The 2nd International Workshop on Network Intelligence
in conjunction with IEEE INFOCOM 2019
29 April – 2 May 2019 – Paris, France

DeepViNE: Virtual Network Embedding with Deep Reinforcement Learning

M. Dolati, S. B. Hassanpour, M. Ghaderi, A. Khonsari

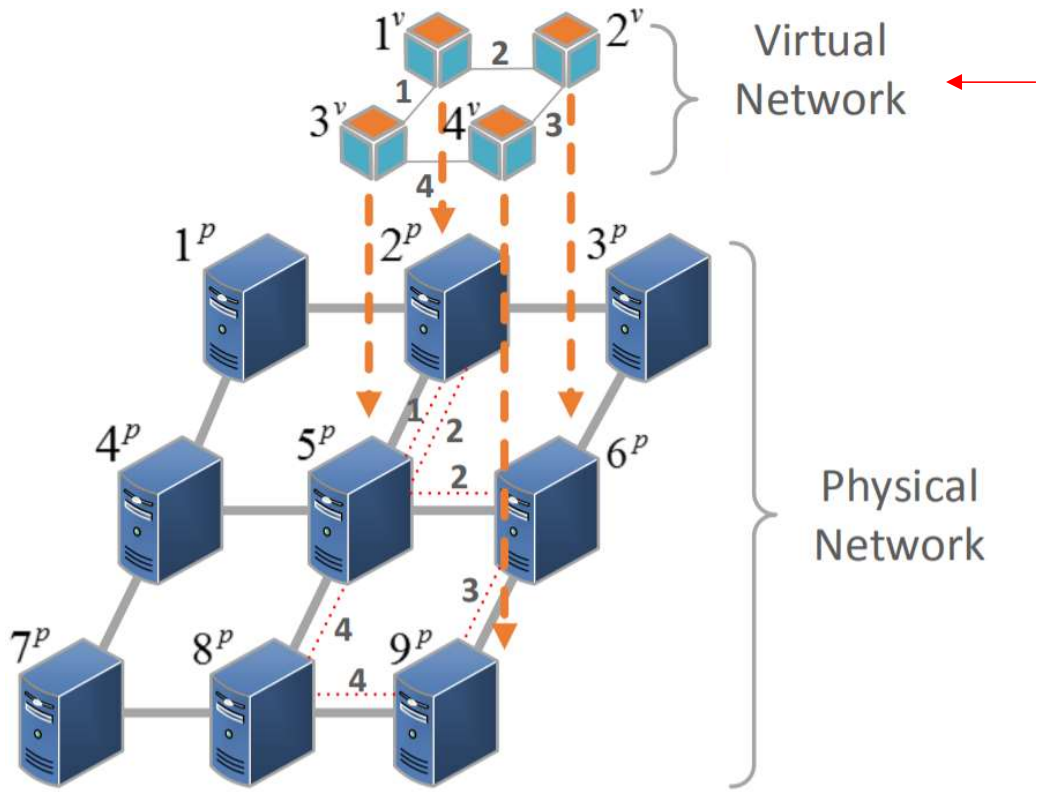
Presenter: Mahdi Dolati

mahdidolati@ut.ac.ir

University of Tehran and University of Calgary

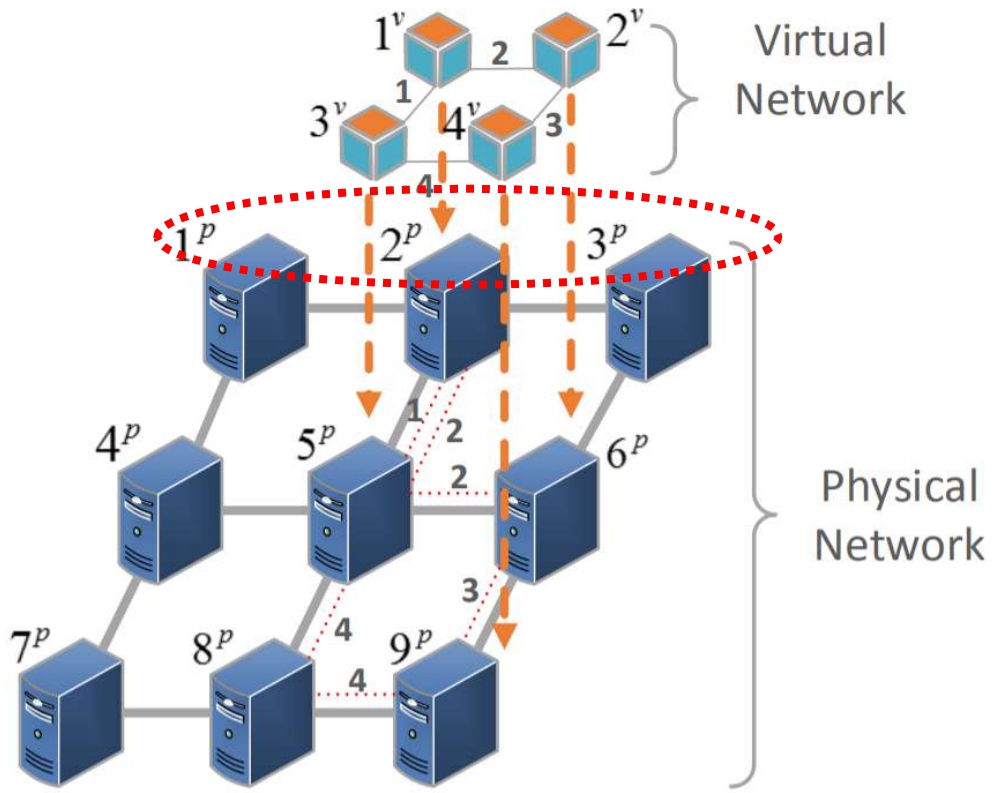
Virtual Network Embedding

- Network Virtualization
  - Is one of the key technologies of future networks.
  - Allows multiple **virtual networks (VN)** to coexist on the same **physical network (PN)**.



Virtual Network Embedding

- Node mapping
  - CPU demand
- Link mapping
  - Bandwidth demand
- Request arrives
  - Little information about future
  - Long-term objective

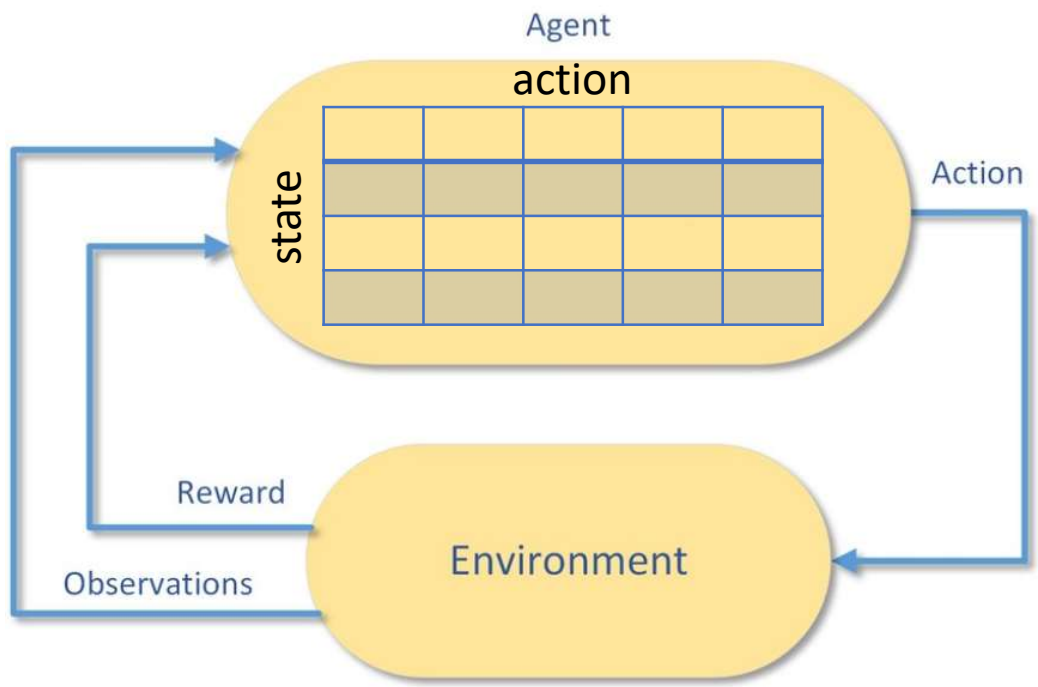


Options

- Optimal
- Heuristic
- Online-algorithm-based
- Learning-based

Reinforcement Learning

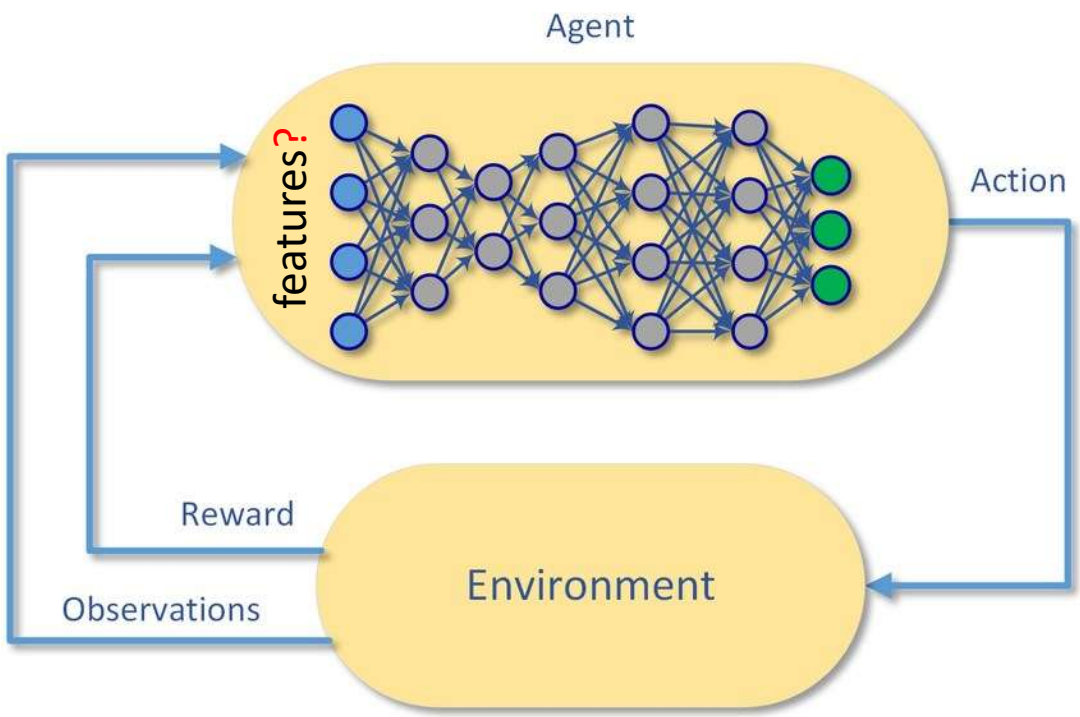
- Learn from interaction -> Brings adaptability
- Curse of dimensionality



$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\Big(r_t + \gamma \max_a Q(s_{t+1}, a) - Q(s_t, a_t)\Big)$$

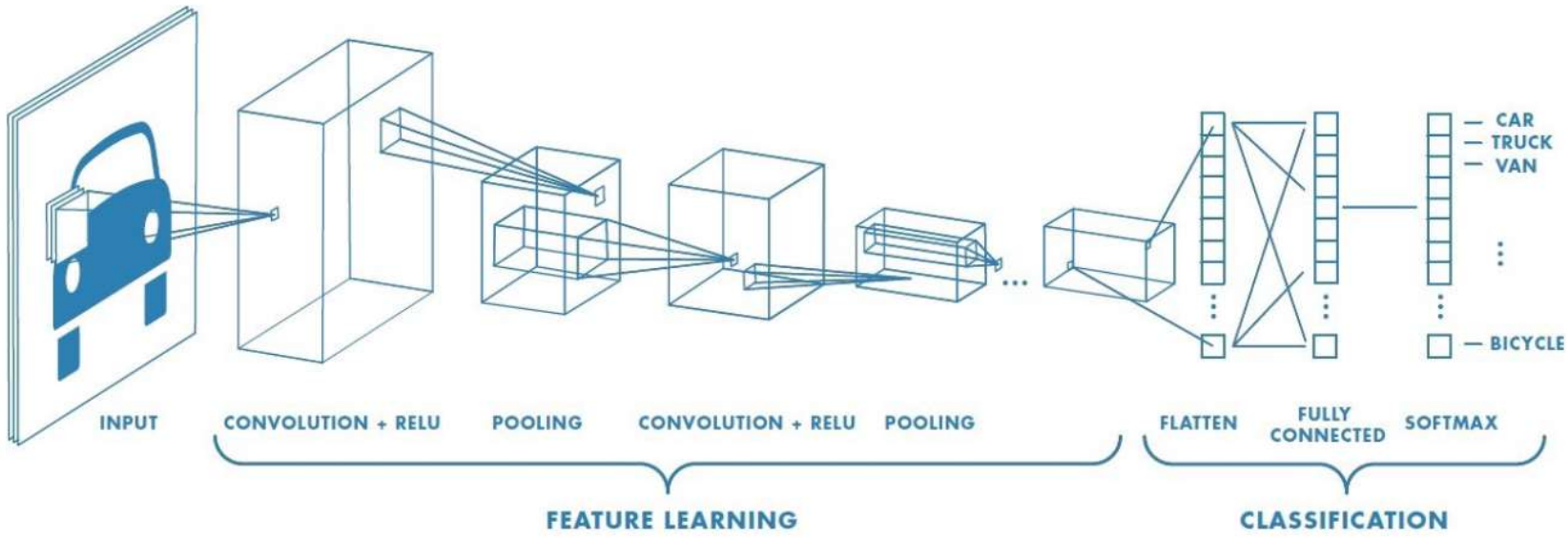
Reinforcement Learning

- Learn from interaction -> Brings adaptability
- Curse of dimensionality -> Use neural networks



$$L_\theta = \mathbb{E}\left[r_t + \gamma \max_a Q_\theta(s_{t+1}, a) - Q_\theta(s_t, a_t)\right]$$

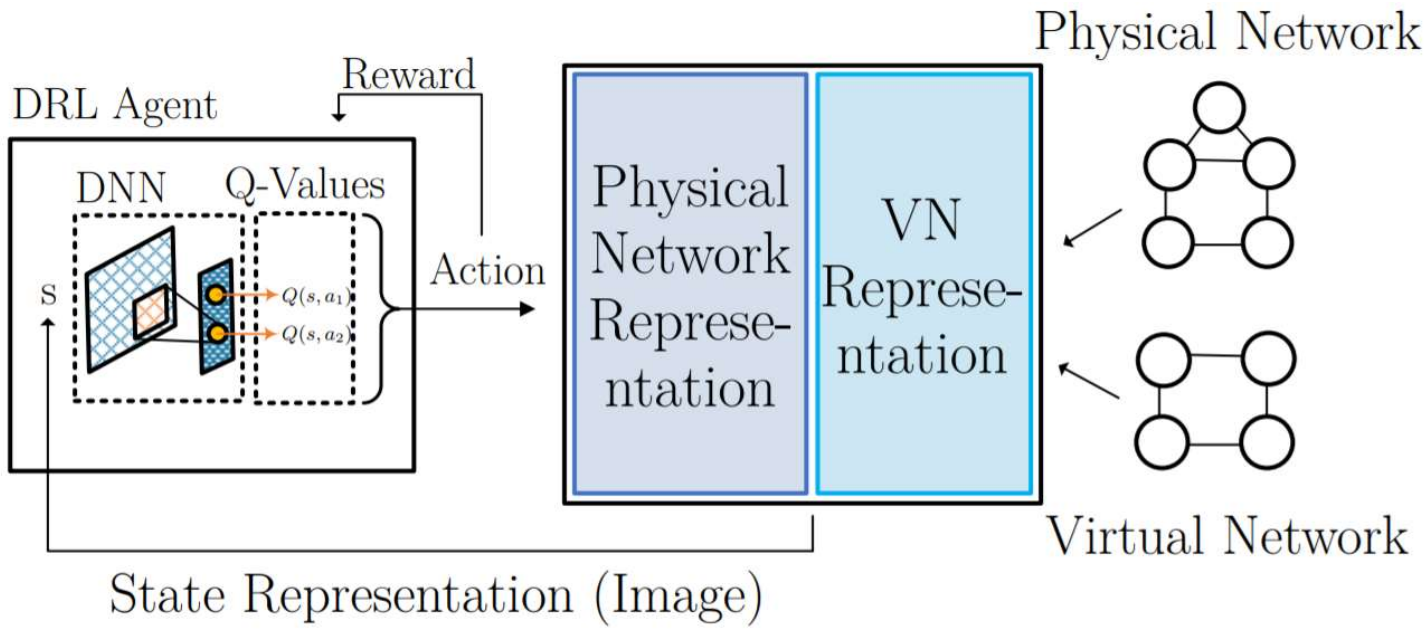
Image Source: Mohammadi, Mehdi & Al-Fuqaha, Ala. (2018). Enabling Cognitive Smart Cities Using Big Data and Machine Learning: Approaches and Challenges. IEEE Communications Magazine. 56. 94-101.



Deep Learning

- Convolutional neural network
- With origins in image recognition area
  - Automates feature extraction
  - End-to-end learning
- Over-fitting

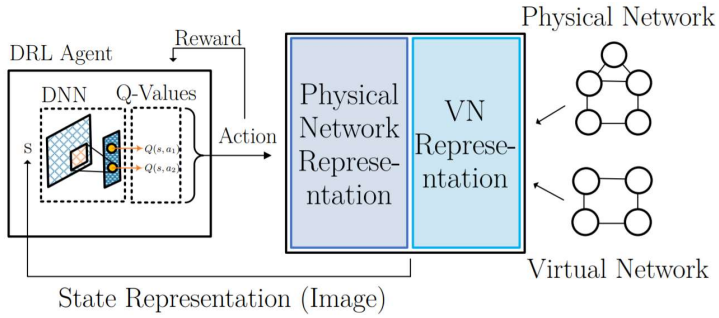
Image Source: https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53



Architecture

- Deep Reinforcement Learning (DRL)

Two Challenging Problems



Action Space

Affects learning efficiency

Image Representation

Perceivable by a DNN

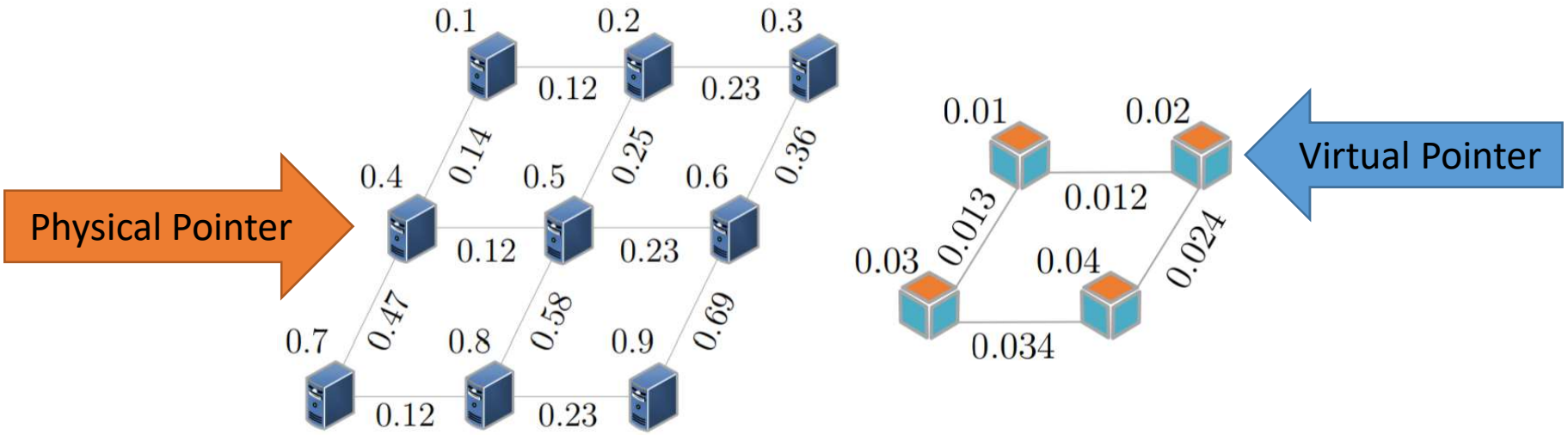
Algorithm Realization

Action definition

Image construction

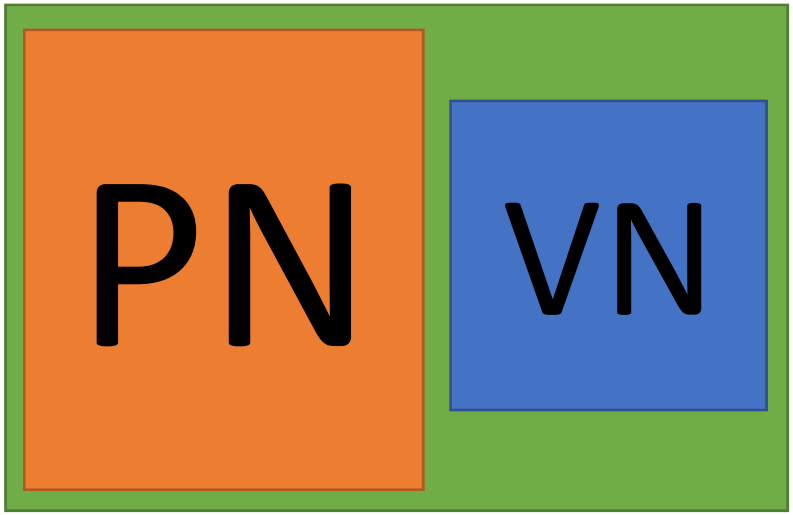
Reward Signal

Neural network architecture



Action Definition

- Exploration and Solution
  - Move a pointer: UP, DOWN, LEFT, RIGHT (8 actions)
  - Embed (1 action)



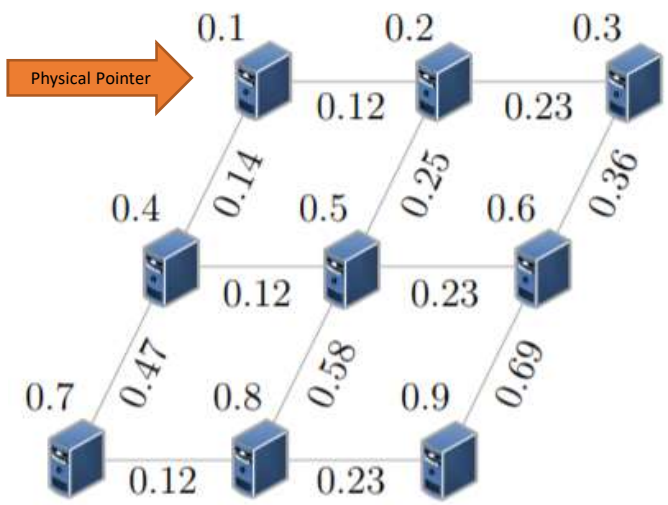
Image

Image Construction

Encode the input as a 2D matrix

Image Construction

PN Encoding

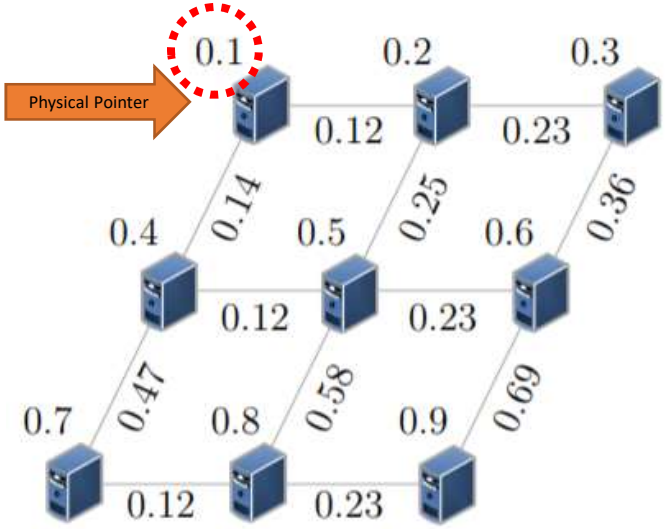


Available CPU
Physical Pointer
Embedded Virtual Nodes
Available Bandwidth

0.1	1	0000	0.12	0.2	0	0001	0.23	0.3	0	0000
0.14				0.25				0.36		
0.4	0	0000	0.45	0.5	0	0100	0.56	0.6	0	0010
0.47				0.58				0.69		
0.7	0	0000	0.78	0.8	0	0000	0.89	0.9	0	1000

Image Construction

PN Encoding – CPU

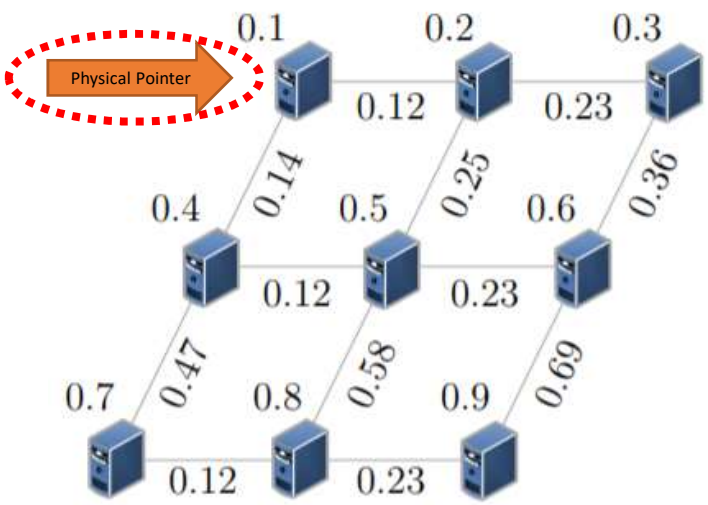


Available CPU
Physical Pointer
Embedded Virtual Nodes
Available Bandwidth

0.1	1	0000	0.12	0.2	0	0001	0.23	0.3	0	0000
0.14				0.25				0.36		
0.4	0	0000	0.45	0.5	0	0100	0.56	0.6	0	0010
0.47				0.58				0.69		
0.7	0	0000	0.78	0.8	0	0000	0.89	0.9	0	1000

Image Construction

PN Encoding – Pointer

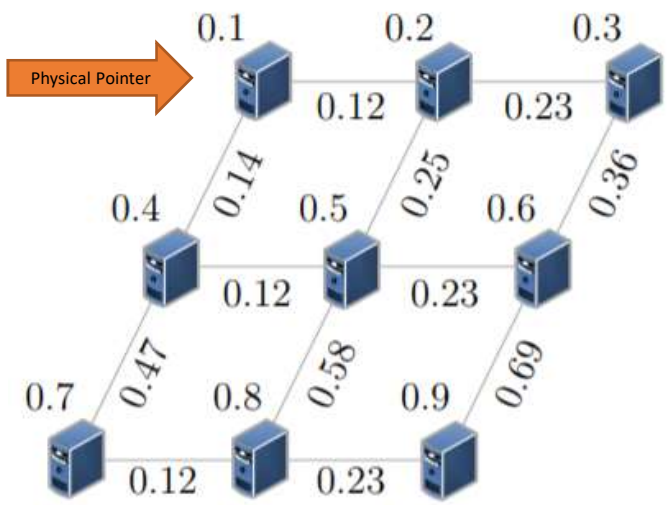


Available CPU
Physical Pointer
Embedded Virtual Nodes
Available Bandwidth

0.1	1	0000	0.12	0.2	0	0001	0.23	0.3	0	0000
0.14				0.25				0.36		
0.4	0	0000	0.45	0.5	0	0100	0.56	0.6	0	0010
0.47				0.58				0.69		
0.7	0	0000	0.78	0.8	0	0000	0.89	0.9	0	1000

Image Construction

PN Encoding – Embedded

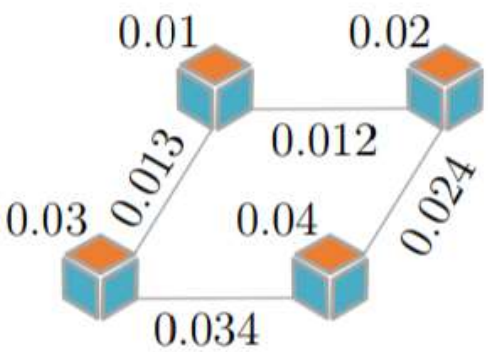


Available CPU
Physical Pointer
Embedded Virtual Nodes
Available Bandwidth

0.1	1	0000	0.12	0.2	0	0001	0.23	0.3	0	0000
0.14				0.25				0.36		
0.4	0	0000	0.45	0.5	0	0100	0.56	0.6	0	0010
0.47				0.58				0.69		
0.7	0	0000	0.78	0.8	0	0000	0.89	0.9	0	1000

Image Construction

VN Encoding

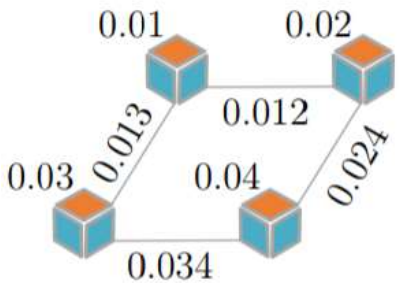


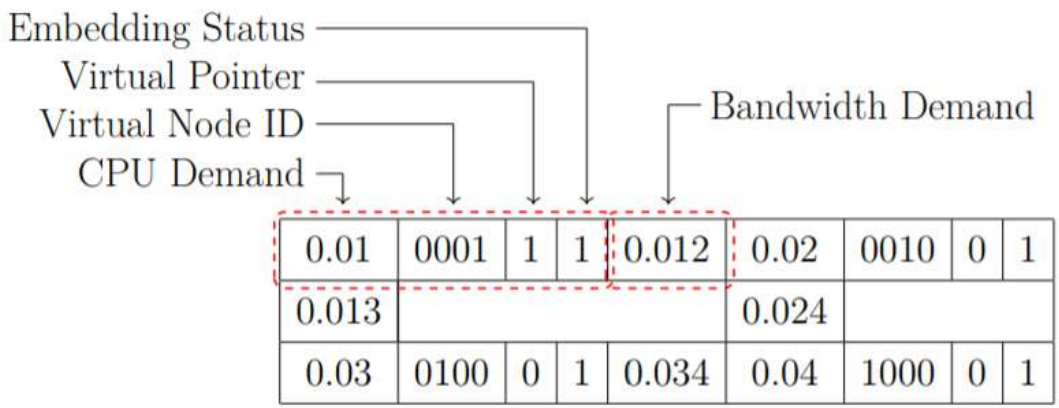
Embedding Status
Virtual Pointer
Virtual Node ID
CPU Demand
Bandwidth Demand

0.01	0001	1	1	0.012	0.02	0010	0	1
0.013					0.024			
0.03	0100	0	1	0.034	0.04	1000	0	1

Image Construction

VN Encoding





One-hot encoding scheme for IDs

- Unique
- Easy to represent the union
  - 0001 + 0010 -> 0011

Take advantage of channels

Use three channels: Each channel represents the image from a specific viewpoint, e.g., an RGB image has three channels to represent Red, Green, and Blue.

Channel 2: When a virtual node or link is embedded we replace its corresponding encoding in the VN representation with an array of appropriate number of 1's.

Channel 3: When remaining capacity of a physical node or link is not enough to embed any virtual node and link we show it by an array of 0's with appropriate length.

Reward Signal

Complete embedding: 1

Embedding a single virtual node: 0.01

Violating the resource constraints: −0.01

- The episode is terminated

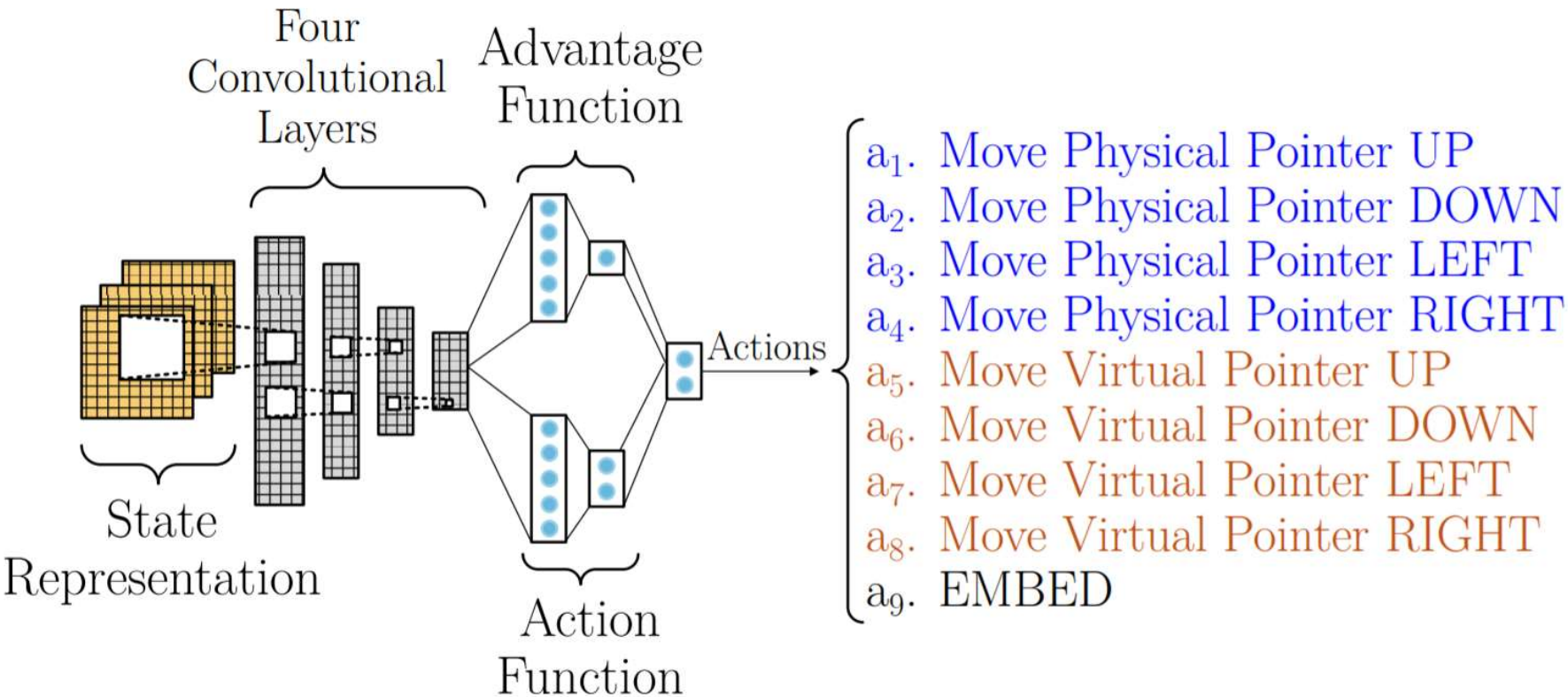
Moving the virtual and physical pointers: 0

- Allows the agent to explore

The cumulative reward is maximized when the maximum number of VNs are embedded.

Neural Network Architecture

Dueling technique



Evaluation

PN has 25 nodes

Each VN has 9 nodes

Physical Resources from [50, 100]

Virtual Demand from [1, 10]

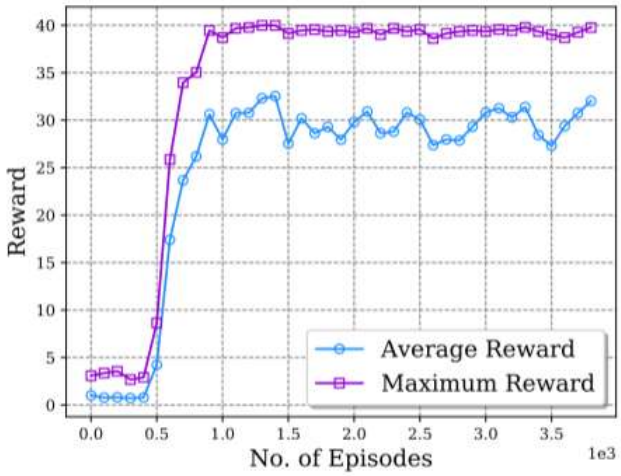
Episodic Operation:

- Each episode is a sequence of 800 iterations of action selection.
- VNs arrive sequentially
- On average 40 VNs
- End of episode: All VNs leave the system

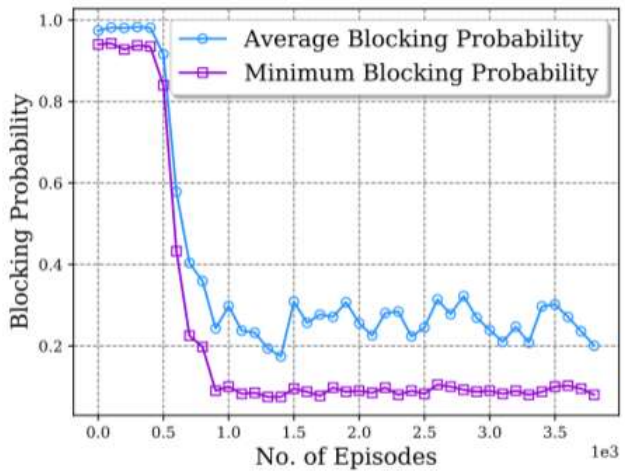
Tensorflow

Competing approaches

- BestFit, FirstFit, Random, NeuroViNE, GRC



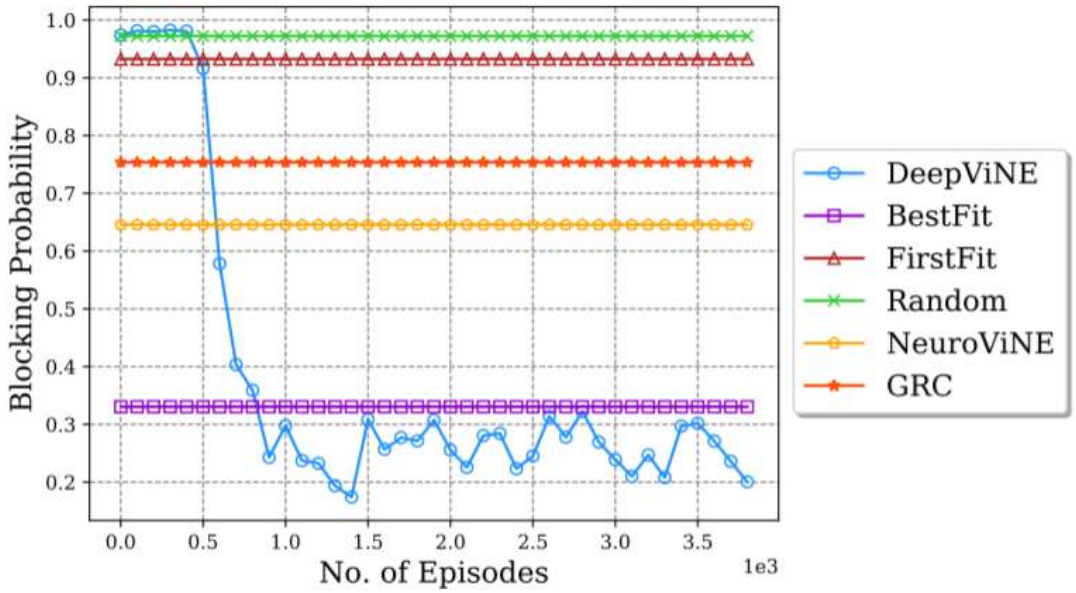
(a) Received reward.



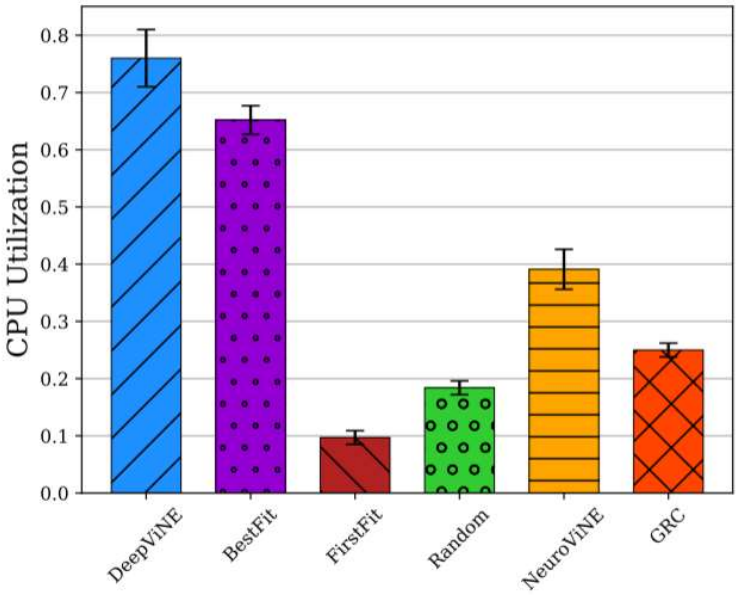
(b) Blocking probability.

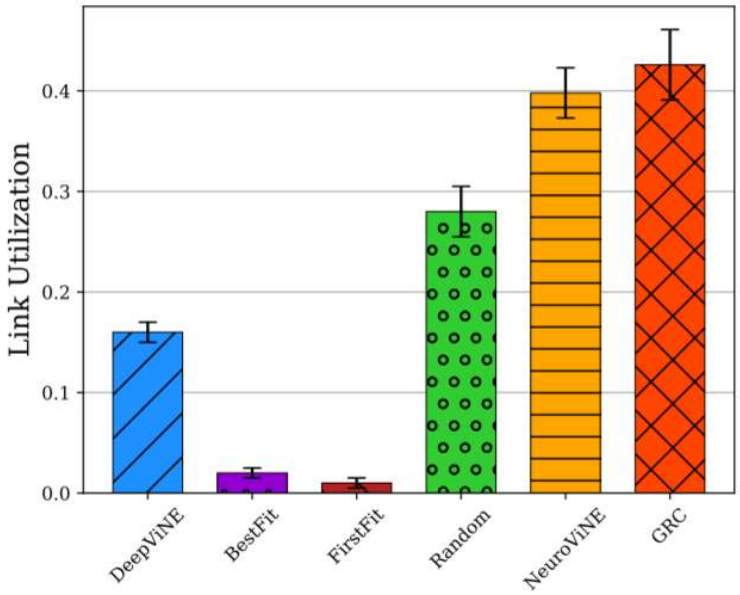
Evaluation

- Reward and objective are aligned
- Fast convergence


Blocking Probability
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
1e3
No. of Episodes
DeepViNE
BestFit
FirstFit
Random
NeuroViNE
GRC

Evaluation





Evaluation

- DeepViNE considers the physical link and node resources simultaneously, which eventually leads to higher CPU and link utilization, but also lower blocking probability.
- Underlying assumptions
  - Load-balancing (GRC, Neuro-ViNE)
  - Aggressive packing

Conclusion

- We presented a DRL-based VN embedding algorithm and overcame two important challenges
  - Small action space for fast convergence
  - Image representation of problem inputs
  - We presented simulation results with several existing works
- An interesting extension of this work is to modify the reward signal to model other system objectives such as minimizing energy consumption of the physical network.