Learning Aims

Problem recognition

- Know what features of a problem make it soluble by computational methods
- Categorise different types of problem and solutions
- Explore different strategies for problem-solving
- Discuss problem recognition

Computable problems

- A problem is defined as being computable if there is an algorithm that can solve every instance of it in a finite number of steps.
- Some problems may be, in theory, computable, but if they take millions of years to solve, they are, in a practical sense, insoluble.
- An example of such a problem is the cracking of a secure password.
- If you choose a password of 10 characters or more, comprising a mixture of random letters, numbers and special symbols, it will be impossible to crack. You can test the strength of your passwords on various websites.

Methods of problem solving

There are many ways of problem solving, including:

- Abstraction and decomposition
- Enumeration (listing all cases)
- Simulation and automation
- Theory and mathematics

Enumeration - Brute force method

- Many problems and algorithmic puzzles can be solved by exhaustive search – trying all possible solutions until the correct one is found.

- Thousands of problems which were in the past insoluble have, thanks to the power of modern computers, become soluble.

For example, a database of fingerprints or DNA can within a reasonable time find the identity of an individual, if his or her fingerprints or DNA are on the database.

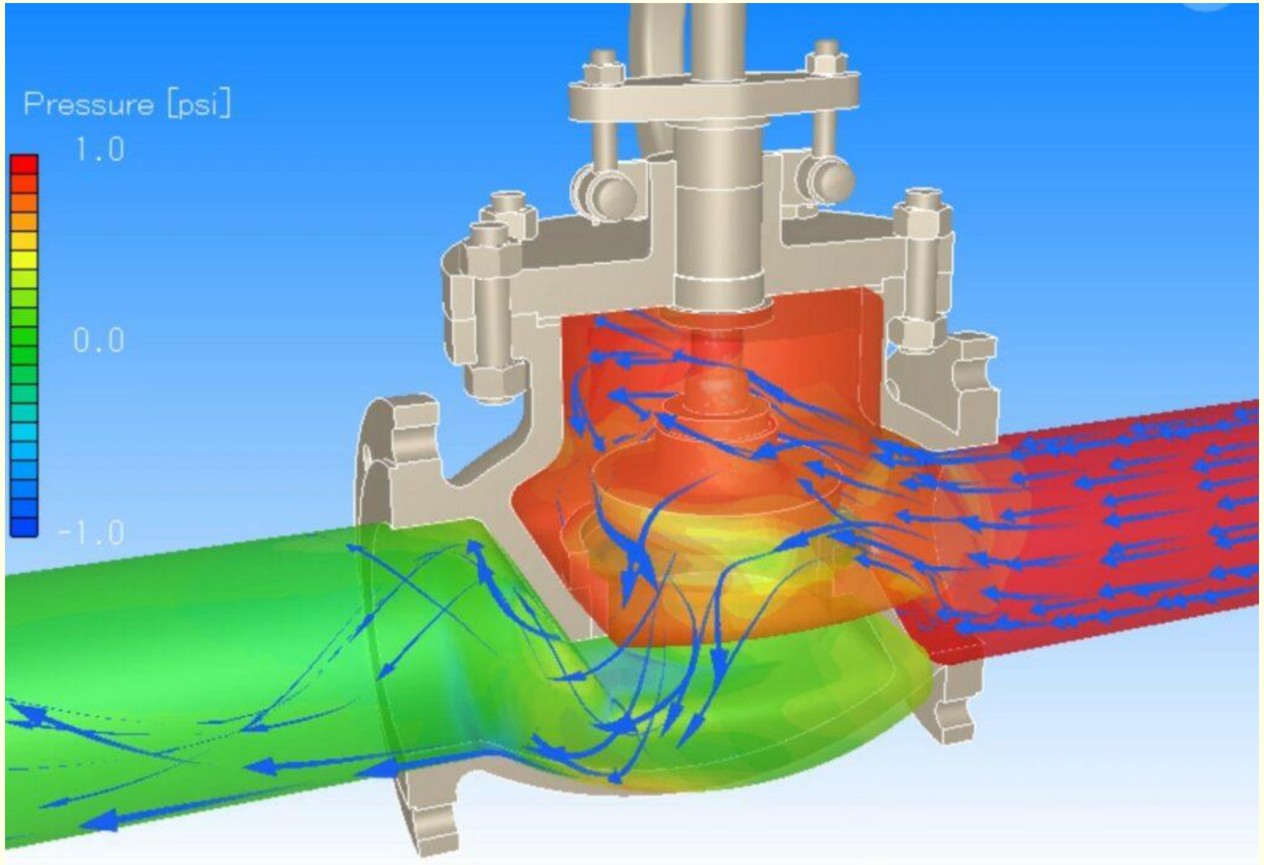
The main problem with the exhaustive search strategy is that it becomes very slow and inefficient as the problem gets bigger. This is because, as the size of the problem increases, the number of possible solutions grows very quickly, making it harder and more time-consuming to check every option.

Simulation

Simulation is the process of designing a model of a real system in order to understand the behaviour of
the system, and to evaluate various strategies for its operation.

Such problems include:

- Financial risk analysis
- Population predictions
- Queueing problems
- Climate change predictions
- Engineering design problems

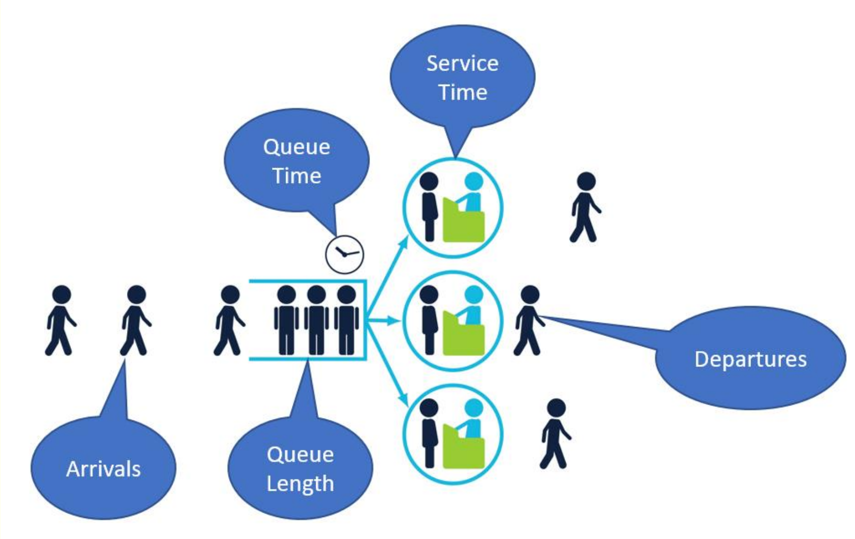


Simulations and abstraction

Simulating a system invariably makes use of abstraction to reduce the problem to its essentials, removing all unnecessary details.

Queueing problems, for example, include problems of finding out how many checkouts are needed in a new supermarket

Q2: How would abstraction be applied to this type of problem? What factors would be relevant, and what would be irrelevant?



- Simulation can also involve building a physical model of, for example, a spacecraft, ship or wind turbine, so that its behaviour can be studied.
- This is obviously useful when it would be too expensive, dangerous or impractical to carry out tests on the real thing.
- A model can be used to evaluate performance or test outcomes.