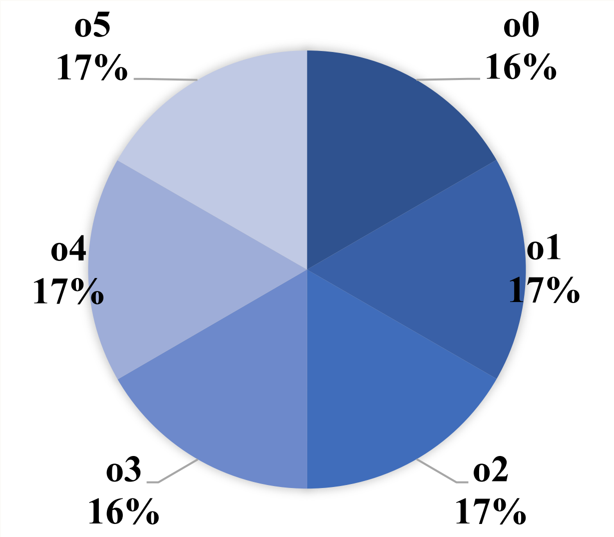
…

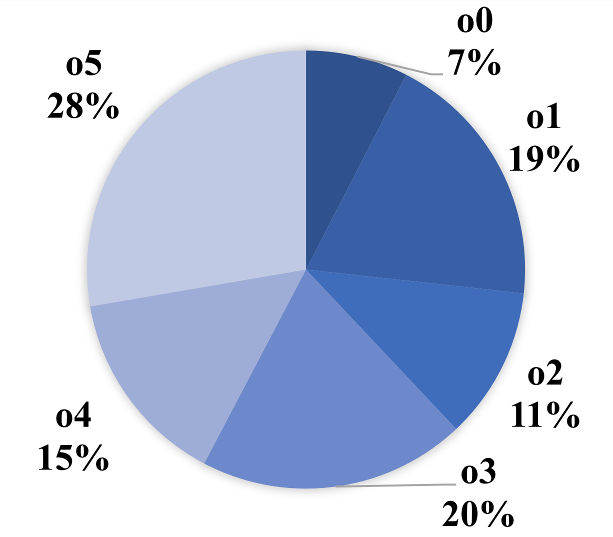
…

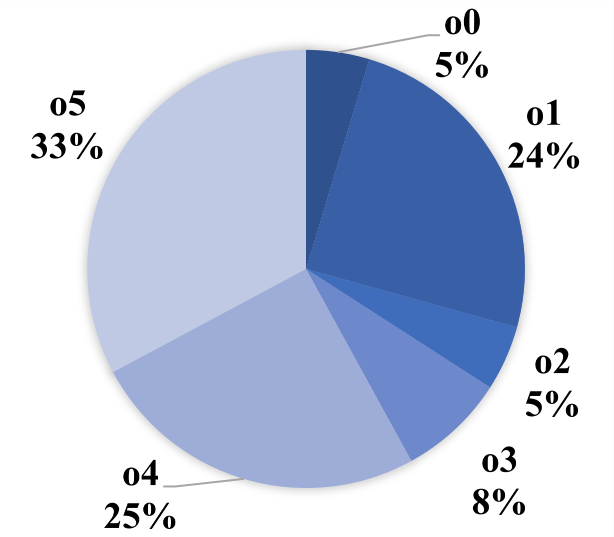
	o_1	o_2	o_3
o_1	1	1	2
o_2	1	1	1
o_3	2	2	1

[2] Meng, W., & Qu, R. (2021). Automated design of search algorithms: Learning on algorithmic components. *Expert Systems with Applications*, *185*, 115493.

Method 1: Markov Chain







Random method

No learning

Simple RL scheme

Learning the performance of individual operators

MC enhanced by RL

Learning the transitions between operators

[2] Meng, W., & Qu, R. (2021). Automated design of search algorithms: Learning on algorithmic components. *Expert Systems with Applications*, *185*, 115493.

Method 2: Sequence Classification

Offline learning

Learn good sequences of basic o_i to make predictions

Task: Sequence classification

Method: Long short-term memory (LSTM) and Transformer

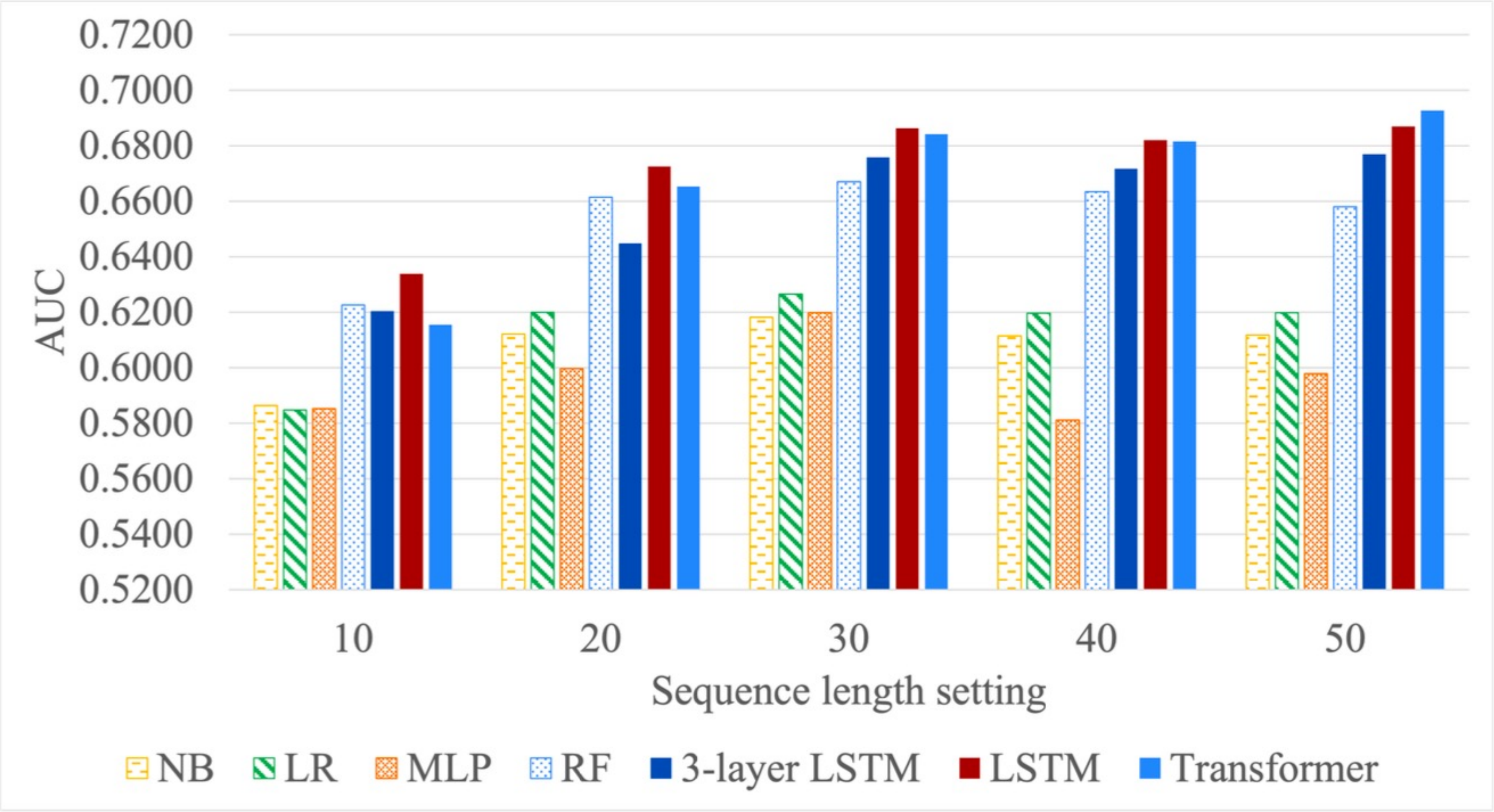
	Input	Output
ID	**Compositions of basic o_i**	**o_i to apply next**
1	$\{o_2\}, \{o_1\}, \{o_3\}, \{o_2\}, \{o_3\}$	$\{o_1\}$
2	$\{o_1\}, \{o_1\}, \{o_2\}, \{o_2\}, \{o_1\}$	$\{o_2\}$
…	…	

- **Search stage**: Index of iteration of the sequence
- **Operator features**: ID, operation type, involved routes, performance – solution quality change etc.
- **Instance features**: Vehicle capacity, customer distribution, time window density and width etc.

[3] Meng, W., & Qu, R. (2024). Automated design of local search algorithms: Predicting algorithmic components with LSTM. *Expert Systems with Applications*, *237*, 121431.

Method 2: Some Key Findings

Figure 5.6: The comparison of learning models in terms of the AUC performance.



- New task: sequence classification
- New models: LSTM and Transformer
- Key features: Search stage and instance features

[3] Meng, W., & Qu, R. (2024). Automated design of local search algorithms: Predicting algorithmic components with LSTM. *Expert Systems with Applications*, *237*, 121431.

Method 3: Sequence Rule Mining

Offline learning

Extract rules from good sequences of basic o_i

Task: Rule mining

Method: Sequential rule mining

Run ID	Compositions of basic operator o_i
1	$\{o_2\}, \{o_1\}, \{o_3\}, \{o_2\}, \{o_3\}, \{o_1\}, \ldots$
2	$\{o_1\}, \{o_1\}, \{o_2\}, \{o_2\}, \{o_1\}, \{o_2\}, \ldots$
…	…

ID	Sequences with length l
seq1	$\{o_3\}, \boxed{\{o_2\}, \{o_3\},} \{o_1\}, \{o_1\}$
…	…

[4] Meng, W., & Qu, R. (2023, July). Sequential Rule Mining for Automated Design of Meta-heuristics. In *Proceedings of the Companion Conference on Genetic and Evolutionary Computation* (pp. 1727-1735).

Method 3: Some Key Findings

Top 10 sequential rules for automated composition

Rules	sup:	conf:
$o^{bw}_{xchg} \rightarrow o_{rr}$	**1132**	**0.60**
$o^{in}_{ins} \rightarrow o_{rr}$	**1134**	**0.59**
$o^{in}_{xchg} \rightarrow o_{rr}$	**1111**	**0.57**
$o^{bw}_{xchg} \rightarrow o^{bw}_{ins}$	**1018**	**0.54**
$o^{in}_{xchg} \rightarrow o^{bw}_{ins}$	**1050**	**0.53**
$o^{in}_{ins} \rightarrow o^{bw}_{ins}$	**990**	**0.51**
$o^{bw}_{ins} \rightarrow o_{rr}$	**1198**	**0.51**
$o_{rr} \rightarrow o^{bw}_{ins}$	**1005**	**0.41**
$o^{bw}_{ins} \rightarrow o^{in}_{xchg}$	735	0.31
$o^{bw}_{ins} \rightarrow o^{in}_{ins}$	715	0.30

Instances	Best-known solutions in the literature	RN-GCOP AVG	SeqRuleGCOP AVG
C103	10,828.06[23]	12,364.31	**12,042.12**
C203	3,591.17[23]	4,502.51	**4,296.84**
R107	11,104.66[25]	14,564.69	**14,544.92**
R208	2,726.82[20]	4,087.51	**4,074.72**
RC103	12,261.67[26]	**14,881.08**	15,216.38
RC203	4,049.62[6]	4,784.47	**4,595.81**

[4] Meng, W., & Qu, R. (2023, July). Sequential Rule Mining for Automated Design of Meta-heuristics. In *Proceedings of the Companion Conference on Genetic and Evolutionary Computation* (pp. 1727-1735).

Method 3: Some Key Findings

Common sequential rules

- $X_o \longrightarrow Y_o$

Rules	sup:	conf:
$o^{bw}_{xchg} \rightarrow o_{rr}$	**1132**	**0.60**
$o^{in}_{ins} \rightarrow o_{rr}$	**1134**	**0.59**
$o^{in}_{xchg} \rightarrow o_{rr}$	**1111**	**0.57**
$o^{bw}_{xchg} \rightarrow o^{bw}_{ins}$	**1018**	**0.54**
$o^{in}_{xchg} \rightarrow o^{bw}_{ins}$	**1050**	**0.53**
$o^{in}_{ins} \rightarrow o^{bw}_{ins}$	**990**	**0.51**
$o^{bw}_{ins} \rightarrow o_{rr}$	**1198**	**0.51**
$o_{rr} \rightarrow o^{bw}_{ins}$	**1005**	**0.41**
$o^{bw}_{ins} \rightarrow o^{in}_{xchg}$	735	0.31
$o^{bw}_{ins} \rightarrow o^{in}_{ins}$	715	0.30

	Operator	Description
X_o	o^{in}_{xchg}	Swap two customers in one route
	o^{bw}_{xchg}	Swap two customers from different routes
	o^{in}_{ins}	Move one customer to other position within the same route
Y_o	o^{bw}_{ins}	Move one customer to other position of another route
	o_{rr}	Remove 10% customers and reinsert them

[4] Meng, W., & Qu, R. (2023, July). Sequential Rule Mining for Automated Design of Meta-heuristics. In *Proceedings of the Companion Conference on Genetic and Evolutionary Computation* (pp. 1727-1735).

Method 3: Some Key Findings

***Useful** and **interpretable** knowledge to support algorithm design*

Operator impact to optimisation

		Operator	Description	Impact to NV	Impact to TD
X_o	A_o^1	o_{xchg}^{in}	Swap two customers in one route	No	Small
		o_{xchg}^{bw}	Swap two customers from different routes	No	Small
		o_{ins}^{in}	Move one customer to other position within the same route	No	Small
Y_o	A_o^2	o_{ins}^{bw}	Move one customer to other position of another route	Small	Small
	A_o^3	o_{rr}	Remove 10% customers and reinsert them	Large	Large

[4] Meng, W., & Qu, R. (2023, July). Sequential Rule Mining for Automated Design of Meta-heuristics. In *Proceedings of the Companion Conference on Genetic and Evolutionary Computation* (pp. 1727-1735).

Conclusions

- A general AutoGCOP framework for automated composition of GCOP components for designing local search algorithms.
- Investigation of machine learning techniques from different learning perspectives:

Table 7.1: A summary of the main studies of different learning methods in the thesis.

Chapters	Learning tasks	Learning methods	Learning style	Knowledge type	Aim of learning
Chapter 4	RL	MC enhanced with RL	Online	Predictive	To forecast the next operator given the current operator
Chapter 5	Sequence classification	LSTM, Transformer	Offline	Predictive	To forecast the next operator given the previously applied operators and other information
Chapter 6	Rule inference	Sequential rule mining RL	Offline	Descriptive	To find frequent sequential rules between operators

Future Work Directions

How to learn

- Modelling AutoAD tasks as ML tasks
- Evaluating effectiveness and limitations

What to learn

- Decision-making in algorithm design: interconnection
- Uncovering hidden knowledge: interpretability

New testbed

- Other application domains

Link to thesis

References

[1] Qu, R., Kendall, G. and Pillay, N., 2020. The general combinatorial optimization problem: Towards automated algorithm design. *IEEE Computational Intelligence Magazine*, *15*(2), pp.14-23.

[2] Meng, W. and Qu, R., 2021. Automated design of search algorithms: Learning on algorithmic components. *Expert Systems with Applications*, *185*, p.115493.

[3] Meng, W. and Qu, R., 2024. Automated design of local search algorithms: Predicting algorithmic components with LSTM. *Expert Systems with Applications*, *237*, p.121431.

[4] Meng, W. and Qu, R., 2023, July. Sequential Rule Mining for Automated Design of Meta-heuristics. In *Proceedings of the Companion Conference on Genetic and Evolutionary Computation* (pp. 1727-1735).

[5] Yi, W., Qu, R., Jiao, L. and Niu, B., 2022. Automated design of metaheuristics using reinforcement learning within a novel general search framework. *IEEE Transactions on Evolutionary Computation*.

University of Nottingham
UK | CHINA | MALAYSIA

Thank you!

Weiyao Meng

University of Nottingham
Weiyao.meng2@Nottingham.ac.uk
https://weiyaomeng.github.io/

Weiyao Meng
Data Scientist (KTP Associate) | PhD in CS