In The Name of God The Most Compassionate, The Most Merciful

Brushless PM Machines

Design, Optimization and Analysis

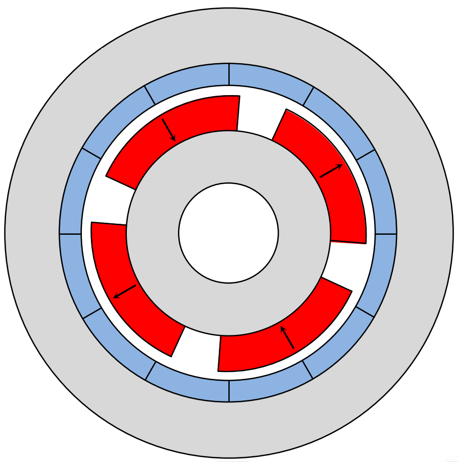
Table of Contents

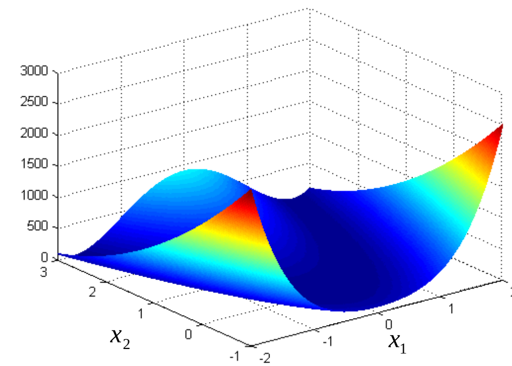
Aims and Objectives are to

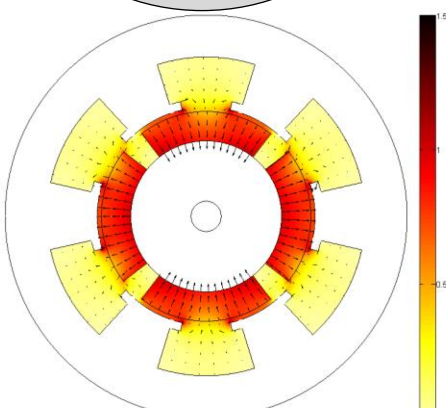
optimally design a

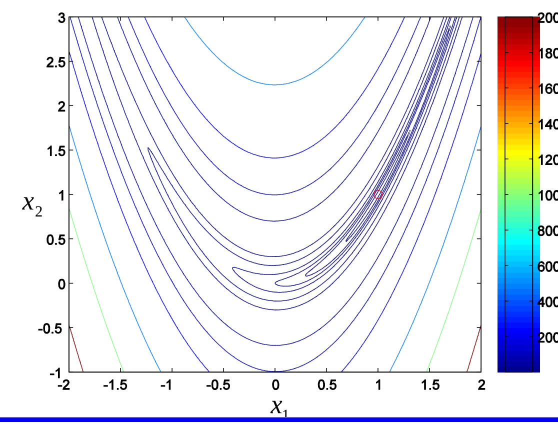
permanent Magnet brushless motor using a

metaheuristic optimisation technique







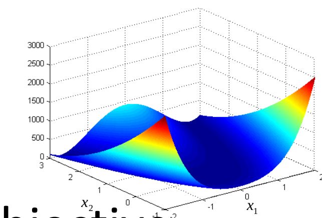


Optimisation problem

Any **optimisation problem** can be generally expressed by the following minimisation form   where $\mathbf{x}$ is the vector of opt. var.

$$\mathbf{x}_{opt} = \arg\min_{\mathbf{x}} f(\mathbf{x}) \text{ where } \qquad \mathbf{x} = [x_1 \; x_2 \; \ldots \; x_n]$$

subject to a number of equality and inequality constraints



$$g_i(\mathbf{x}) \le 0 \qquad i = 1, 2, \ldots, p$$

$$h_i(\mathbf{x}) = 0 \qquad i = 1, 2, \ldots, q$$

In the case of **multi-objective optimisation problems** (say *m* objective functions)

$$\mathbf{x}_{opt} = \arg\min_{\mathbf{x}} \left[f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x}) \right]^T$$

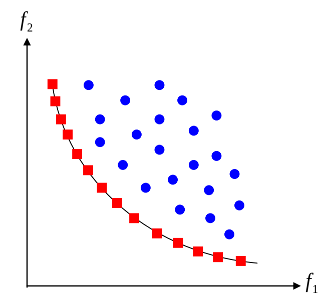
Multi-objective optimisation problems are generally solved using two main methods:

aggregating function

$$\mathbf{x}_{opt} = \arg\min_{\mathbf{x}} \sum_{k=1}^{m} w_k f_k(\mathbf{x})$$

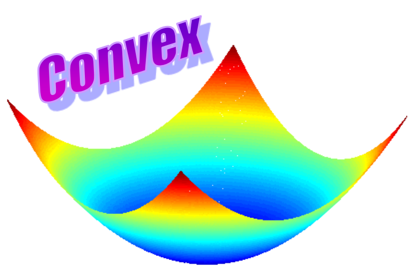
and

Pareto technique (Pareto points are those solutions in which any improvement in one objective function is only possible with worsening at least one of the other objectives)

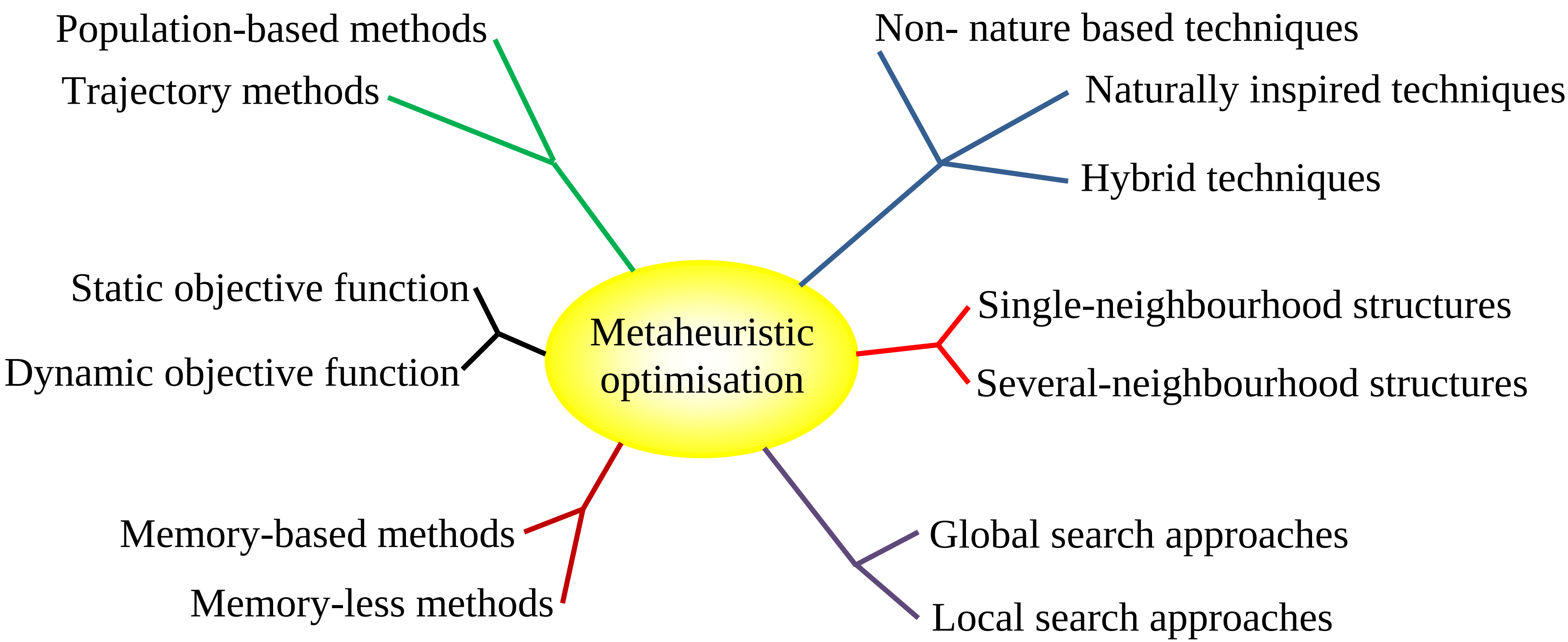


Metaheuristic optimisation

- Metaheuristic optimisation techniques **iteratively explore** the optimisation search space for an optimal solution.
- A **guided random search** mechanism is normally used to gradually improve the candidate solution in terms of the fitness value.
- Metaheuristic optimisation techniques have **several advantages** compared to that of the gradient-based techniques:
  - They can be applied to optimisation problems with **discontinuous** and/or **non-differentiable objective functions** as well as to those with continuous and differentiable fitness functions.
  - They can be used in problems with **non-convex** as well as convex objective function
  - They can deal with **very large search space**.
  - In general they are **not case dependant** and can be applied to any problem with no or minor modification.
- Despite these advantages they generally **do not guarantee the global optimum** solution.



Classification of metaheuristic optimisation techniques

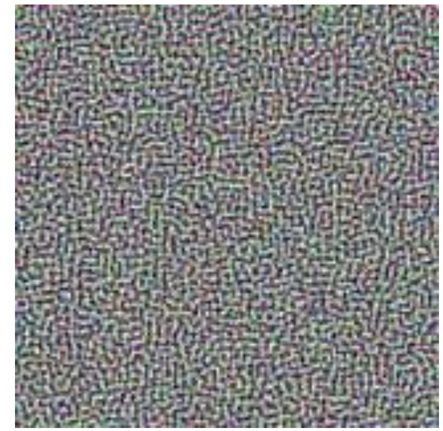


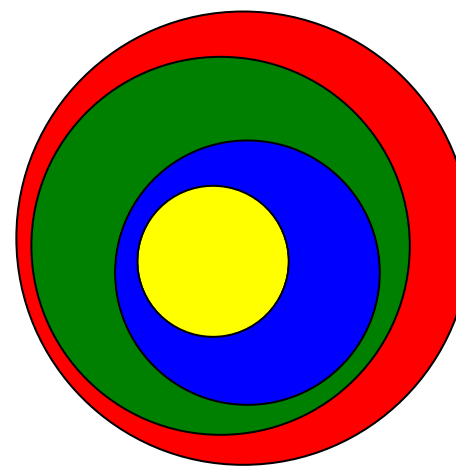
Metaheuristic optimisation

- **Trajectory methods**
  - Simulated annealing (1983)
  - Tabu search (1986)
  - Local search methods
    - GRASP (1995)
    - Variable neighbourhood (1997)
    - Guided local search (1997)
    - Iterated local search
- **Population-based methods (mostly evolutionary techniques)**
  - **Evolutionary algorithms**
    - Evolutionary programming (1964)
    - Evolution strategy (1965)
    - Genetic algorithms (1975)
    - Genetic programming (1980)
    - Memetic algorithm (1989)
    - Cultural algorithm (1994)
    - Differential evolution (1996)
    - Bacteriologic algorithms (1999)
    - Invasive weed opt. (2006)
    - Biogeography-based opt. (2008)
  - **Swarm intelligence (agent-based optimisation)**
    - Ant colony optimisation (1992)
    - Particle swarm optimisation (1995)
    - Shuffled frog-leaping (2003)
    - Glow-worm swarm opt. (2005)
    - Bee colony optimisation (2005)
    - Honey bee mating opt. (2006)
    - Monkey search (2007)
    - Firefly algorithm (2008)
    - Cuckoo search (2009)
  - **Natural/physical phenomena**
    - Stochastic diffusion search (1989)
    - Cross entropy (1997)
    - Harmony search (2001)
    - Intelligent water drops (2007)
    - Central force optimisation (2007)
    - Gravitational search (2009)
    - Galaxy-based search (2011)
    - Spiral optimisation (2011)

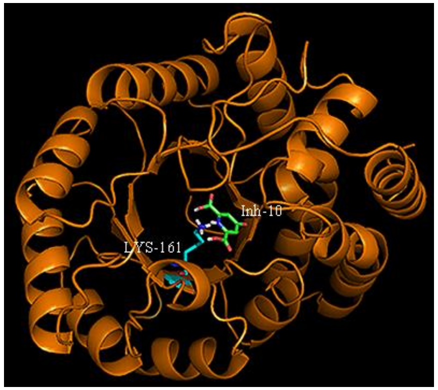
Trajectory methods

- **Simulated annealing** (**SA**) is inspired by annealing in metallurgy in which a metal is heated up and cooled down controllably to reconfigure its crystals and remove the impurity.
- **Tabu search** (**TS**) is the improvement of local search and hill climbing techniques by incorporating a memory list of the visited potential solutions to prevent revisiting those solutions.
- In **variable neighbourhood search** (**VNS**), a metaheuristic is obtained through a systematic change of neighbourhood of a local search heuristic.





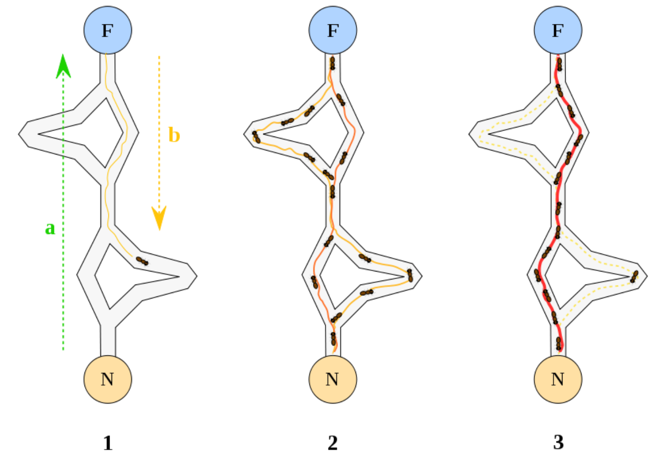
Population-based techniques
1- Evolutionary algorithms



- **Evolutionary programming** (EP) mainly uses mutation and selection operators not the recombination.
- **Evolution strategy** (ES) was introduced based on the ideas of adaptation and evolution and uses mutation and selection as primarily search operators.
- **Genetic algorithm** (GA) normally uses selection, crossover and mutation operators. Elitism may be included.
- In **differential evolution** (DE), an innovative recombination operator is defined to create new solution candidates.
- **Bacteriologic algorithm** (BA) is inspired by evolutionary ecology which is the study of living organisms in the context of their environment, with the aim of discovering how they adapt.
- **Biogeography-based optimisation** (BBO) is the study of geographical distribution of biological organisms.
- **Cultural algorithm** (**CA**) uses behavioural traits which are passed from generation to generation via a number of socially motivated operators.

Population-based techniques
2- Swarm intelligence

- **Ant colony optimisation** (**ACO**) algorithm is mostly suited to the optimal path finding problems and inspired by ants' natural foraging behaviour in which ants can find the shortest path from their nest to a source of food.
- **Particle swarm optimisation** (**PSO**) algorithm simulates the biological behaviours of bird flocks, inset swarms or fish schools in which each individual interacts with all or a part of the population to find a desirable path.
- **Bee colony optimisation** (**BCO**) algorithm is inspired by the biological behaviour of honeybees in collecting nectar.
- **Monkey search** (**MS**) is inspired by the behaviour of monkeys climbing trees to find food.

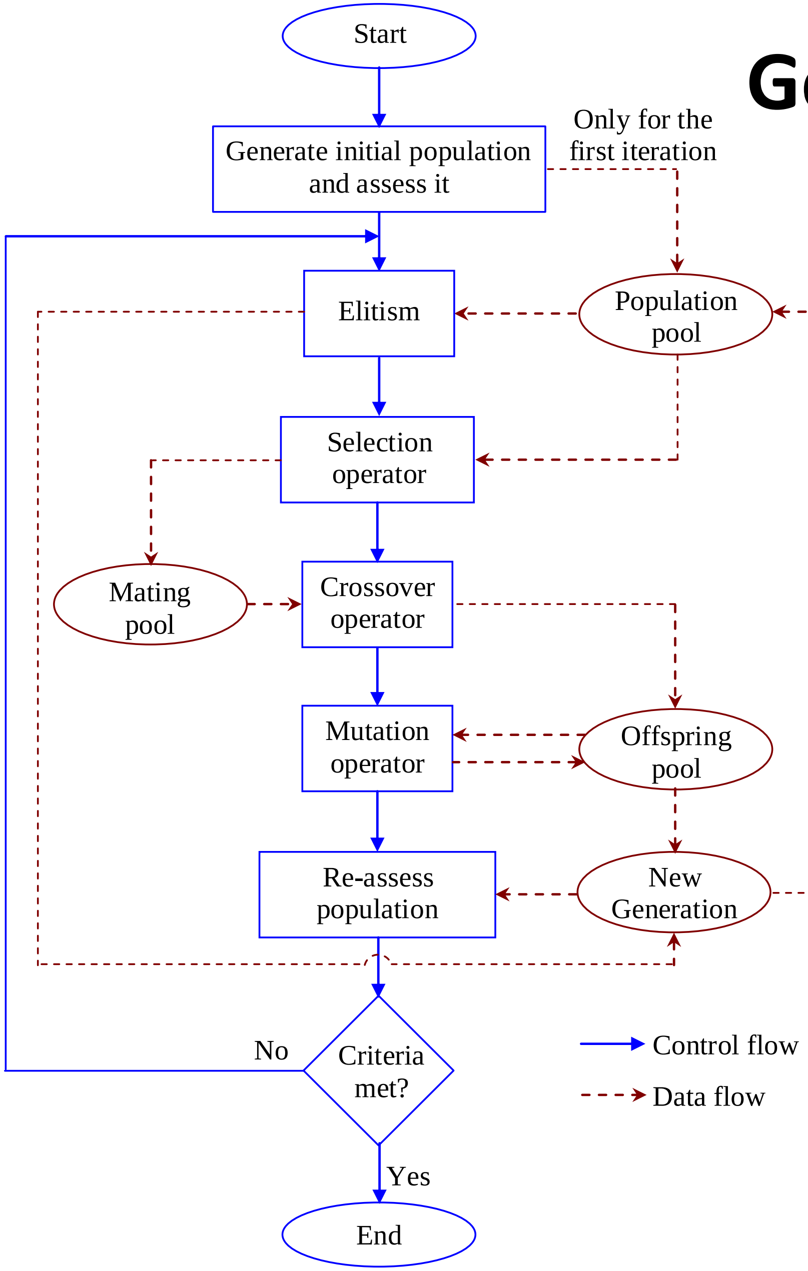


Population-based techniques
3- Natural/physical phenomena

- **Galaxy-based search** imitates the spiral arm of galaxies to search its surrounding.

- **Harmony search** (**HS**) is an optimisation approach mimicking the improvisation of music players where music players improvise their instrument's pitches for a better harmony.

Genetic Algorithms (GAs)



Operators:

1- Elitism

2- Selection

3-Crossover

4- Mutation

Genetic Algorithms (GAs)

- A GA comprises two basic elements: **a set of individuals**, i.e. potential solutions (the population) and **a set of biologically inspired operators** active over the population.

- A new set of solutions is created at each generation, by the process of **selecting** individuals according to their level of fitness in the problem domain and **breeding** them using the operators.

- This process leads to the evolution of populations of individuals that are **better suited** to their environment than the individuals that they were created from, just as in natural adaptation.

Genetic Algorithms (GAs)

- Let the **optimization problem** be specified as

$$\min_{x_i \in X_i} f_{\text{o}}(x_1, x_2, \ldots, x_m)$$

where $\mathbf{x} = [x_1, x_2, \ldots, x_m]$ is the decision variables vector, $f_{\text{o}}(\mathbf{x})$ is the objective function, m denotes the number of decision variables, X_i is the set of the allowable values for x_i, which may be represented as lower and upper bounds.

- An **initial population**, i.e. a set of initial values for each of the optimization variables, with N_{ind} individuals is generated, **assessed** and sent to the population pool.

Genetic Algorithms (GAs)

- The **elitism** is used to find a number (N_{elite}) of the **best individuals** and directly send them to the new generation pool.
- Using a **selection** process the mating pool is formed.
- The selection operator **selects better individuals** with a higher probability and it is highly possible to select the most powerful individuals more than one time.
- A **roulette wheel** selection is commonly used.

Genetic Algorithms (GAs)

- A **crossover** operator, e.g. **single-point crossover**, is applied on the mating pool with crossover rate of P_x to form the offspring pool.
- The population in the offspring pool may be slightly affected by a **mutation** operator with probability of P_m in order to explore a wider search space.
- The resultant of the offspring pool is sent to the new generation pool along with those individuals have already been selected by the elitism.
- The individuals of the new generation need to be **assessed**.
- If the predefined **criteria** are met, then the best individual of the latest generation is selected as the final solution of the optimisation problem; otherwise, the new generation is sent to the population pool and the procedure is repeated until the criteria are met.
- Normally one of the criteria is the **number of generations**.

Particle Swarm Optimization (PSO)

- PSO is one of the commonly used metaheuristic optimization techniques inspired by the **group behavior and integrated movements of some creatures**, such as birds and fish, especially to find food.
- Integrated movements provide various **advantages** for the creatures; for instance, **finding food easier and faster, better protection during predators attack**.
- In this form of life, the status of each individual is a function of the position of other **neighboring members** and the general position of the **swarm** that causes the optimum movement for all members according to the objectives of the group.
- In the PSO algorithm, the position of each member is updated based on the information stored in the memory of **that member** and also the **general group movement**.

Particle Swarm Optimization (PSO)

- Each group has a **leader** that is trying to lead other members.
- This phenomenon can be observed in a procession of some kinds of **migratory birds**.
- The **best solution of the PSO algorithm** in each epoch is obtained ($Gbest$) and chosen as the leader of the group.
- Also, the memory of each member is updated to find the **best solution of that member** ($Pbest$).
- Using these two values, the **velocity** of each member is calculated using the following relation:

$$V_i^{k+1} = \omega V_i^k + rand\, C_1 (Gbest^k - X_i^k) + rand\, C_2 (Pbest_i^k - X_i^k)$$

where X_i^k and V_i^k are, respectively, the position and velocity of the i^{th} particle in the kth epoch. C_1 , C_2 and ω are the specific parameters of the PSO algorithm

Particle Swarm Optimization (PSO)

- Having calculated the **velocity** of each particle, the **new position** of the particle is calculated using the following equation:

$$X_i^{k+1} = X_i^k + V_i^{k+1}$$

- It has been experimentally shown that for the sake of a **fast convergence**, the selection of the coefficients must be within the following inequalities:

$$C_1 + C_2 \leq 4 \qquad C_1 > 0 \qquad C_2 > 0 \qquad 0.4 \leq \omega \leq 0.9$$

- In general, the optimal values of these coefficients are **case dependent** and must be calculated individually by trial and error for each optimization problem.
- However, several approaches have been reported for **optimal selection** of these coefficients.

Particle Swarm Optimization (PSO)