Combinatorial Optimization for Sensor Placement with Deep Reinforcement Learning

Dr. Conor Muldoon, University College Dublin

Sensor placement problem

- Subset selection
  - Given a finite set of locations, choose a subset that maximises the utility
  - Example utility functions include the entropy of Gaussian processes and the mutual information of Gaussian processes of the selected set and ùnselected set
  - Set cover is a special case
  - NP-hard
- Combinatorial problem
  - There may be places where sensors or seismometer cannot be placed.
  - Sensors can only be placed to a certain accuracy

Submodular Optimisation

- Set function F on V is called submodular if

$$\text{For all } A,B \subseteq V: F(A)+F(B) \geq F(A \cup B)+F(A \cap B)$$

- Equivalent diminishing returns characterization:

$$\text{For } A \subseteq B,\ s \notin B,\ F(A \cup \{s\}) - F(A) \geq F(B \cup \{s\}) - F(B)$$

- The more sensors you have, the lower the loss, but decreases less with additional sensors
- Monotone function
- Greedy algorithm [Krause, et. al, 2008]
  - (1-1/e)-approximation algorithm [Nemhauser, et. Al, 1978]

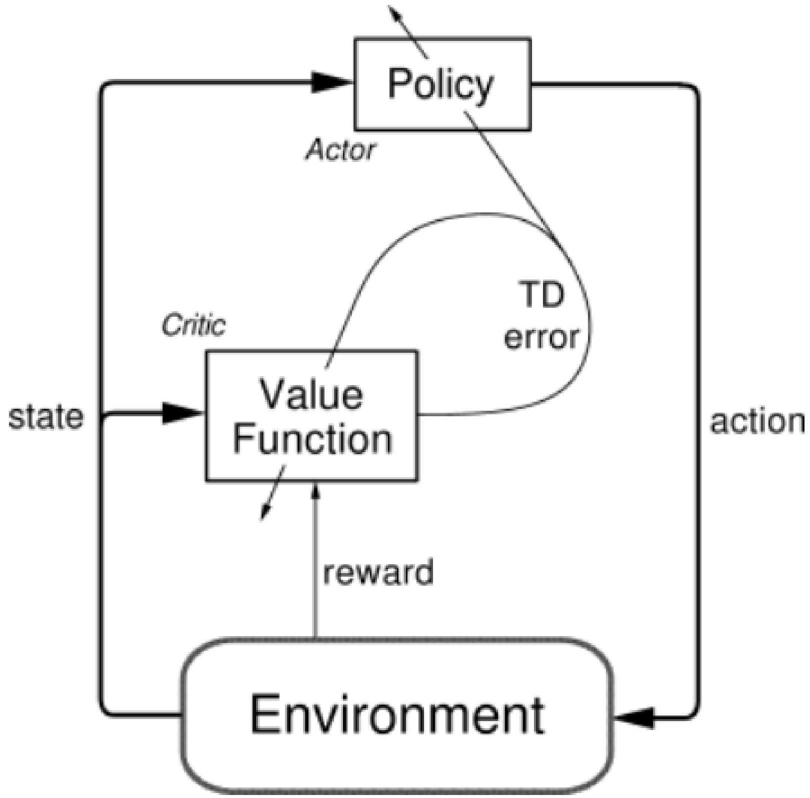
Numerical and metaheuristic algorithms

- Particle swarm optimisation
- Basin-hopping
  - Inspired from Monte-Carlo minimization
- Differential evolution

Combinatorial optimisation with deep reinforcement learning

- Introduced by Google brain [Bello, et al., 2016]
- Learn heuristics for approximate solving NP-hard optimisation problems using deep reinforcement learning.
  - Travelling salesperson problem, knapsack problem
- Actor-critic architecture
- Pointer networks
  - Additive attention mechanism
  - Encoders, decoders
  - Recurrent neural networks
  - LSTM cells
- ADAM optimizer

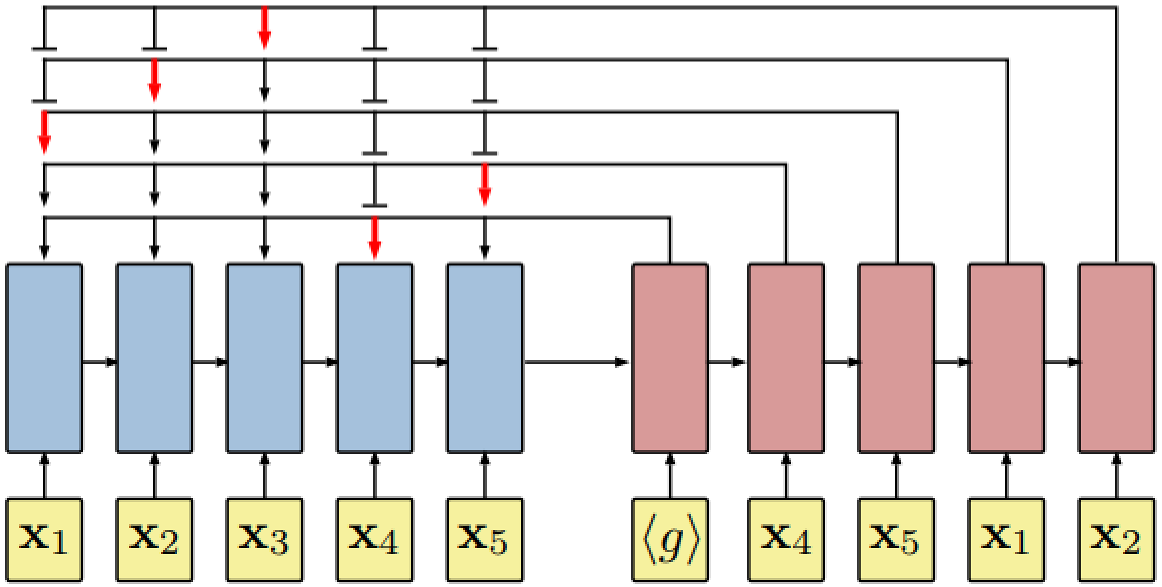
Actor-critic architecture



Actor-critic architecture [Sutton & Broto, 2018].

Sequence to sequence learning and pointer networks

- Example: Predictive replies to emails.



Vinyals, et al., Pointer Networks, 2015

Approximating NP-hard problems

- Travelling salesperson problem

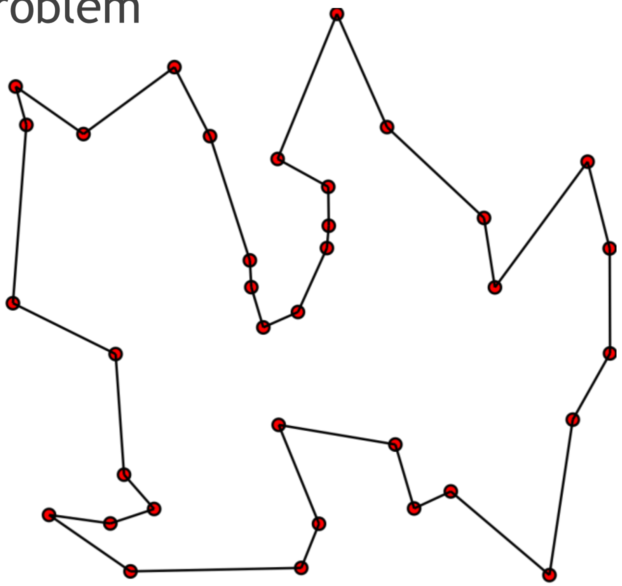


Image source Wikipedia

- Knapsack problem
  - Sequence order not relevant

$$\text{maximize} \sum_{i=1}^{n} v_i x_i$$

$$\text{subject to} \sum_{i=1}^{n} w_i x_i \leq W \text{ and } x_i \in \{0,1\}$$

Sensor selection with pointer networks

- Introduce new loss function
- Rather using the permutation to find the shortest tour, use the first k elements of as the selected sensor locations.
- Different loss function
- Pointer network passes the result of the additive attention mechanism to softmax
- Greedy decoding
  - Choose locations with the highest probability first from the softmax function
- Sample from multimodal distribution using softmax values as parameters

Loss function

- Scaled locations
- For a Wiener Filter, the normalized residual (squared mean error between the actual Newtonian noise and the estimated one) [Harms, 2015][Badaracco, 2021:

$$R(\omega) = 1 - \frac{\vec{C}^{\dagger}_{sn} \mathbf{C}^{-1}_{ss} \vec{C}_{sn}}{C_{nn}}$$

- $\vec{C}^{\dagger}_{sn}$ is the vector of the cross power spectral densities between all sensors and the test mass.
- $\mathbf{C}^{-1}_{ss}$ is the matrix of the cross power spectral densities of all sensors.
- C_{nn} is the power spectral density of the Newtonian noise in the test mass.

Preliminary implementation

- Trained on Tesla M10 GPU
- Implemented in PyTorch
- 80 sensor locations in pointer network
- Works in principle, but several improvements in relation to scalability in terms of potential sensor locations.
- Recurrent networks with a large dictionary will be very deep if viewed from an unrolled perspective.
- Architecture and encoding could be improved
- GPU memory issues

Future Directions

- Active search
  - Overfitting is not a problem in this instance
- Beam search with truncation
  - Heuristic/optimisation for breadth first search
- Transformer rather than LSTM cells
- Deep reinforcement learning for numerical optimisation.

References

Krause, A., Singh, A., & Guestrin, C. (2008). Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies. *Journal of Machine Learning Research*, *9*(2).

Nemhauser, G. L., Wolsey, L. A., & Fisher, M. L. (1978). An analysis of approximations for maximizing submodular set functions—I. *Mathematical programming*, *14*(1), 265-294.

Bello, Irwan, et al. "Neural combinatorial optimization with reinforcement learning." *arXiv preprint arXiv:1611.09940* (2016).

Vinyals, O., M. Fortunato, and N. Jaitly. "Pointer networks. In Advances in Neural Information Processing Systems 28, p." (2015).

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

Travelling salesperson image: https://en.wikipedia.org/wiki/Travelling_salesman_problem

Harms, J. (2015). Terrestrial gravity fluctuations. *Living reviews in relativity*, *18*(1), 1-150.

Badaracco, F.. "Newtonian Noise studies in 2nd and 3rd generation gravitational-wave interferometric detectors." (2021).