Artificial Intelligence and Machine Learning (AIML)

2023–24

- **Last lecture**: essential mathematical background
- **This lecture**: combinatorial optimization in AI, combinatorial explosion and computational complexity

AIML

Combinatorial optimization: overview

- Many **symbolic AI** (GOFAI) problems are **optimization** problems
- **Minimize objective function** over all possible **configurations**
- Exact vs. approximate methods
- Exact: **guaranteed** to find an **optimal solution**; approximate: **not guaranteed** to find an optimal solution, may **come close** ("good enough")
- Exact methods slow, approximate methods unreliable

AIML

Combinatorial optimization: overview

- Minimize objective function *F* with respect to configurations *X* from a set of possible configurations,

$$X^\star = \arg\min_{X' \in \mathcal{X}} F\left(X'\right)$$

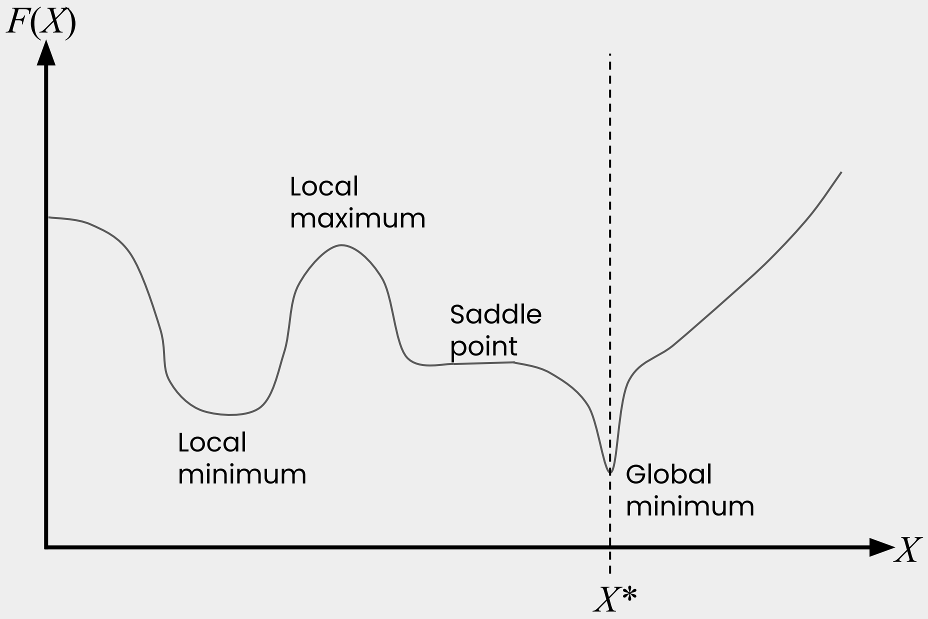
- Globally optimal solution satisfies:

$$F\left(X'\right) \geq F\left(X^\star\right), \ \forall X' \in \mathcal{X}$$

- Locally optimal solution for neighbourhood of all configurations

$$F\left(X'\right) \geq F\left(X^\star\right), \ \forall X' \in \mathcal{N}$$

Optimizing objective functions



AIML

Exhaustive solution

- Generate all possible configurations, compute objective function for each, select the best one
- Always applicable, always guaranteed to find an optimal solution
- Simple to formulate, often there are simple methods to systematically generate all possible configurations
- Often the starting point for more efficient, exact algorithms (see dynamic programming later)

AIML

Combinatorial explosion

- Size of the configuration space: typically exponential or worse complexity, number of possible configurations as a function of the **problem size** *N*
- Complexity orders: from constant time/space (tractable) to factorial (intractable)
- In general: polynomial and better are considered "practical", whereas exponential and worse are considered "impractical", when taking into account realistic computational resources
- Implication that exhaustive, although generally applicable, simple and exact, is rarely practical; justifies the need for AI: smart solutions to otherwise intractable problems

AIML

Complexity classes

constant	$O(1)$	Tractable
logarithmic	$O(\log N)$	
linear	$O(N)$, $\mathrm{O}(Nk)$	
log-linear	$O(N \log N)$	
polynomial	$O(N^k)$	
exponential	$O(k^N)$	Intractable
factorial	$O(N!)$	

Computational complexity

- Space versus time complexity
- Trading off time for space and vice-versa

References and further reading

- **MLSP**, Section 1.8, Section 2.6
- **CLRS**, Chapter 3
- **R&N**, Section A.1