Nature-inspired Optimization Algorithms Applied to Control Systems

Enrique V. Carrera

PROCONET Research Group

Department of Electrical Engineering

Universidad de las Fuerzas Armadas – ESPE

February 15, 2022

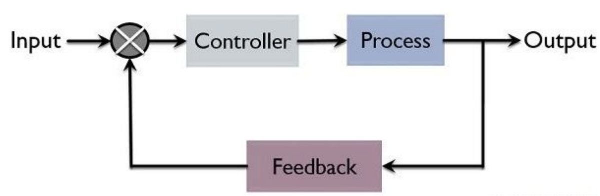
Outline

1. Introduction
2. Fuzzy Logic Controllers
3. Nature-inspired Algorithms
4. Application Examples
5. Conclusions

Introduction

Control Systems

- Currently, control systems are a central part of industry and automation
- A control system is a set of devices that regulates/manages other devices or systems through control loops
- Today, control systems are usually computerized by means of sensors, microcontrollers, PLCs, IEDs, etc.
- Control systems are used to enhance production, efficiency and safety in many areas



Parameter Optimization

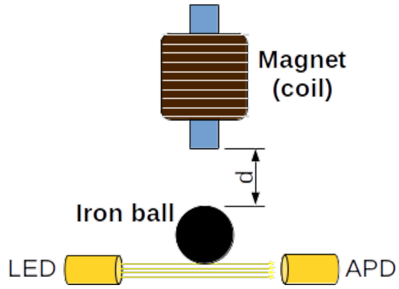
- Even simple control systems have certain parameters that must be adjusted to achieve the desired performance of the controlled processes
- There are some established methods for setting the parameters of certain types of control systems
- However, modern control systems, such as fuzzy logic controllers (FLCs), can have tens or even hundred of parameters
- Therefore, finding the optimal set of parameters to achieve the desired performance is a very difficult task
- The optimization of modern controllers has been carried out through some heuristic procedures with acceptable results but also high costs

Nature-inspired Optimization

- Various nature-inspired algorithms have been proposed to solve optimization problems in a wide range of applications
  - Genetic algorithms, differential evolution, firefly algorithm, particle swarm optimization, bat algorithm, ant colony optimization, cuckoo search, etc.
  - Image processing, job scheduling, chemical processes, operation and control of electric power systems, vehicle routing, control of autonomous vehicles, mobile networking, multi-objective optimization, etc.
- The selection of the controller parameters can be seen as an optimization problem that could be formulated in terms of a cost/objective function
- This talk is about the methodology to adjust the parameters of two FLCs through nature-inspired algorithms

Study Cases

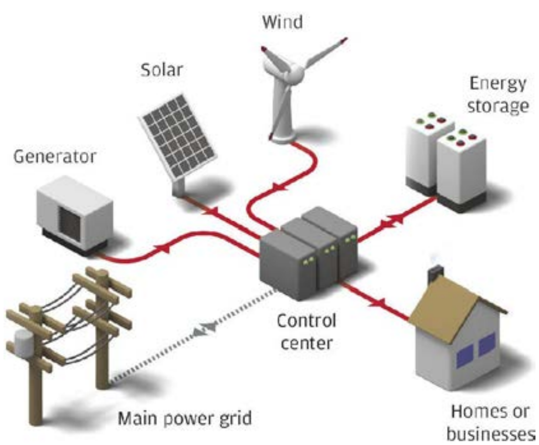
- Specifically, we optimize the controller parameters of:
  - A magnetic levitation system (MLS)
  - A microgrid energy management system (EMS)
- A magnetic levitator is a system formed by electromagnets that allow ferromagnetic objects to remain in the air



Introduction Fuzzy Logic Controllers Nature-inspired Algorithms Application Examples Conclusions

Study Cases

- The EMS of a residential microgrid aims to minimize power peaks and fluctuations on the power profile exchanged with the utility network



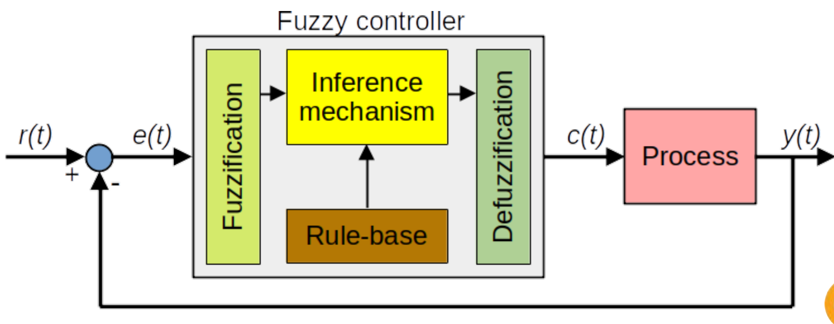
Fuzzy Logic Controllers

Fuzzy Logic Controllers

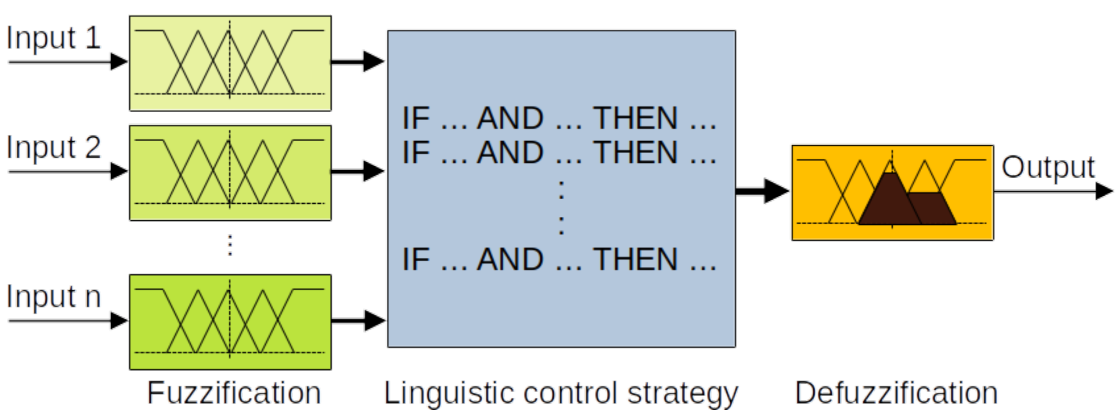
- In contrast to classical control theory that has been successful in controlling well-defined systems, many engineering problems require more sophisticated control techniques
  - Due to the presence of strong non-linearities, changing environments with uncertainties, or difficult to model systems
- FLC technique provides a formal methodology for representing, manipulating, computing and incorporating human intelligence directly into automatic control systems
- FLCs offer potential advantages over conventional control schemes
  - Less dependency on quantitative models, decision making in natural language, learning capability, greater degree of autonomy, easier and cheaper implementation, friendly user interface

Fuzzy Logic Controllers

- By using the linguistic approach provided by the fuzzy theory, human knowledge about the system to be controlled can be integrated into control theory
- FLCs cover a wider range of operating conditions and can be applied from a simple home heating controller to even large industrial control systems

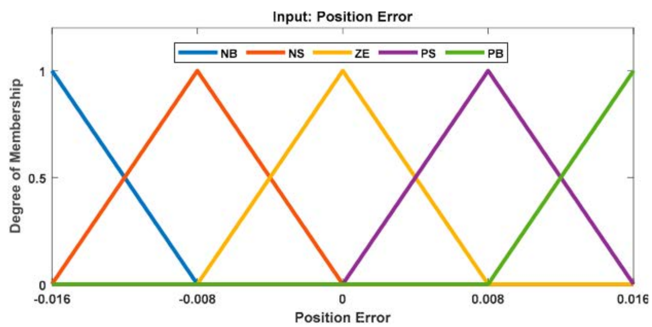


FLC Stages

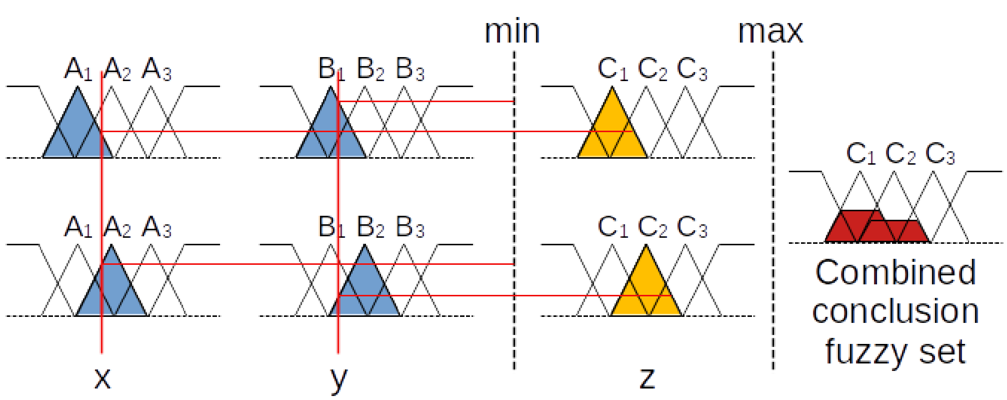


FLC Stages

- The fuzzification transforms the input variables in numeric format into linguistic variables
- The inference mechanism approximates the human reasoning to assign outputs according to the linguistic information



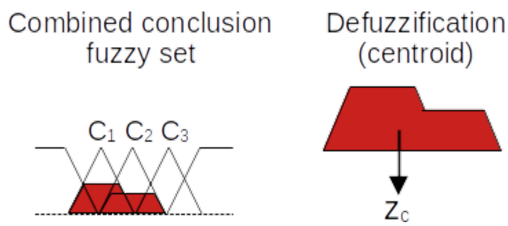
FLC Stages



Rule 1: IF x==A1 AND y==B1 THEN z=C1
Rule 2: IF x==A2 AND y==B2 THEN z=C2

FLC Stages

- The defuzzification converts the fuzzy output to a numerical value required to control the system



- Thus, the main parameters to optimize, in addition to the number, position and shape of the membership functions, are the rule database and the computing mechanisms

Nature-inspired Algorithms

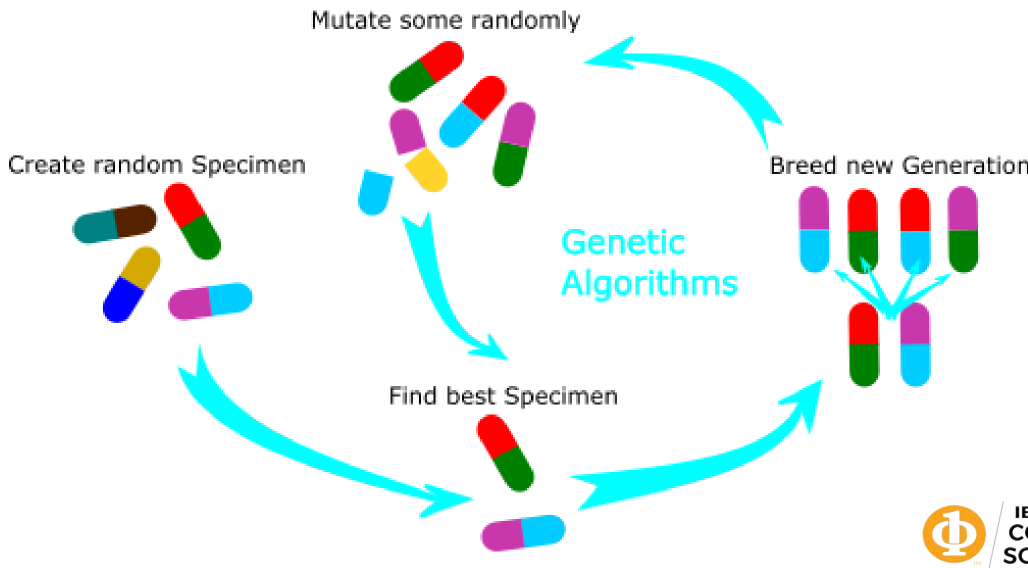
Nature-inspired Algorithms

- Even with the ever-increasing power of modern computers, it is still impractical and undesirable to use simple brute force approaches to optimize many real-world applications
- Nature-inspired algorithms tend to be global optimizers, using a swarm of multiple interacting agents to generate the search movements in the search space
- Such global optimizers are typically simple, flexible and yet surprisingly efficient
- However, these meta-heuristics do not guarantee that the optimal solution will always be found
- We have selected only a few featured algorithms to study the optimization of control systems

Genetic Algorithms

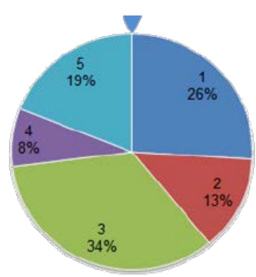
- Genetic algorithm (GA) is a model of biological evolution based on the theory of natural selection
- It is necessary to focus on its three main activities
  - Solution representations through a high-dimensional vector
  - Solution modifications by mutation and crossover operators
  - Solution selection through the fitness assigned to solutions
- The genetic operators form an essential part of GA as a problem-solving strategy
- There are many advantages of GAs over traditional optimization algorithms, for instance the ability to deal with complex problems and their innate parallelism

Genetic Algorithms



Genetic Algorithms

- A simple example: find the maximum of a two-dimensional function $g(x, y)$
  - Solution representation $\vec{x}_i = [x_i, y_i],\ i = 1, \ldots, N$
  - Crossover operator $C(\vec{x}_i, \vec{x}_j) = [x_i, y_j]$ and $[x_j, y_i]$
  - Mutation operator $M(\vec{x}_i) = [x_i + \epsilon_0, y_i + \epsilon_1]$
  - Solution selection $f(\vec{x}_i) = g(x_i, y_i)$ plus roulette wheel



Differential Evolution

- Like GAs, differential evolution (DE) is a method that optimizes a problem by interactively trying to improve a candidate solution with regard to a given measure of quality
- DE is used for multidimensional real-valued functions (*i.e.,* numerical optimization) but does not require the gradient of the problem being optimized
  - DE does not need that the optimization problem can be differentiable
  - DE can be used on optimization problems that are not even continuous, are noisy, change over time, etc.
- DE optimizes a problem by maintaining a population of candidate solutions and creating new solutions by combining the existing ones according to its simple formulas

Differential Evolution

- A candidate solution randomly selected at pseudo-time t is interactively updated using

$$\vec{x}_i^{t+1} = \vec{x}_i^t + \Delta\vec{x}_i^t$$

- The mutation operator $\Delta\vec{x}_i^t$ is calculated as

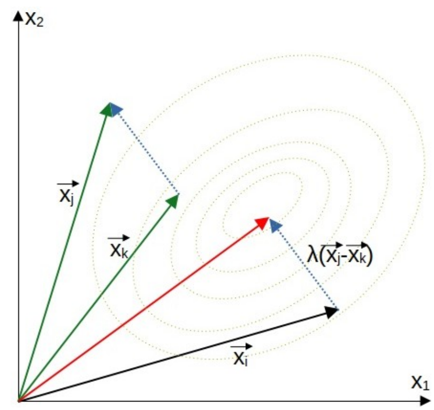
$$\Delta\vec{x}_i^t = \lambda(\vec{x}_j^t - \vec{x}_k^t)$$

- This mutation operator can have specific mutation rates for each vector component and $\vec{x}_i^{t+1}$ only replaces $\vec{x}_i^t$ if

$$f(\vec{x}_i^{t+1}) > f(\vec{x}_i^t)$$

- Some variations of DE also use a kind of separate crossover operator

Differential Evolution



- In summary, DE is a very effective global search algorithm with a simple mathematical structure

Particle Swarm Optimization

- Similar to previous algorithms, PSO is a stochastic and multi-agent parallel global-search technique that develops global optimum solutions based on local solutions
- The main inspiration of PSO comes from the swarming motion of bird flocks and schools of fish
- The potential solutions, called particles, move in the problem space following the current optimum particles
- Thus, the mutation operator considers not only the current positions of the particles but also their velocity
- The movements of the particles are guided by their own best-known positions in the search-space as well as the best-known position of the entire swarm

Particle Swarm Optimization

- The position and velocity of particle i at any iteration or pseudo-time t are interactively updated using

$$\vec{x}_i^{t+1} = \vec{x}_i^t + \Delta\vec{x}_i^t$$
$$\vec{v}_i^{t+1} = \vec{v}_i^t + \Delta\vec{v}_i^t$$

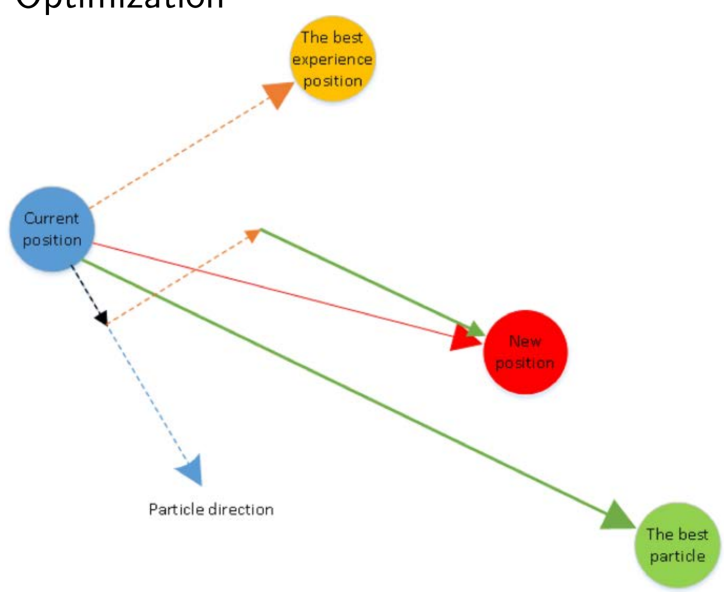
with

$$\Delta\vec{x}_i^t = \vec{v}_i^{t+1}$$
$$\Delta\vec{v}_i^t = \alpha\epsilon_1(\vec{g}^* - \vec{x}_i^t) + \beta\epsilon_2(\vec{x}_i^* - \vec{x}_i^t)$$

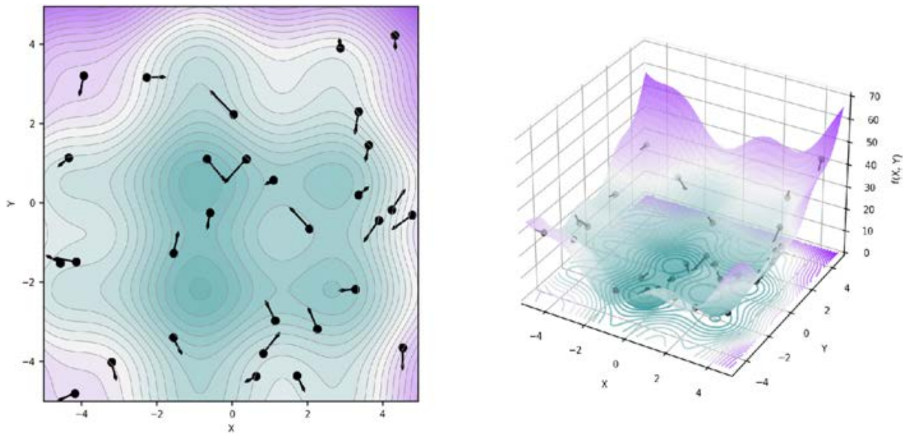
where ϵ_1 and ϵ_2 are random numbers evenly distributed in $[0, 1]$

- PSO is computationally efficient in terms of both speed and memory requirements

Particle Swarm Optimization



Particle Swarm Optimization



Finding the minimum of a two-dimensional function using PSO

Cuckoo Search

- The Cuckoo Search (CS) algorithm is a more recent alternative to nature-inspired optimization algorithms
- CS is based on the aggressive reproduction strategy of some cuckoo species and their interactions with host species such as warblers
- The main assumptions of this model are
  - Each cuckoo only can lay one egg at a time and dump it in a randomly chosen nest
  - The best nests with high quality of eggs will carry over to the next generations (elitism)
  - The number of available host nests is fixed
- The algorithm is also enhanced by the so-called Lévy flights rather than by simple isotropic random walks

Cuckoo Search

- The eggs laid by cuckoos (*i.e.*, the candidate solutions) can be discovered and abandoned with a probability p_a, realized by a Heaviside step function H with the use of a random number ϵ in [0,1]
- Any solution at iteration t is updated by

$$\vec{x}_i^{t+1} = \vec{x}_i^t + \Delta\vec{x}_i^t$$

and

$$\Delta\vec{x}_i^t = \alpha\vec{s} \otimes H(p_a - \epsilon) \otimes (\vec{x}_j^t - \vec{x}_k^t)$$

where the step size $\vec{s}$ is drawn from a Lévy distribution with an exponent λ

- The similarity of two eggs (solutions $\vec{x}_j$ and $\vec{x}_k$) can be roughly measured by their difference $|\vec{x}_j - \vec{x}_k|$

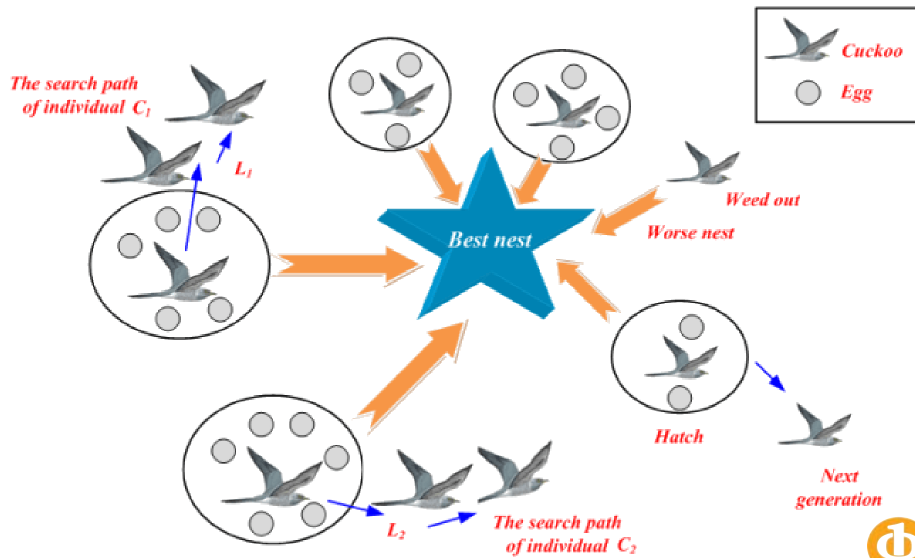
IEEE COMPUTER SOCIETY

Cuckoo Search

- Intensification and diversification search strategies are important elements of this algorithm
- *Intensification* focuses on the search in a local region where a good solution has been found, while *diversification* generates diverse solutions far enough from the good solutions to explore the search space on a global scale
- In essence, CS has strong operators at both local and global scales
- Recent studies show that CS is potentially more efficient than other algorithms
  - CS has well-balanced intensification/diversification search strategies
  - It is also a robust, precise and fast algorithm
- There are different variants of the CS algorithm that have been proposed in recent years

IEEE COMPUTER SOCIETY

Cuckoo Search



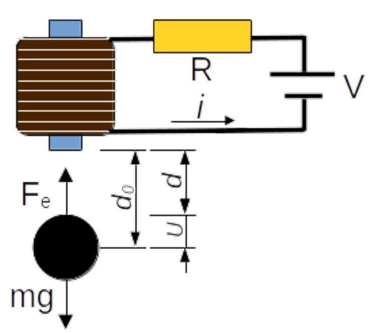
Other Nature-inspired Algorithms

- The *firefly algorithm* is based on the attraction and flashing behavior of tropical fireflies
  - It is a little bit similar to DE since each position is updated according to the attractiveness of a random pair of fireflies
- The *bat algorithm* is inspired on the echolocation of micro-bats and the associated frequency-tuning characteristics
  - Its formulation is similar to PSO since it represents the position and velocity of solutions, but the velocity is updated as a function of frequency ranges and loudness
- The *flower pollination algorithm* is based on the pollination processes and characteristics of flowering plants, including biotic and abiotic pollination as well as flower constancy

Application Examples

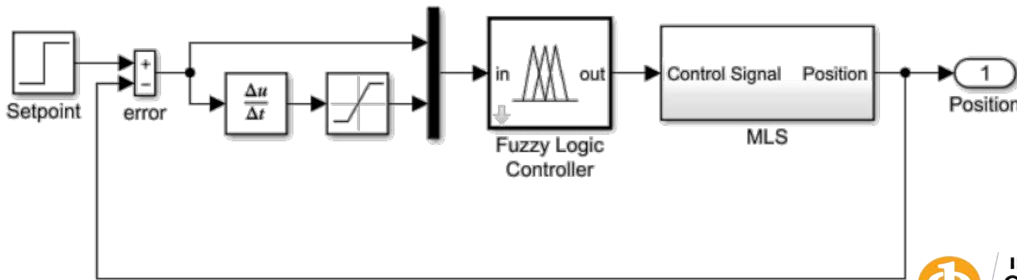
Magnetic Levitation System

- The MLS presented in the figure could be modeled through the variables:
  - Two inputs: the error position of the ferromagnetic sphere ($u = d_0 - d$), and its derivative (u')
  - One output: the variation in the coil current (i')



Magnetic Levitation System

- The proposed control system structure is shown below
  - Due to the non-linear dynamics presented by the MLS, a FLC is applied to keep the sphere in position d_0 (simple and robust non-linear control system)
  - The controller is easy to implement as it is based on the linguistic description of the global system behavior



Magnetic Levitation System

- The basic FLC uses symmetric and uniformly distributed membership functions considering 5 linguistic levels
- Therefore, the rule database includes the following 25 rules

		Position error				
		NB	**NS**	**ZE**	**PS**	**PB**
Derivative of error	**NB**	PB	PB	PS	PS	ZE
	NS	PB	PS	PS	ZE	NS
	ZE	PS	PS	ZE	NS	NS
	PS	PS	ZE	NS	NS	NB
	PB	ZE	NS	NS	NB	NB

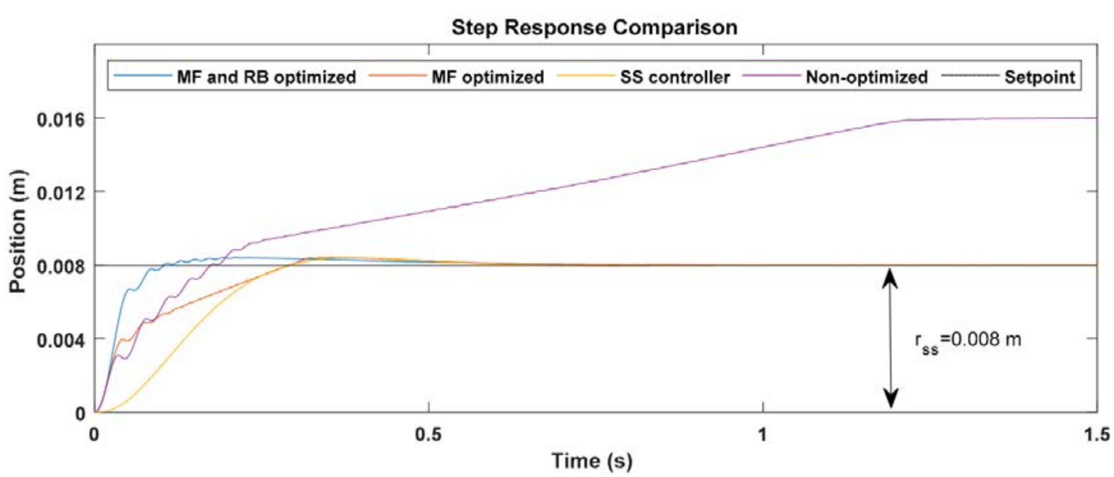
- The proposed FLC assumes a Mamdani-based inference and a center-of-gravity defuzzification method

Magnetic Levitation System

- In addition, the CS algorithm was applied to optimize the membership function positions and the rule database
- The objective/cost function takes into account the actual position error, the settling time of the step response and the maximum overshoot percentage
- We compare the step response of the MLS using
  - A classical state-space controller
  - The non-optimized FLC
  - The FLC with optimal membership functions
  - The FLC with optimal membership functions and rule database
- We have 33 variables to optimize for the membership functions and 25 for the rule database, and we use a fixed population size of 25 nests

IEEE COMPUTER SOCIETY

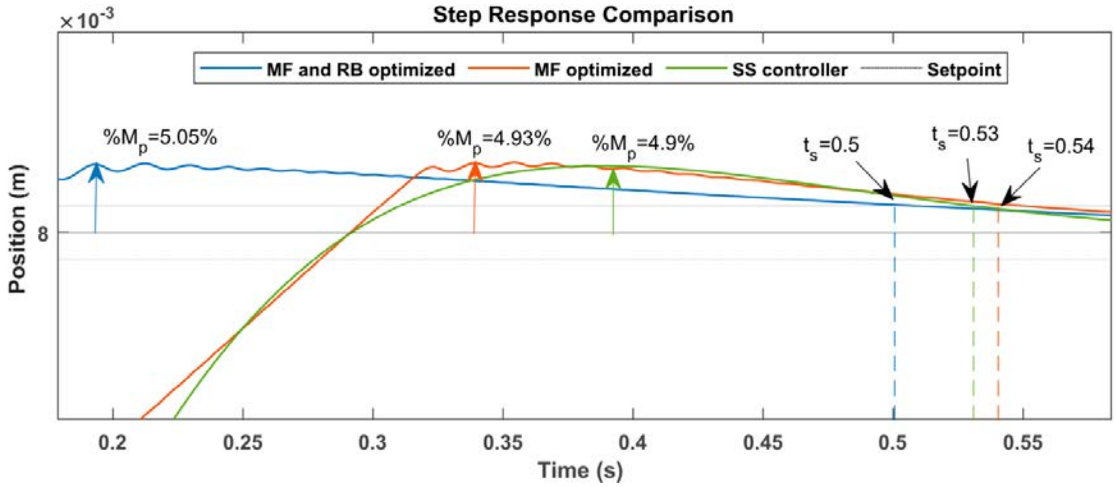
Magnetic Levitation System



Step response comparison of various control strategies

IEEE COMPUTER SOCIETY

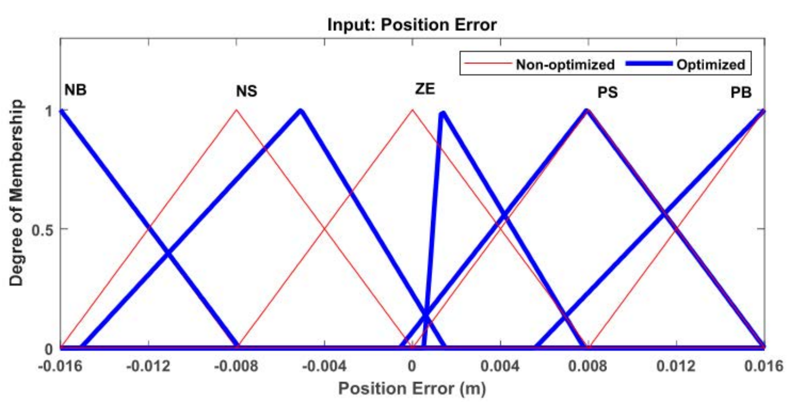
Magnetic Levitation System



Step response comparison of various control strategies

Magnetic Levitation System

- The proposed approach has successfully achieved the optimization goals such as settling time, overshoot and steady-state vertical position
- For that, the membership functions have been redefined



Magnetic Levitation System

- The rule database has also been updated (15 rules)

		Position error				
		NB	**NS**	**ZE**	**PS**	**PB**
Derivative of error	**NB**	PB	PB	PB	PS	PS
	NS	PB	ZE	PS	NS	NB
	ZE	ZE	NS	NS	NS	NB
	PS	NB	NS	NB	NB	NB
	PB	NB	NS	NB	NB	NB

- The cost function of each type of controller is summarized as

Type of controller	Cost
MF optimized	4.7×10^{-3}
SS controller	4.0×10^{-3}
MF and RB optimized	0.5×10^{-3}

IEEE COMPUTER SOCIETY

Energy Management System

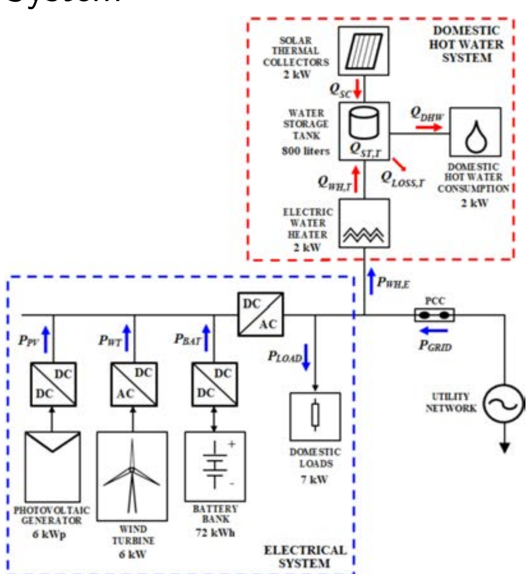
- Let's analyze a more complex FLC: an EMS for a residential grid-connected microgrid
- The objectives pursued by an EMS are usually economic, related to minimize the operating costs or maximize its income
- However, due to new regulations from the network operators, current objectives include to minimize power peaks and fluctuations in the power profile exchanged with the utility network
- Therefore, the fuzzy-based energy management strategy for the studied residential microgrid considers energy state forecasting, peak shaving, and demand management

IEEE COMPUTER SOCIETY

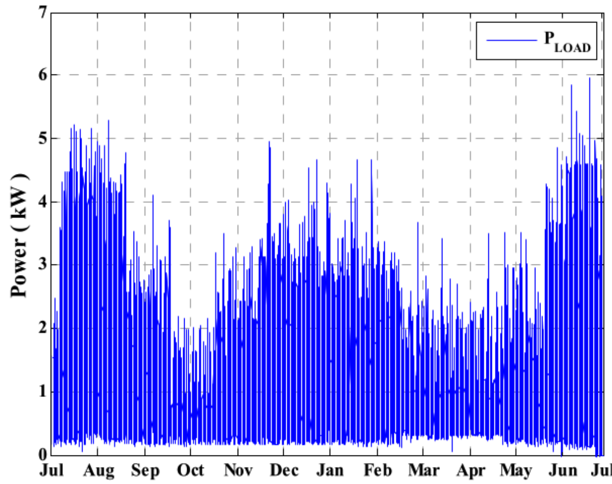
Energy Management System

- As mentioned, the main benefit of FLCs is that qualitative knowledge of the desired system behavior can be included as *if/then* linguistic rules
- However, the design of the FLC block involved in these EMSs requires the selection of several parameters such as the type and number of membership functions, the rule database, and the variation range of each variable
- The following evaluation is performed considering real recorded and predicted data of renewable energy source electricity production and load demand during one year
- The electro-thermal microgrid architecture considered in this work is shown in the next figure

Energy Management System



Energy Management System



Domestic load demand considered for the one-year simulation

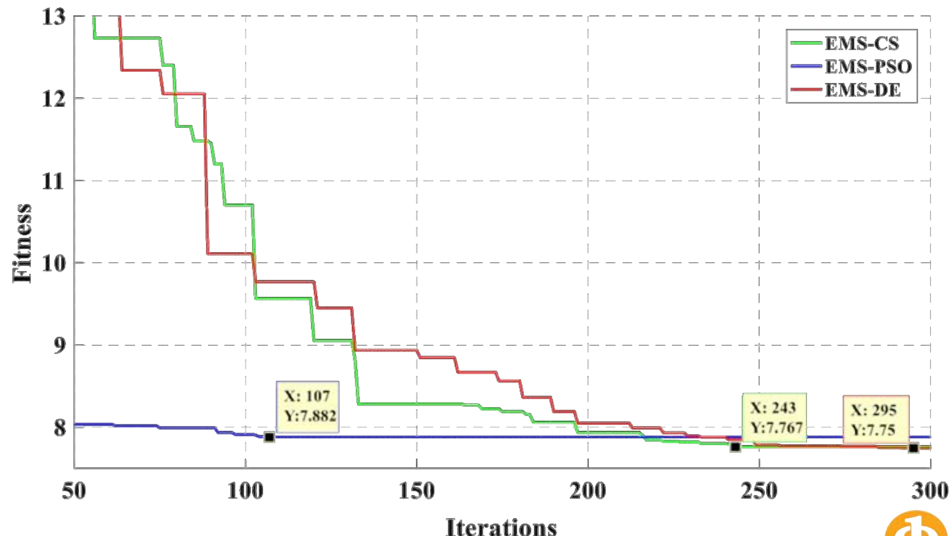
Energy Management System

- The FLC of the EMS has two inputs, one output, and 25 rules that have been developed and validated experimentally
- The inputs of the FLC are the current battery state-of-charge and the power forecast error for the previous 3-hours
- The single output of the FLC is used to smooth the grid power profile
- The basic FLC uses symmetric and uniformly distributed membership functions considering 5 linguistic levels for the inputs and 9 levels for the output
- The total number of variables to optimize is 72
- The FLC also assumes a Mamdani-based inference and a center-of-gravity defuzzification method

Energy Management System

- The overall cost function includes all criteria used to assess the quality of the grid power profile
- Numerical simulations on the specific microgrid example are presented to compare the performances of three nature-inspired optimization algorithms: PSO, DE and CS
- In addition, we benchmark the performance of an EMS based on the same FLC but tuned off-line using a trial and error methodology
- In terms of computing time, the PSO algorithm is faster than CS and DE algorithms
- But the CS algorithm offers some advantages related to the reduction of power fluctuations and peaks in the grid power profile

Energy Management System



Energy Management System

- The final cost associated to each evaluated system is

EMS Strategy	Cost
Original EMS	9.00
PSO optimized EMS	7.88
CS optimized EMS	7.76
DE optimized EMS	7.75

- Results demonstrates that the nature-inspired optimization algorithms obtain an acceptable response in relation to the main objective pursued by the EMS
- Since we have evaluated a simple and also a more complex control system with successful results, we hope this methodology can be applied to many other systems

IEEE COMPUTER SOCIETY

Conclusions

IEEE COMPUTER SOCIETY

Conclusions

- A procedure to optimize the parameters of a FLC using nature-inspired algorithms has been presented
- The main advantage of this approach is that little or no expertise is needed for tuning the FLC parameters
- Numerical simulations have made it possible to compare the performance of various nature-inspired optimization algorithms
- The obtained results highlight that the proposed algorithms can be applied to any system where the FLC technique is used, in order to optimize an objective function that involves several system parameters
- Different nature-inspired optimization algorithms present similar results

IEEE COMPUTER SOCIETY

Conclusions

- Convergence time is a critical parameter that should be considered in any deployment, especially if a real-time implementation is required
- The set of input parameters that each algorithm needs must be understood to provide stability and robustness to the optimization task
- Future work will focus on the design of a real-time adaptive fuzzy-based energy management system
  - The new design will update the FLC parameters at the same time as the forecast variables, which could improve the EMS efficiency
- We are also looking for a simple method to adjust the variables of the nature-inspired optimization algorithms
- Finally, we also want to evaluate other type of control systems where these nature-inspired heuristics can be employed

IEEE COMPUTER SOCIETY

References

1. Arcos D. *et al.* "A comparison of fuzzy-based energy management systems adjusted by nature-inspired algorithms." *Applied Sciences*, 11(4), 1663, Nov 2021
2. Garcia G. *et al.* "The cuckoo search algorithm applied to fuzzy logic control parameter optimization." In *Nilanjan Dey (editor), Springer Tracts in Nature-Inspired Computing*, Springer, Ch. 8, pp. 175–206, Jun 2020
3. Arcos D. *et al.* "Adjustment of the fuzzy logic controller parameters of the energy management strategy of a grid-tied domestic electro-thermal microgrid using the cuckoo search algorithm." In *Proc. of IECON'19*, pp. 115–121, Oct 2019
4. Garcia G. *et al.* "Fuzzy logic controller parameter optimization using metaheuristic cuckoo search algorithm for a magnetic levitation system." *Applied Sciences*, 9(12), 2458, Jun 2019

IEEE COMPUTER SOCIETY

evcarrera@espe.edu.ec

IEEE COMPUTER SOCIETY