The Cross-Entropy Method

compiled by D.Gueorguiev 3/20/25

The travelling salesman problem (TSP), the quadratic assignment problem (QAP), and the max-cut problem (MCP) are combinatorial optimization problems where the problem studied is completely known and static. In contrast, the buffer allocation problem (BAP) is a noisy estimation problem where the objective function needs to be estimated since it is unknown.

# References

[1] [A Tutorial On The Cross-Entropy Method, Pieter-Tjerk de Boer et al, MIT, 2003](https://github.com/dimitarpg13/reinforcement_learning_and_game_theory/blob/main/articles/ReinforcementLearning/A_Tutorial_on_the_Cross-Entropy_Method_deBoer_2003.pdf)

[2] [Learning Tetris Using the Noisy Cross-Entropy Method, Istvan Szita, Andras Loerincz, 2006](https://github.com/dimitarpg13/reinforcement_learning_and_game_theory/blob/main/articles/ReinforcementLearning/Learning_Tetris_Using_the_Noisy_Cross-Entropy_Method_Szita_2006.pdf)

[3] [Approximate Dynamic Programming Finally Performs Well In The Game of Tetris, Victor Gabillon et al, INRIA, 2013](https://github.com/dimitarpg13/reinforcement_learning_and_game_theory/blob/main/articles/ReinforcementLearning/NIPS-2013-approximate-dynamic-programming-finally-performs-well-in-the-game-of-tetris-Paper.pdf)

[4] <https://en.wikipedia.org/wiki/Cross-entropy_method>

[5] [The Cross-Entropy Method: A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation and Machine Learning, RY Rubinstein, DP Kroese, 2004](https://github.com/dimitarpg13/reinforcement_learning_and_game_theory/blob/main/books/The_Cross_Entropy_Method_A_Unified_Approach_Rubinstein_Kroese_2004.pdf)

# Appendix

## The Buffer Allocation Problem

The BAP finds the optimal buffer capacity on each stage in a flow line i.e. the minimum buffer capacity able to guarantee the target throughput.

Let us consider a flow as depicted in the Figure below:

A diagram of a block diagram

Description automatically generated

Figure 1: flow line under consideration in the BAP

There are machines, depicted with circles, and inter-machine buffers of finite capacity . All parts are processed by all machines of the line, ordered according to the arrival sequence. Parts are always available in front of the first machine, and parts can immediately leave the system after being processed by the last machine. The inter-machine buffer has finite capacity . A full buffer causes the *blocking* of the upstream machine, and an empty buffer causes the *starvation* of the downstream machine. The processing times of each machine can be generated as independent and identically distributed variables from a general distribution or calculated as the sum of the two variables corresponding to the processing and the repair time, if the machine is considered unreliable. The term *processing time* will represent the processing time of reliable machines, and the processing time without including the repair time of unreliable machines.

**Notation**

index , set and number of system stages

set of all the stages but the last one

lower and upper bound of the capacity of buffer

index, set and number of parts in the simulation horizon

processing time of part at machine

buffer capacity index

set of all the possible capacities for buffer except the lower bound