METAHEURISTICS

Optimization models

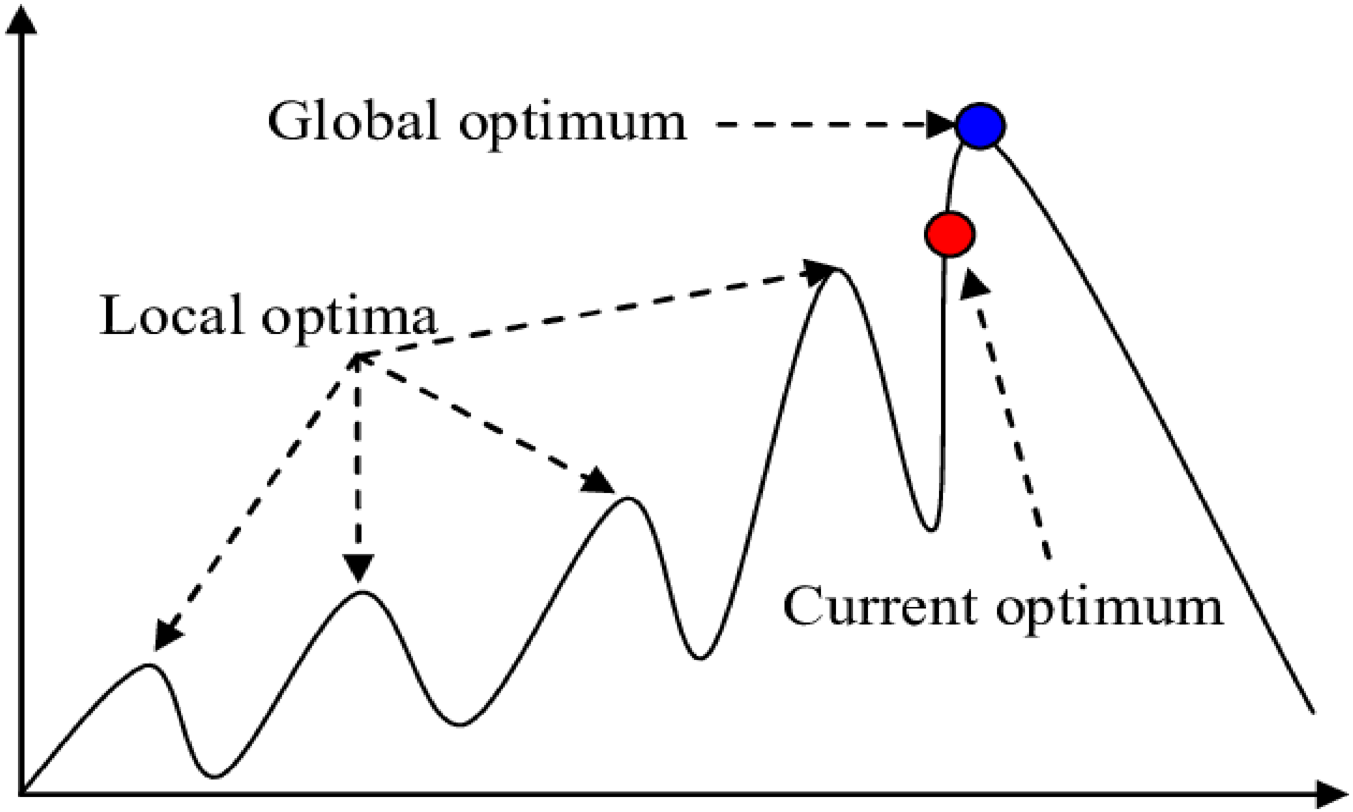
Optimization methods

Adopted from:

Talbi, E.-G. (2009). Metaheuristics: from design to implementation (Vol. 74): John Wiley & Sons.
Blum, C. & A. Roli 2003. Metaheuristics in combinatorial optimization: Overview and conceptual comparison. ACM Computing Surveys (CSUR) 35(3): 268-308.
Hybrid Metaheuristics, May 2010, BIOMA 2010, Ljubljana, Slovenia, *file:///C:/Users/dell/Downloads/Hybrid_Metaheuristics.pdf*

Nasser R. Sabar, 2015, The University of Nottingham Malaysia Campus
Nasser.Sabar@Nottingham.edu.my

Optimization model



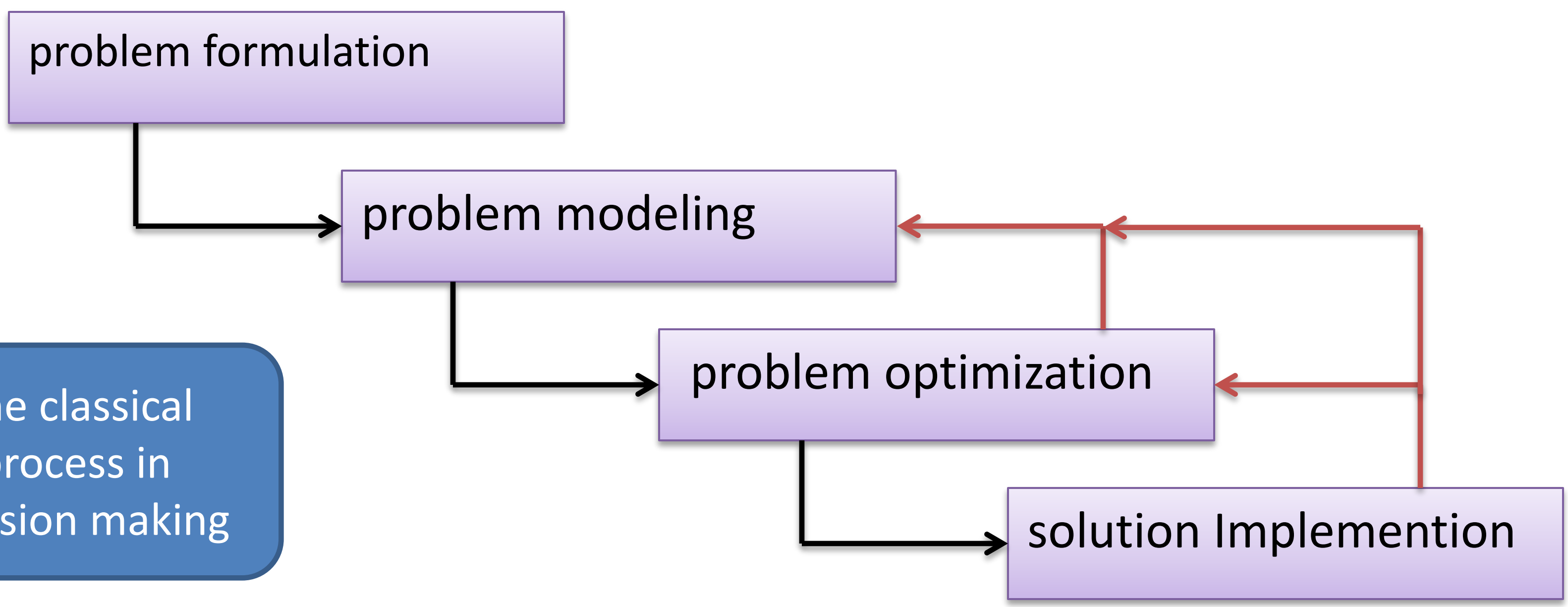
Optimization model

optimization models can be used to formulate and solve optimization problems. More generally, it designed to help us make "better" decisions.

Optimization model

Decision making is everywhere. As the world becomes more and more complex and competitive, decision making must be tackled in a rational and optimal way.

Decision making consists in the following steps:



The classical process in decision making

Optimization model

PROBLEM FORMULATION

1. a decision problem is identified.

2. an initial statement of the problem is made.

3. The **Decision variables** and the **objective(s)** of the problem are outlined.

Optimization model

PROBLEM MODELING

An abstract mathematical model is built for the problem.

The model will reduce the problem to well-studied optimization models. Usually, models we are solving are simplifications of the reality.

The models involve approximations and some times they skip processes that are complex to represent in a mathematical model.

Optimization model

PROBLEM OPTIMIZATION

Once the problem is modeled, the solving procedure generates a "good" solution for the problem. The solution may be optimal or suboptimal.

Notice that we are finding a solution for an abstract model of the problem and not for the originally formulated real-life problem.

The obtained solution performances are indicative when the model is an accurate one.

Optimization model

PROBLEM IMPLEMENTATION

The obtained solution is tested practically by the decision maker and is implemented if it is “acceptable” or Not.

If the solution is unacceptable, the model **and/or** the optimization algorithm has to be improved and the decision making process is repeated.

Optimization model

PROBLEM FORMULATING – Example1

A company produces two kinds of products: A and B. the profit of one unit of A and B is $40 and $70, respectively. However, the company has limitations in its **labor** (a total of 501 labor hours available per time slot; each A needs 4 h and B 3 h), **machine** (a total of 401 machine hours available, each A needs 2 h and B 5 h), and **marketing requirements** (the need to produce 10 units of A and 20 units of B respectively). The decision problem is how many A and B should be produced to obtain the maximum profit.

Optimization model

PROBLEM FORMULATING – Example1

Decision variables:

X_1: units of A to be produced;

X_2 : units of B to be produced.

Objective function:

Maximize total profit: 40 x_1 + 70 x_2;

Constraints:

Labor constraint (hours): $4x_1 + 3x_2 \leq 501$;

Machine constraint (hours): $2x_1 + 5x_2 \leq 401$;

Marketing requirement for x_1 (units): $x_1 \geq 10$;

Marketing requirement for x_2 (units): $x_2 \geq 20$:

Optimization model

PROBLEM FORMULATING – Example2

A given company produces two products **Prod1** and **Prod2** based on two kinds of raw materials **M1** and **M2**. The objective consists in finding the most profitable product mix. Following table presents the daily available raw materials for M1 and M2, and for each product Prodi the used amount of raw materials and the profit.

	prod 1	prod 2	Material availability
M1	6	4	24
M2	1	2	6
Profit per unit	5$	4$	

Optimization model

PROBLEM FORMULATING - Example

Decision variables:

x_1: amounts of Prod1;

x_2 : amounts of Prod2.

Objective function:

Maximize total profit: $5x_1 + 4x_2$;

Constraints:

$6x_1 + 4x_2 \leq 24$;

$1x_1 + 2x_2 \leq 6$;

Marketing requirement for x_1 (units): $x_1 \geq 0$;

Marketing requirement for x_2 (units): $x_2 \geq 0$:

Optimization Methods

The goal of optimization methods is to find an optimal or near-optimal solution with low computational effort.

There are two different types of optimization methods: **Exact optimization methods** and **Heuristic optimization methods** (approximate methods).

Exact optimization methods:

1. **G**uarantee finding an optimal solution in unreasonable time.
2. Applied to small instances of difficult problems.

Heuristic optimization methods : **(ACCEPTABLE PERFORMANCE AT ACCEPTABLE COSTS)**

1. Generate high quality solutions in a reasonable time for practical use.
2. No guarantee of finding a global optimal solution.
3. Find "good" solutions on large-size problem instances.

Optimization Methods

Heuristics classified into two families:

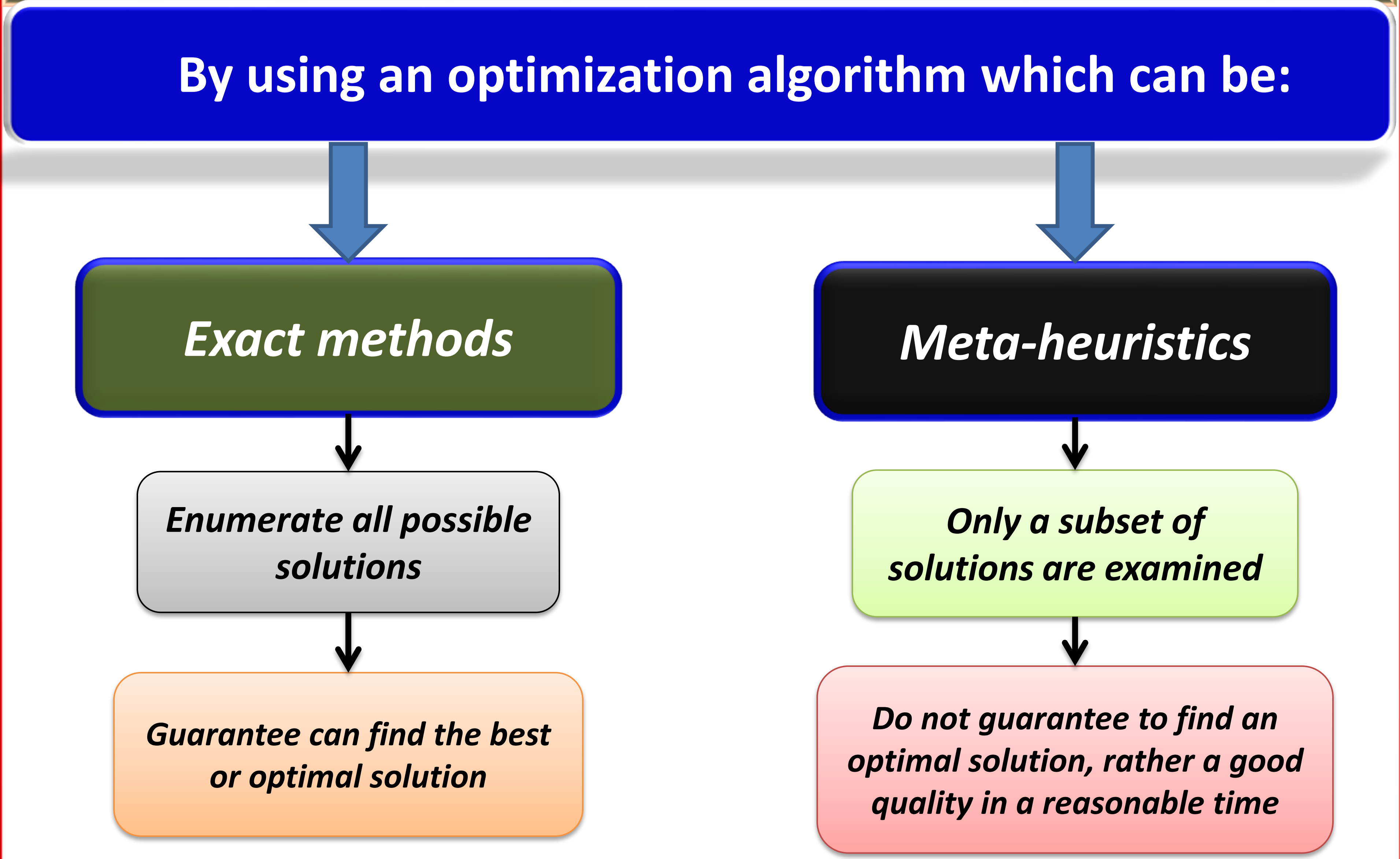
Specific heuristics:

designed to solve a specific problems and/or instances.

Metaheuristics:

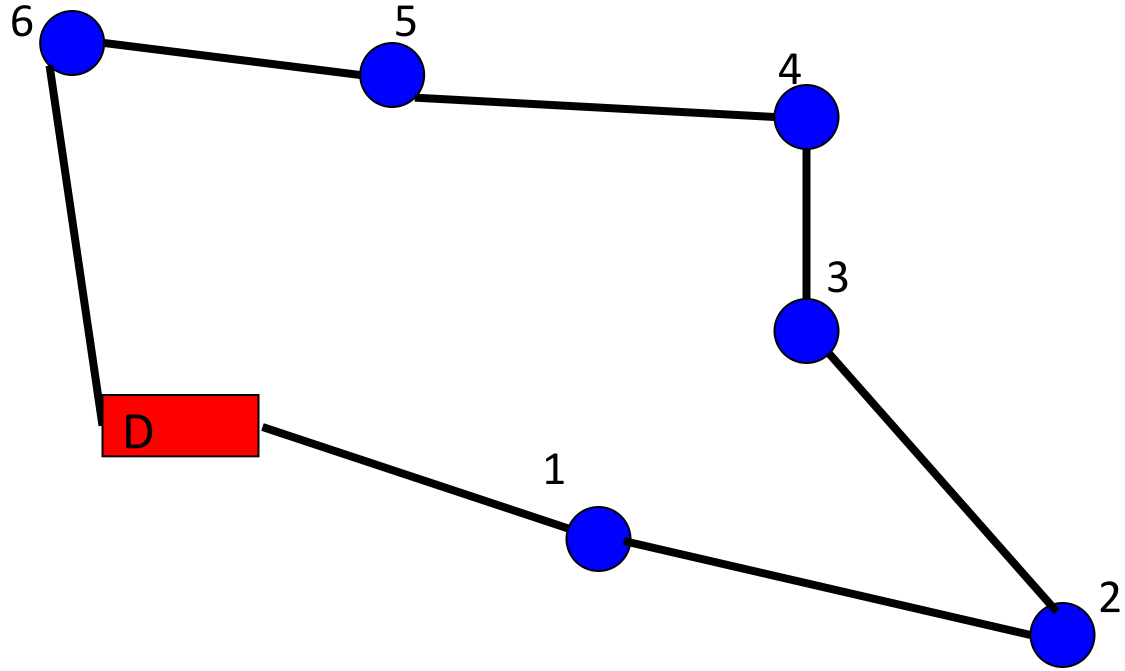
Applied to solve almost any optimization problem.

How to solve optimization problems?



Example : TSP

6 cities

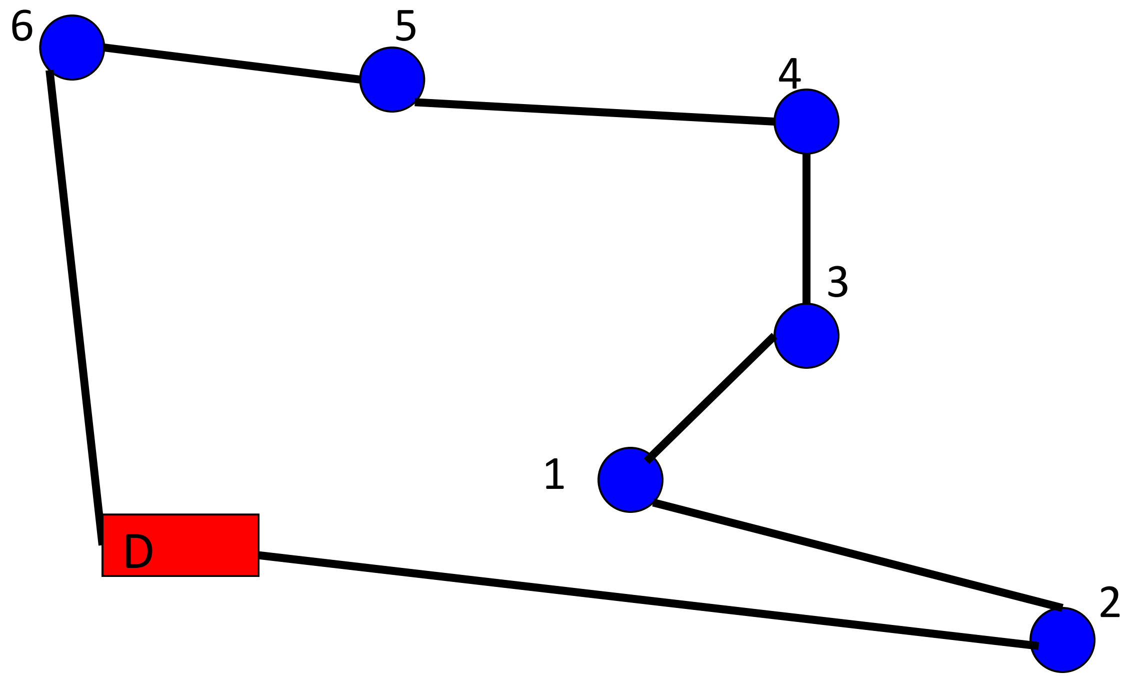


solution/ route #1

1→2→3→4 →5→6

Example : TSP

6 cities

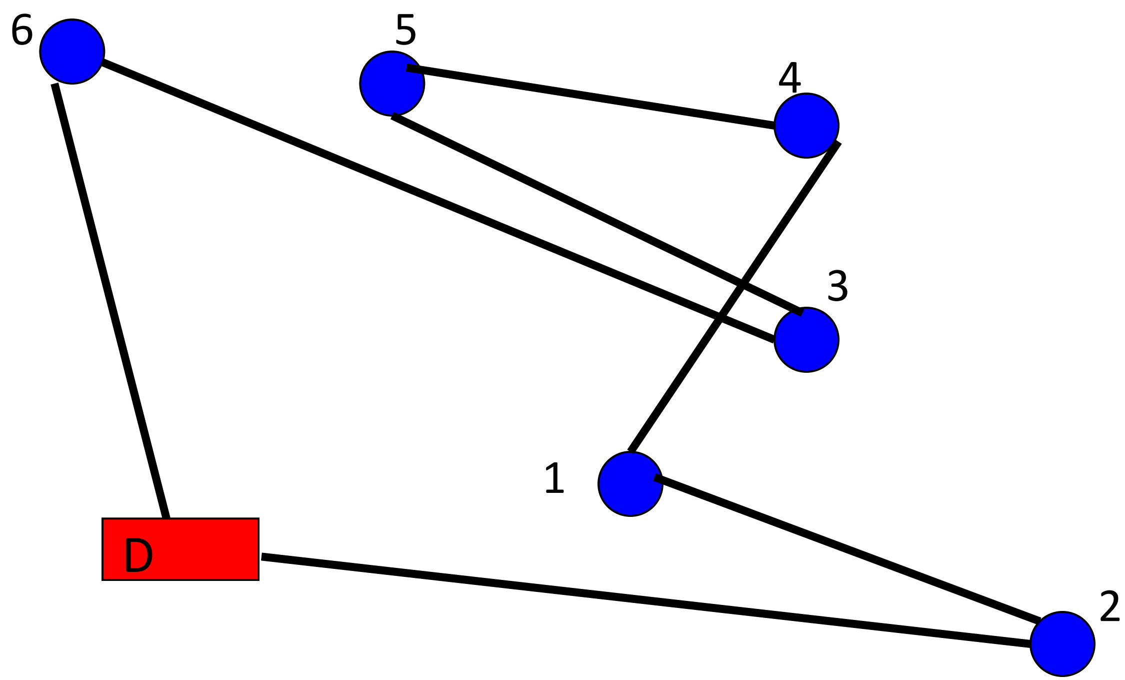


solution/ route #2

2→1→3→4 →5→6

Example : TSP

6 cities



solution/ route #3

6→3→5→4 →1→2

Example: TSP

Q: How many solutions/routes can be generated?

A: The number of possible routes is (n!)

6 cities TSP has *720* possible solutions

10 cities TSP has *181,000* possible solutions

20 cities TSP has *10,000,000,000,000,000* possible solutions

But in real world scenarios, we have more than 20 cities and some times up to *250* or more cities !!!!

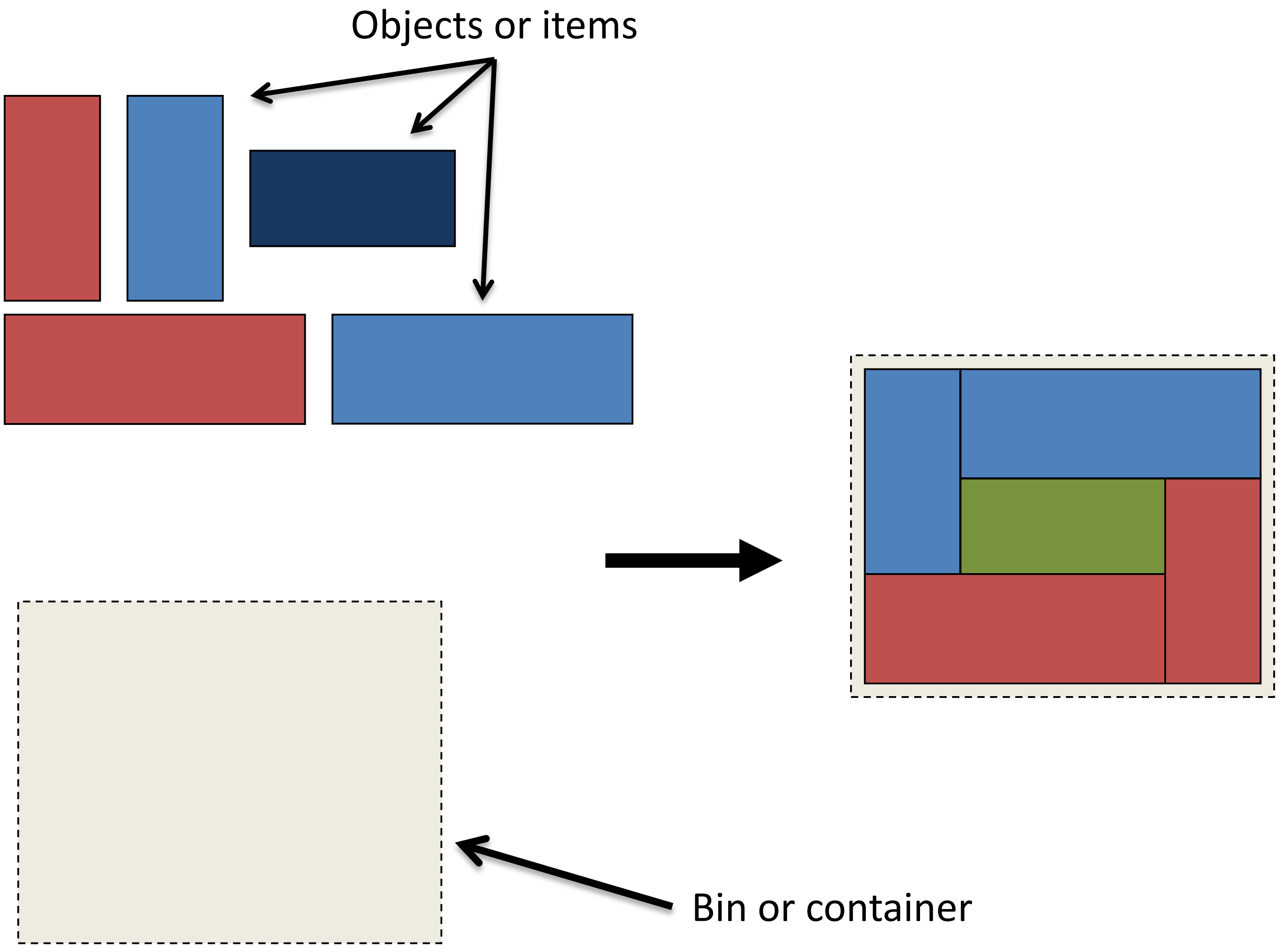
Example : Bin Packing

Given a bin capacity C>0 and a list of objects {p_1, p_2,...., p_n},

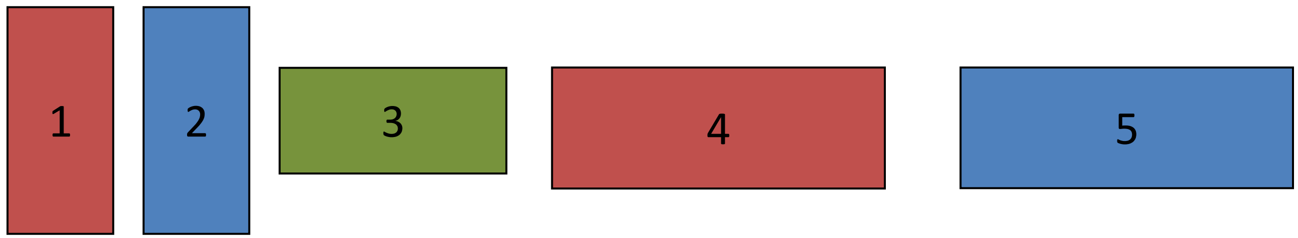
Each p_i has size s_i, such that $0 <= s_i <= C$. i.e. none of the objects too big to fit in a bin

What is the smallest number of bins needed to accommodate all objects/items?

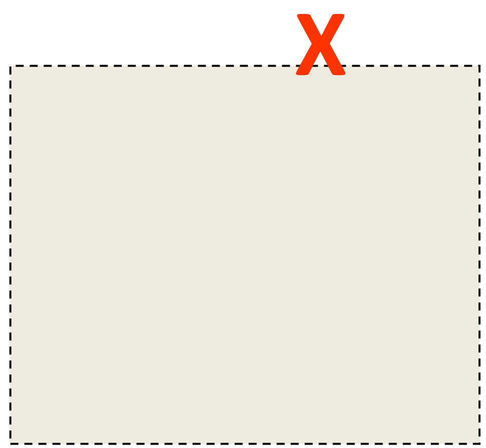
Example : Bin Packing



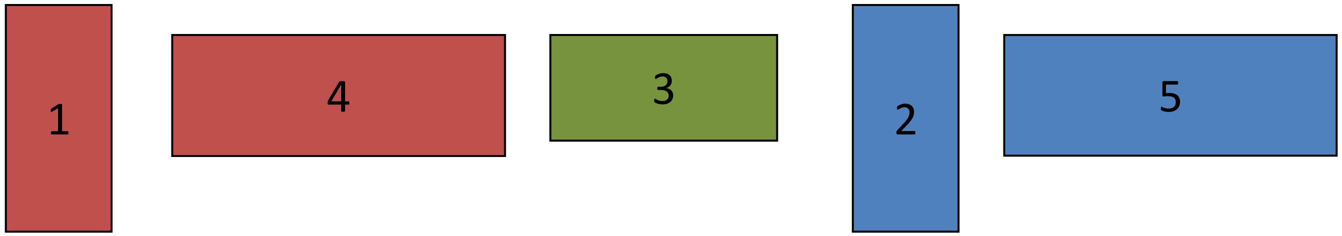
Example : Bin Packing



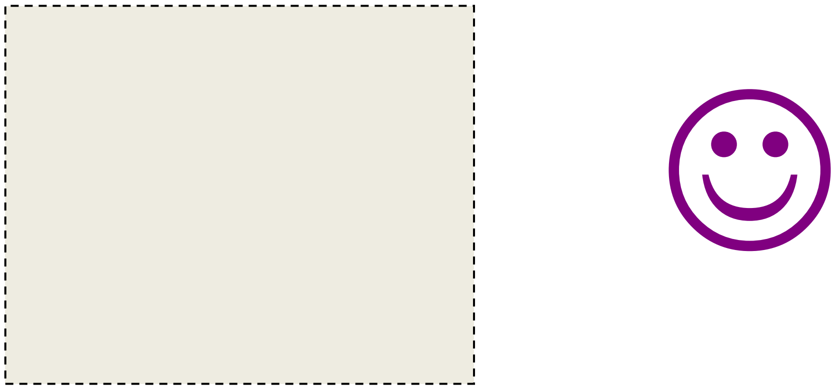
Sequence = 1,2,3,4,5



Example : Bin Packing



Sequence = 1, 4, 3, 2, 5



When we changed the sequence, we got a solution

Example : Bin Packing

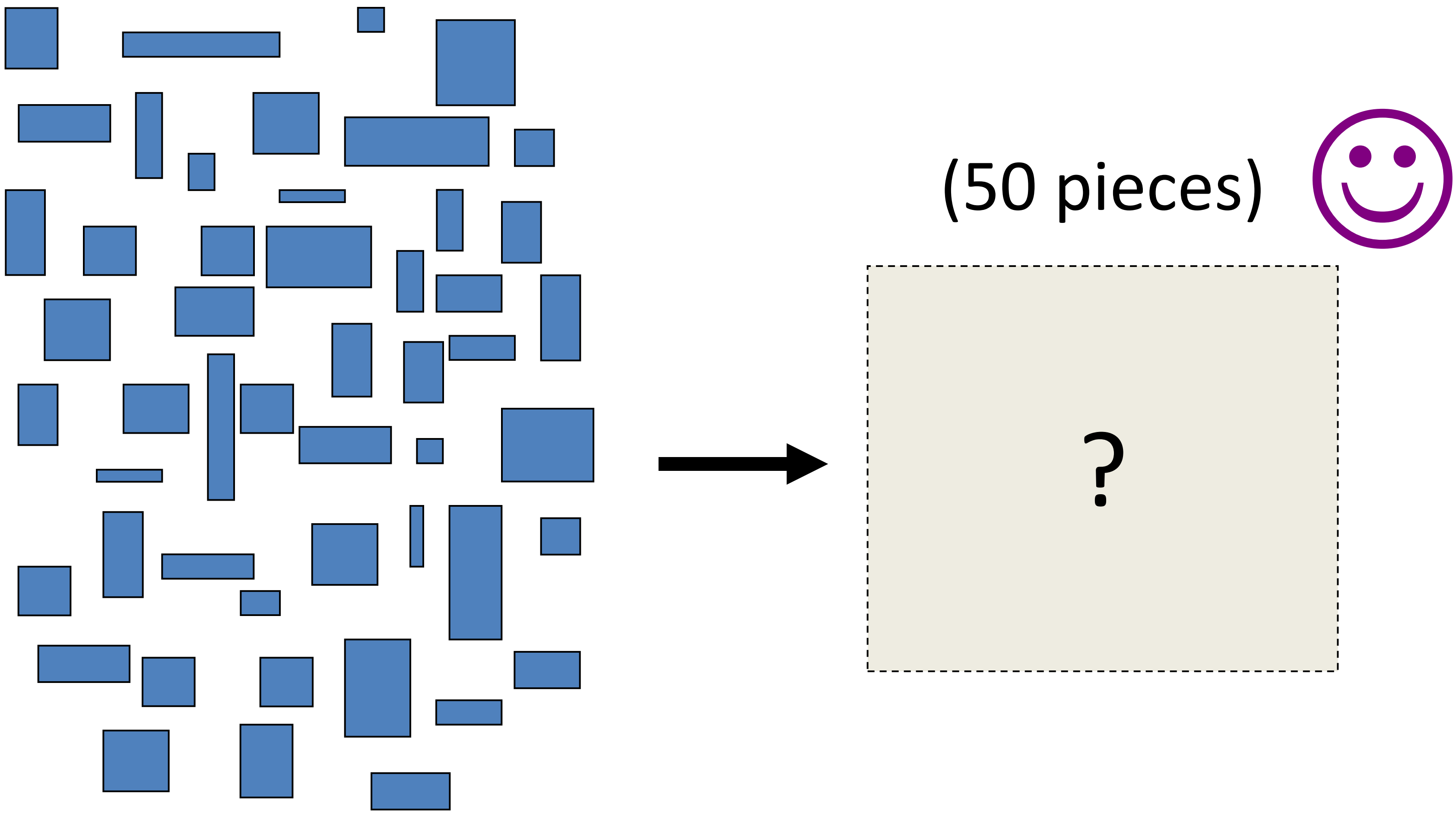
For 5 pieces there are 5 x 4 x 3 x 2 x 1 different orderings (sequences) **for the placement (*n!*) = 120 combinations**

- A piece of cake for a computer to try each combination in (nearly) no time at all

Very easy to do!

Example: Bin Packing

How do we pack all of these?



Example: Bin Packing

50 pieces means 50 x 49 x.....2 x 1 different orderings=

30414093201713378043612608166064768844377641568960512000000000000

That's quite a lot really and its not easy to enumerate/generate all possible solutions

Example : Bin Packing

If a computer could evaluate 1000 orderings per second it'd still take approximately:

964424568801159882154128873860501295166718720476931506

years! (Just for 50 Items)

What does this mean?

It is impossible to enumerates all possible solutions within a reasonable or acceptable computational times.

For this reason, exact methods or those based on mathematic can be only used for a small size problems

Although they can generate optimal solution, their computation time grow exponentially as the problem size increase.

Why these problem are hard and what we can find in real world problem [Talbi. 2009]

Huge search spaces

The model of the problem is difficult to define

Quality of solutions vary with respect of time

Their search space is heavily constrained

For this reason, We use meta-heuristic algorithms as they

- **faster** than exact optimization algorithms

- **simpler** and more readily understood than most exact optimization algorithms

However

Meta-heuristic do **not**, in general, provide **optimal** solutions, but rather **good quality solutions** in **acceptable amount of time**.

Meta-heuristics

A method for solving very general classes of problems. It combines objective functions or heuristics in an abstract and hopefully efficient way, usually by treating them as black box-procedures