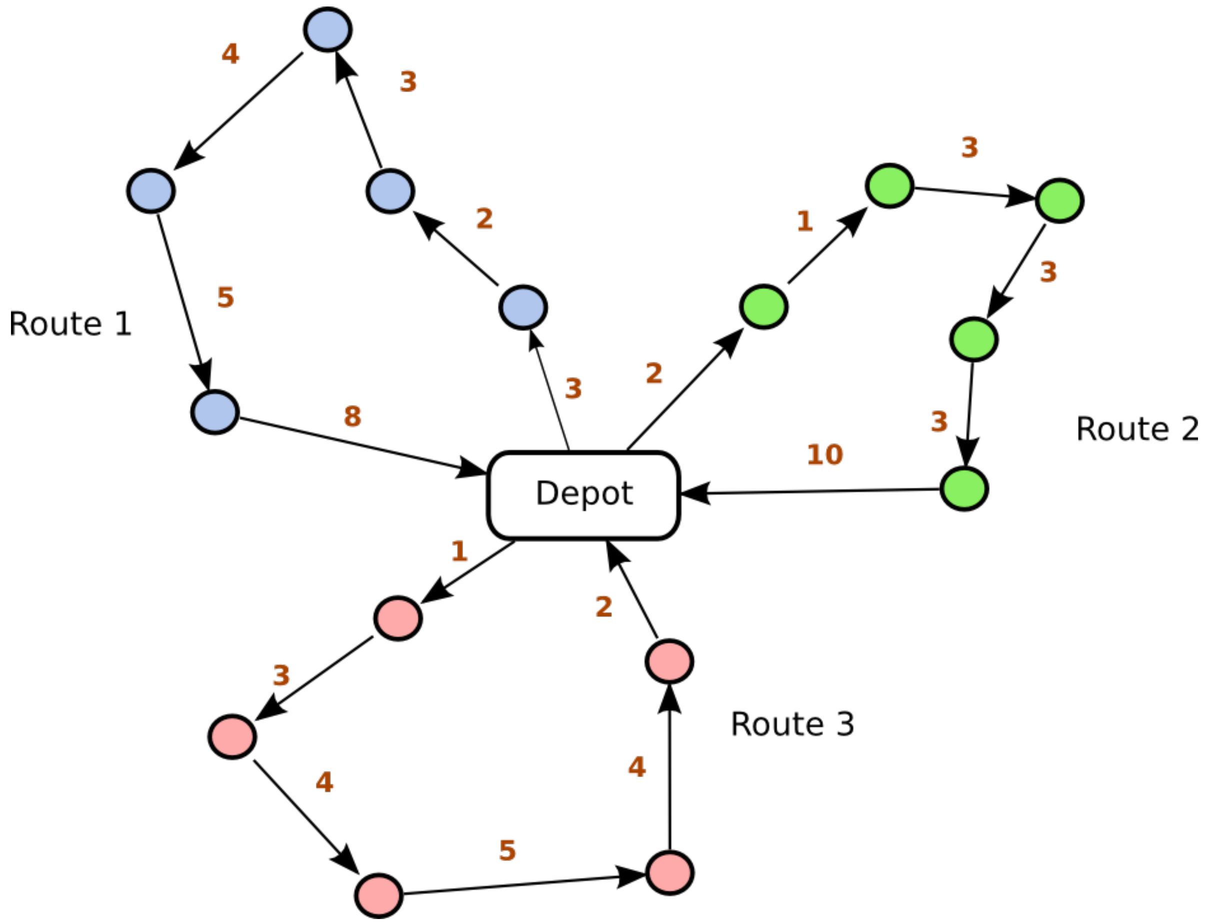
REINFORCEMENT LEARNING FOR SOLVING THE VEHICLE ROUTING PROBLEM



Presented for CS 885, by:
Wei Hu

UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS

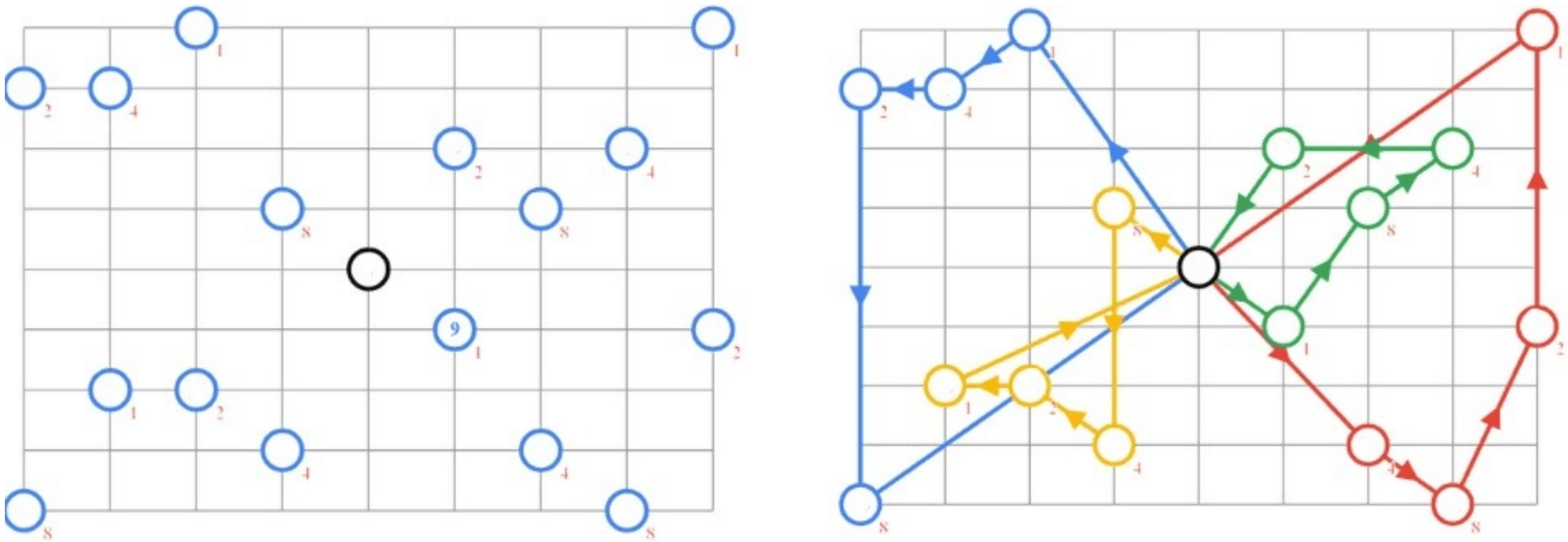
AGENDA

VEHICLE ROUTING PROBLEM (VRP)

- NP-Hard problem that has been studied for decades.
- Hand-crafted heuristics exists.
- Goal is to avoid needing “hand-engineered reasoning.”
- Phrased as an MDP
- Find solutions by increasing the probability of decoding desirable sequences.

THE BASIC VRP

1. One vehicle & one depot
2. Multiple customers
3. Must refuel
4. Find optimal set of routes
5. Minimize distance



THE PROBLEM SPACE

- **Why can’t we consider every instance separately?**
  - Millions of samples needed.
  - Bad runtime.
  - Not generalizable.
- Fix the distribution from which the problems are sampled!

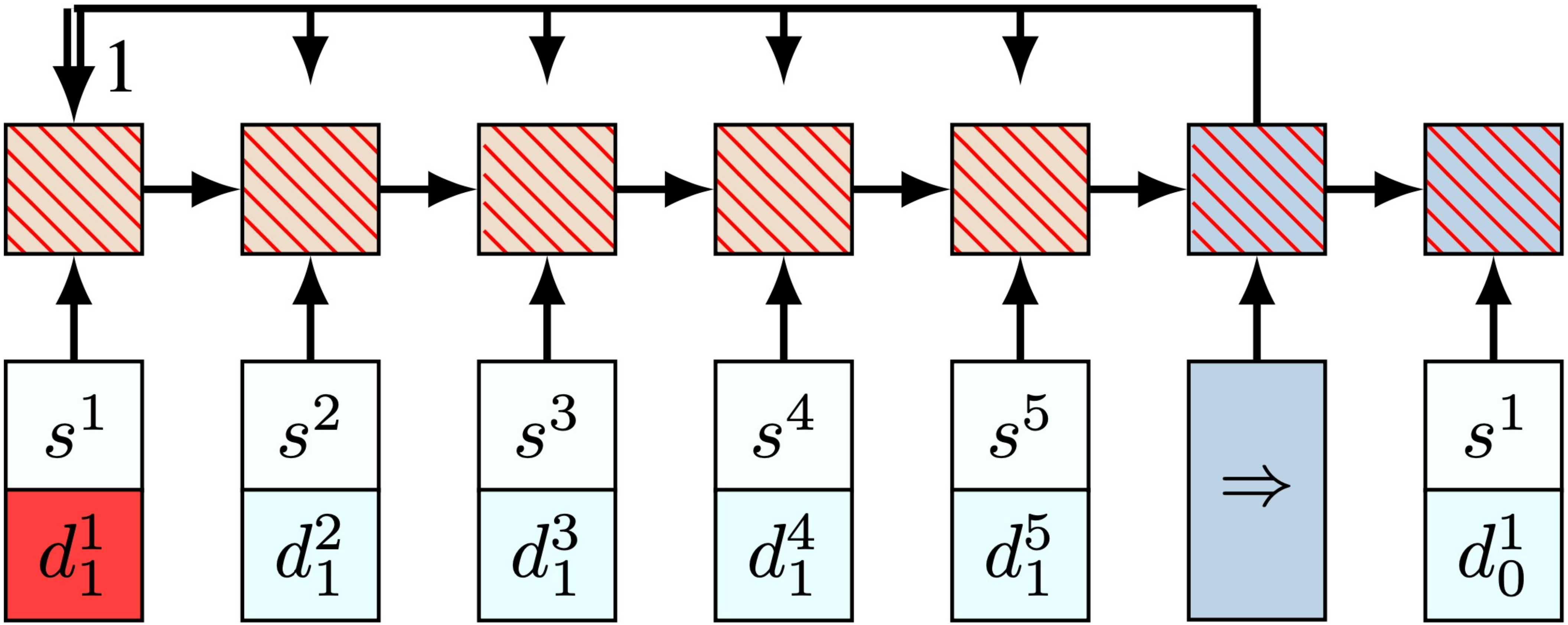
PREVIOUS APPROACHES

- Sequence-to-Sequence Models:
  - Two RNNs
  - Attention Mechanism
- Pointer Network
  - Inspired by Sequence-to-Sequence
  - Works well on Travelling Salesman (supervised learning)
- Neural Combinatorial Network (Bello et al.)
  - Modeled by Pointer Network
  - RL to optimize policy

PREVIOUS APPROACHES

- Bello et al. has previous work using Pointer networks on combinatorial optimization problems.
- However, VRP is not static.
- Pointer networks must be recalculated when there is new information.

POINTER NETWORK



PROPOSED APPROACH

- RNN + Attention used to keep track of visited nodes:
    - VRP consists of an unordered set of locations and demands.
    - No need for RNN encoder.
    - Use embedded inputs instead of RNN hidden states.
- Only requires reward calculation and feasibility verification.
- Robust to change & allows split deliveries.

NOTATION

$$X \doteq \{x^i, i = 1, \cdots, M\}$$

$$\{x_t^i \doteq (s^i, d_t^i), t = 0, 1, \cdots\}$$

$$(t = 0, 1, \cdots), \; y_{t+1}$$

NOTATION

$$Y = \{y_t, t = 0, ..., T\}$$

$$Y_t = \{y_0, \cdots, y_t\}$$

THE MODEL

$$P(Y|X_0) = \prod_{t=0}^{T} \pi(y_{t+1}|Y_t, X_t),$$

THE MODEL

$$P(Y|X_0) = \prod_{t=0}^{T} \pi(y_{t+1}|Y_t, X_t),$$

$$X_{t+1} = f(y_{t+1}, X_t)$$

THE MODEL

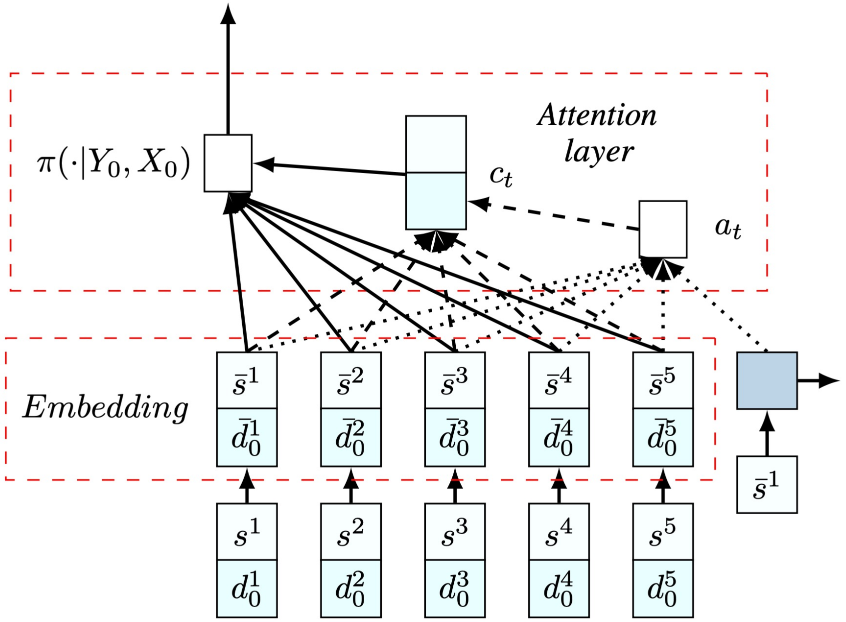
$$P(Y|X_0) = \prod_{t=0}^{T} \pi(y_{t+1}|Y_t, X_t),$$

$$X_{t+1} = f(y_{t+1}, X_t)$$

$$\pi(\cdot|Y_t, X_t) = \text{softmax}(g(h_t, X_t)),$$

NEURAL NETWORK ARCHITECTURE

1. Embedding maps to high dimensional vector space.
2. RNN decoder produces probability distribution.



UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS

ATTENTION MECHANISM

$$a_t = a_t(\bar{x}_t, h_t) = \text{softmax}\left(u_t\right), \qquad \text{where } u_t^i = v_a^T \tanh\left(W_a[\bar{x}_t^i; h_t]\right).$$

ATTENTION MECHANISM

$$a_t = a_t(\bar{x}_t, h_t) = \text{softmax}\left(u_t\right), \qquad \text{where } u_t^i = v_a^T \tanh\left(W_a[\bar{x}_t^i; h_t]\right).$$

$$c_t = \sum_{i=1}^{M} a_t^i \bar{x}_t^i,$$

ATTENTION MECHANISM

$$a_t = a_t(\bar{x}_t, h_t) = \text{softmax}\left(u_t\right), \qquad \text{where } u_t^i = v_a^T \tanh\left(W_a[\bar{x}_t^i; h_t]\right).$$

$$c_t = \sum_{i=1}^{M} a_t^i \bar{x}_t^i,$$

$$\pi(\cdot|Y_t, X_t) = \text{softmax}(\tilde{u}_t), \qquad \text{where } \tilde{u}_t^i = v_c^T \tanh\left(W_c[\bar{x}_t^i; c_t]\right).$$

TRAINING PROCESS

- Policy Gradient
  - Actor
  - Critic
- … just like we covered in class!

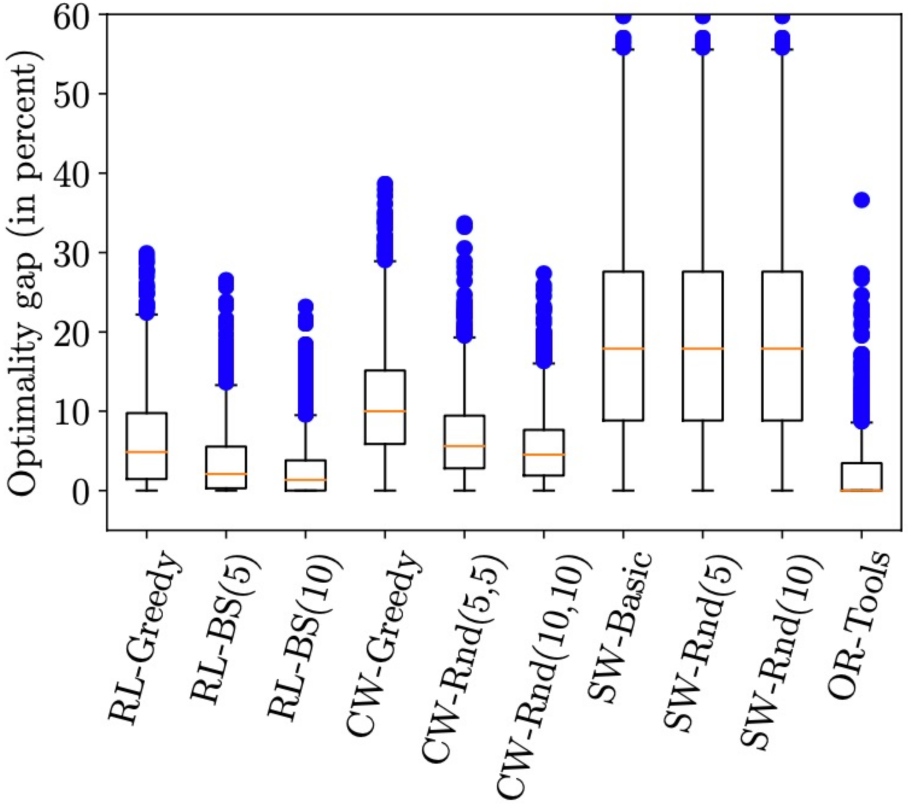
ENVIRONMENT

- Capacitated Vehicle Routing Problem
  - Locations are from random uniform on the unit square
  - Demand is in {1, ..., 9}
- At each time step, the algorithm outputs one of the customer nodes or the depot, which will be visited next.

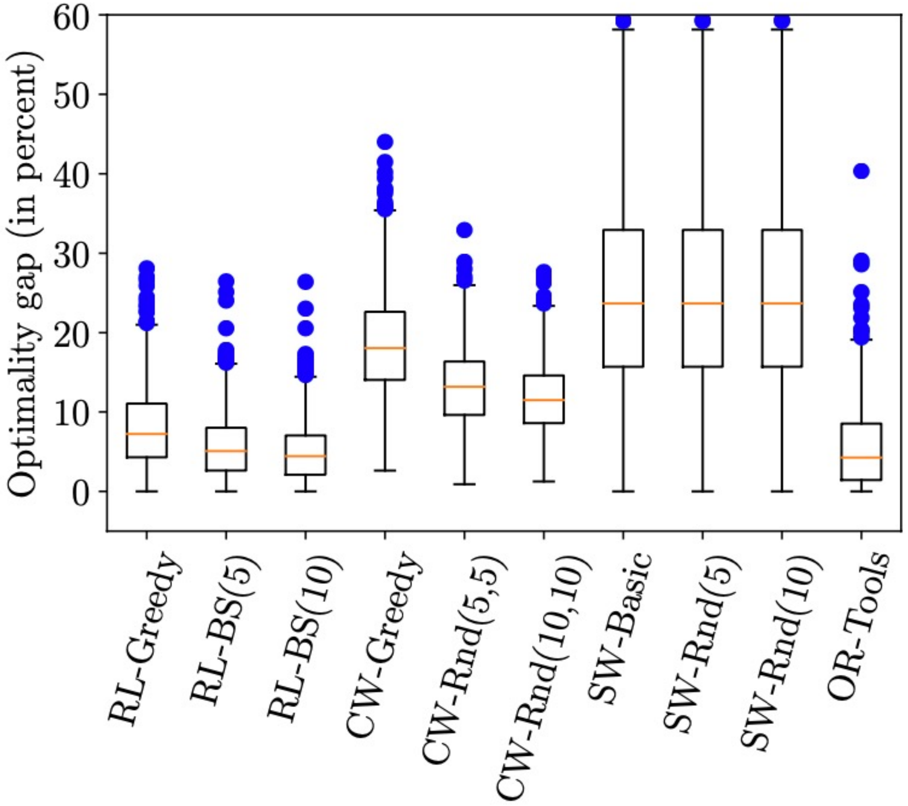
$$d_{t+1}^{i} = \max(0, d_t^i - l_t), \quad d_{t+1}^{k} = d_t^k \ \text{ for } k \neq i, \text{and} \quad l_{t+1} = \max(0, l_t - d_t^i)$$

- Decoders:
  - Greedy
  - Beam Search
- Masking

RESULTS



(a) Comparison for VRP10



(b) Comparison for VRP20

UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS

RESULTS

	RL-Greedy	RL-BS(5)	RL-BS(10)	CW-Greedy	CW-Rnd(5,5)	CW-Rnd(10,10)	SW-Basic	SW-Rnd(5)	SW-Rnd(10)	OR-Tools
RL-Greedy		12.2	7.2	99.4	97.2	96.3	97.9	97.9	97.9	41.5
RL-BS(5)	85.8		12.5	99.7	99.0	98.7	99.1	99.1	99.1	54.6
RL-BS(10)	91.9	57.7		99.8	99.4	99.2	99.3	99.3	99.3	60.2
CW-Greedy	0.6	0.3	0.2		0.0	0.0	68.9	68.9	68.9	1.0
CW-Rnd(5,5)	2.8	1.0	0.6	92.2		30.4	84.5	84.5	84.5	3.5
CW-Rnd(10,10)	3.7	1.3	0.8	97.5	68.0		86.8	86.8	86.8	4.7
SW-Basic	2.1	0.9	0.7	31.1	15.5	13.2		0.0	0.0	1.4
SW-Rnd(5)	2.1	0.9	0.7	31.1	15.5	13.2	0.0		0.0	1.4
SW-Rnd(10)	2.1	0.9	0.7	31.1	15.5	13.2	0.0	0.0		1.4
OR-Tools	58.5	45.4	39.8	99.0	96.5	95.3	98.6	98.6	98.6	

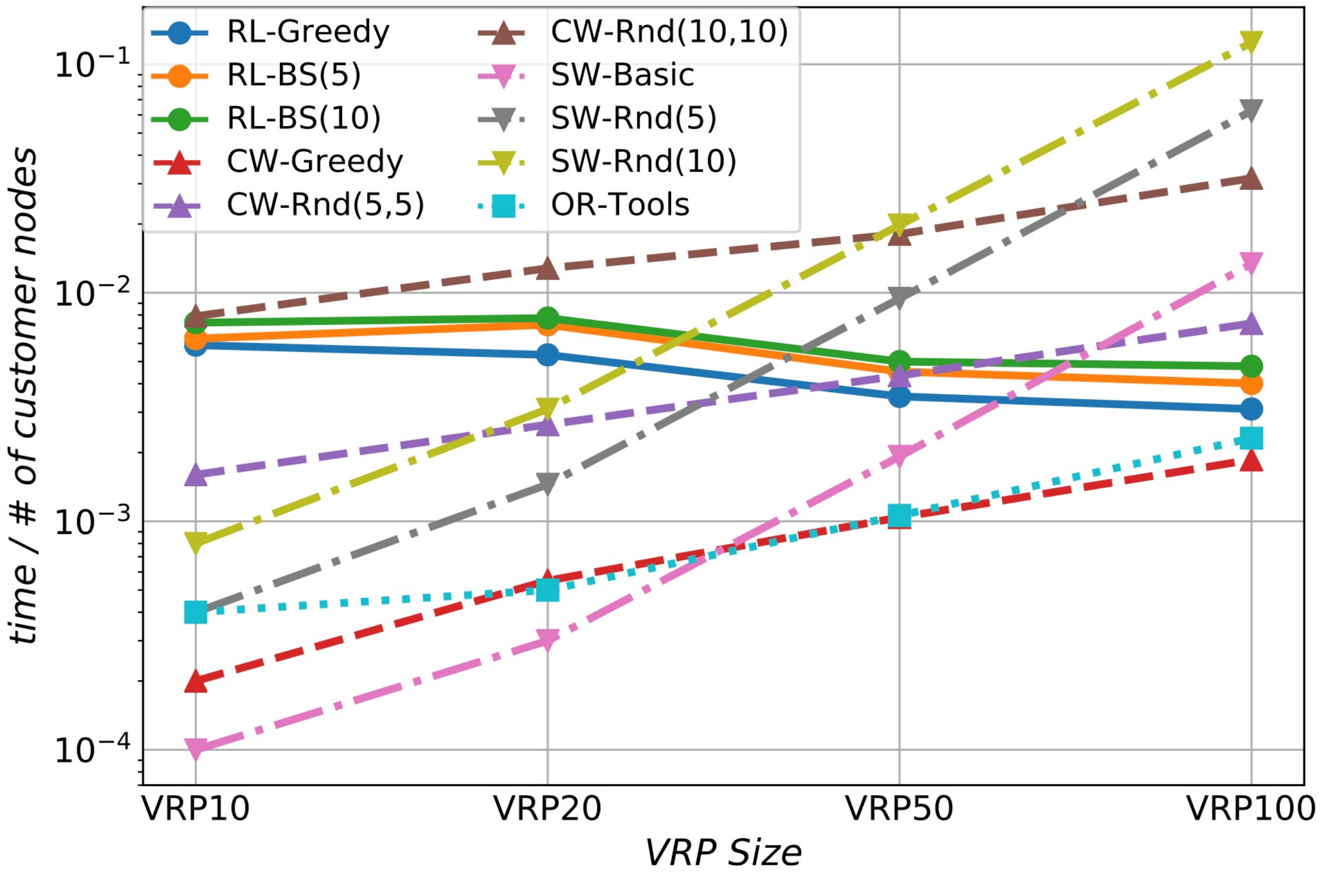
(c) Comparison for VRP50

RESULTS

	RL-Greedy	RL-BS(5)	RL-BS(10)	CW-Greedy	CW-Rnd(5,5)	CW-Rnd(10,10)	SW-Basic	SW-Rnd(5)	SW-Rnd(10)	OR-Tools
RL-Greedy		25.4	20.8	99.9	99.8	99.7	99.5	99.5	99.5	44.4
RL-BS(5)	74.4		35.3	100.0	100.0	99.9	100.0	100.0	100.0	56.6
RL-BS(10)	79.2	61.6		100.0	100.0	100.0	99.8	99.8	99.8	62.2
CW-Greedy	0.1	0.0	0.0		0.0	0.0	65.2	65.2	65.2	0.0
CW-Rnd(5,5)	0.2	0.0	0.0	92.6		32.7	82.0	82.0	82.0	0.7
CW-Rnd(10,10)	0.3	0.1	0.0	97.2	65.8		85.4	85.4	85.4	0.8
SW-Basic	0.5	0.0	0.2	34.8	18.0	14.6		0.0	0.0	0.0
SW-Rnd(5)	0.5	0.0	0.2	34.8	18.0	14.6	0.0		0.0	0.0
SW-Rnd(10)	0.5	0.0	0.2	34.8	18.0	14.6	0.0	0.0		0.0
OR-Tools	55.6	43.4	37.8	100.0	99.3	99.2	100.0	100.0	100.0	

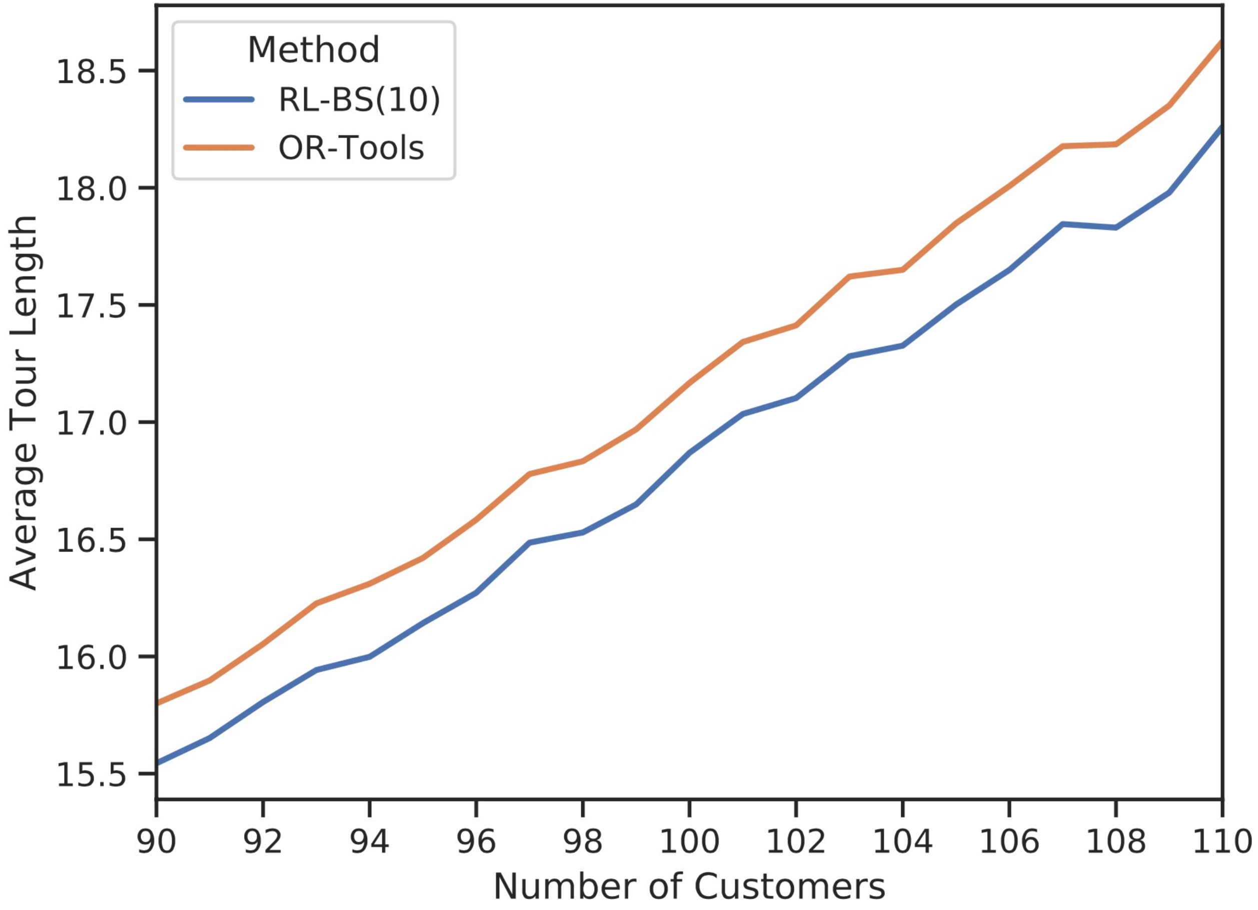
(d) Comparison for VRP100

RESULTS



RL-Greedy
RL-BS(5)
RL-BS(10)
CW-Greedy
CW-Rnd(5,5)
CW-Rnd(10,10)
SW-Basic
SW-Rnd(5)
SW-Rnd(10)
OR-Tools
10^{-1}
10^{-2}
10^{-3}
10^{-4}
time / # of customer nodes
VRP10
VRP20
VRP50
VRP100
VRP Size

RESULTS


Method
RL-BS(10)
OR-Tools
Average Tour Length
18.5
18.0
17.5
17.0
16.5
16.0
15.5
90
92
94
96
98
100
102
104
106
108
110
Number of Customers

DISCUSSION

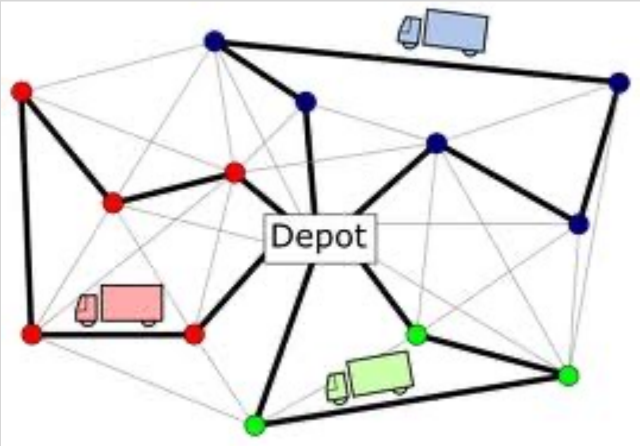
- When the training data is close to the test data, the RL approach delivers near-optimal performance.
- **When are the classic heuristics still preferred?**
- **Is there still utility for the integer programming / dynamic programming approaches?**

GENERALIZED VRP

Multiple Depots

Additional Constraints

Multiple Vehicles



Conclusion

- Competitive with state-of-the-art heuristics.
- Can be deployed in practice.
- Scales well with the problem size.
- Handles stochasticity.
- Approach can be applied to other combinatorial optimization problems!

UNIVERSITY OF WATERLOO

FACULTY OF MATHEMATICS

Thank You

Reach me at wei.hu1@uwaterloo.ca