Pontifícia Universidade Católica
do Rio de Janeiro

Walisson Chaves Ferreira Pinto

Ensemble Grey and Black-box System Identification for Friction Models

Dissertação de Mestrado

Dissertation presented to the Programa de Pós–graduação em Engenharia Mecânica of PUC-Rio in partial fulfillment of the requirements for the degree of Mestre em Engenharia Mecânica.

Advisor: Prof. Helon Vicente Hultmann Ayala

Rio de Janeiro
March 2021

Pontifícia Universidade Católica
do Rio de Janeiro

Walisson Chaves Ferreira Pinto

Ensemble Grey and Black-box System Identification for Friction Models

Dissertation presented to the Programa de Pós–graduação em Engenharia Mecânica of PUC-Rio in partial fulfillment of the requirements for the degree of Mestre em Engenharia Mecânica. Approved by the Examination Committee:

Prof. Helon Vicente Hultmann Ayala
Advisor
Departamento de Engenharia Mecânica – PUC-Rio

Prof. Hans Ingo Weber
Departamento de Engenharia Mecânica – PUC-Rio

Prof. Roberto Zanetti Freire
Escola Politécnica - PUCPR

Rio de Janeiro, March the 24th, 2021

Walisson Chaves Ferreira Pinto

Bachelor in Mechanical Engineering, graduated from the Pontifical Catholic University of Minas Gerais (PUC Minas) in 2017.

Bibliographic data

Pinto, Walisson Chaves Ferreira

Ensemble Grey and Black-box System Identification for Friction Models / Walisson Chaves Ferreira Pinto; advisor: Helon Vicente Hultmann Ayala. – 2021.

123 f: il. color. ; 30 cm

Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Engenharia Mecânica, 2021.

Inclui bibliografia

1. Engenharia Mecânica – Teses. 2. Identificação de Sistemas. 3. Algoritmos Evolucionários. 4. Conjuntos. 5. Modelos de Atrito. 6. Modelos Substitutos.
I. Ayala, Helon Vicente Hultmann. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Engenharia Mecânica. III. Título.

CDD: 621

Acknowledgments

I would like to thank my mother Maria for her support over all these years. Without her help I wouldn't have made it this far. Thank you for your love for me.

I would like to thank my advisor, Prof. Dr. Helon Vicente Hultmann Ayala, for all the times he encouraged, helped and advised me. Thank you for all the teaching and patience over the past few years.

I would also like to thank all the friends I made during the last few years at PUC-Rio.

Thank you to PUC-Rio for the Master's scholarship provided.

This work has been supported by the National Council of Scientific and Technological Development of Brazil (CNPq) through the grant 164794/2018-2.

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

Abstract

Pinto, Walisson Chaves Ferreira; Ayala, Helon Vicente Hultmann (Advisor). **Ensemble Grey and Black-box System Identification for Friction Models**. Rio de Janeiro, 2021. 123p. Dissertação de Mestrado – Departamento de Engenharia Mecânica, Pontifícia Universidade Católica do Rio de Janeiro.

The mathematical abstraction of a physical process is essential in engineering problems, as it can often be impractical or impossible to perform experiments on the real system. Besides, mathematical models are more flexible than physical prototypes, allowing for quick refinement of system designs to optimize various performance measures. The applications of the models can be divided into four parts, namely: design, estimation, control and monitoring. Some specific applications are i) simulations, ii) soft sensors, iii) performance evaluation, iv) statistical quality control and, v) fault detection and diagnosis. This work aims to: i) develop different classes of models capable of accurately simulating the output variable of a system, ii) evaluate the efficiency of optimization algorithms used in the parameter estimation task, iii) assess which friction model is the most appropriate to describe this phenomenon in a positioning system. The results showed that the friction in the positioning system presents a nonlinear and asymmetric behavior since some terms of the friction models related to the positive and negative velocities are significantly different from each other. The final result of the optimization process that used a local search algorithm was highly dependent on the initial conditions and the number of estimated parameters, which increased the simulation error. However, better estimates of the output variable were achieved when this approach was combined with other models of different classes. Through this last approach, the relative error was reduced by more than 20%. The simulations performed with the parameters estimated by the evolutionary algorithms were more accurate, they were able to reduce the relative error by almost 30% when compared with the local search algorithm. Considering the second case study, the decision tree-based optimizer proved to be equally effective compared to evolutionary algorithms. The relative error of the simulations using the parameters estimated by these algorithms was less than 8%. Besides, the shape of the friction reconstructed in the second joint of the robotic manipulator through the parameters estimated by the algorithms is in accordance with the expected.

Keywords

Mechanical Engineering - Thesis; System identification; Evolutionary Algorithms; Ensembles; Friction Models; Surrogate Models

Resumo

Pinto, Walisson Chaves Ferreira; Ayala, Helon Vicente Hultmann. **Identificação de Sistema Conjunto Caixa-Cinza e Caixa-Preta para Modelos de Atrito**. Rio de Janeiro, 2021. 123p. Dissertação de Mestrado – Departamento de Engenharia Mecânica, Pontifícia Universidade Católica do Rio de Janeiro.

A abstração matemática de um processo físico é essencial em problemas de engenharia, pois muitas vezes pode ser impraticável ou impossível realizar experimentos no sistema real. Além disso, modelos matemáticos são mais flexíveis que protótipos físicos, permitindo um rápido refinamento dos projetos do sistema para otimizar várias medidas de desempenho. As aplicações dos modelos podem ser divididas em quatro partes, a saber: projeto, estimativa, controle e monitoramento. Algumas aplicações específicas são i) simulações, ii) *soft sensors*, iii) avaliação de desempenho, iv) controle estatístico de qualidade e v) detecção e diagnóstico de falhas. Este trabalho visa então: i) desenvolver diferentes classes de modelos capazes de simular com precisão a variável de saída de um sistema, ii) avaliar a eficiência dos algoritmos de otimização utilizados na tarefa de estimação de parâmetros, iii) avaliar qual modelo de atrito é o mais adequado para descrever esse fenômeno em um sistema de posicionamento. Os resultados mostraram que o atrito no sistema de posicionamento apresenta comportamento não linear e assimétrico, já que alguns termos dos modelos de atrito relacionados às velocidades positiva e negativa são significativamente diferentes um do outro. O resultado final do processo de otimização que usou um algoritmo de busca local foi altamente dependente das condições iniciais e do número de parâmetros estimados, o que elevou o erro de simulação. Entretanto, melhores estimativas da variável de saída foram alcançadas quando essa abordagem foi combinada com outros modelos de diferentes classes. Através dessa última abordagem o erro relativo foi reduzido em mais de 20%. As simulações realizadas com os parâmetros estimados pelos algoritmos evolucionários foram mais acuradas, eles foram capazes de reduzir o erro relativo em quase 30% quando comparados com o algoritmo de busca local. Considerando o segundo estudo de caso, o otimizador baseado em árvores de decisão se mostrou igualmente eficaz se comparado aos algoritmos evolucionários. O erro relativo das simulações usando os parâmetros estimados por esses algoritmos foi inferior a 8%. Além disso, a forma do atrito reconstruído na segunda junta do manipulador robótico através dos parâmetros estimados pelos algoritmos está de acordo com o esperado.

Palavras-chave

Engenharia Mecânica – Teses; Identificação de Sistemas; Algoritmos Evolucionários; Conjuntos; Modelos de Atrito; Modelos Substitutos;.

Table of contents

List of Figures

List of Tables

List of Abreviations

ACO – Ant Colony Optimization
AI – Artificial Intelligence
ANN – Artificial Neural Network
ARIMA – Auto-Regressive Integrated Moving Average
ARMA – Auto-Regressive Moving-Average
ARMAX – Auto-Regressive Moving-Average with Exogenous Inputs
ARX – Auto-Regressive with Exogenous Inputs
AS – Ant System
BBO – Biogeography-Based Optimization
BGBM – Black-Grex-Box Model
BJ – Box-Jenkins
BO – Bayesian Optimization
CART – Classification and Regression Trees
CO – Classical Optimization
CR – Crossover Constant
DE – Differential Evolution
DOF – Degree of Freedom
DNN – Deep Neural Network
DT – Decision Tree
EAs – Evolutionary Algorithms
EC – Evolutionary Computation
EHSS – Electro-Hydraulic Servo System
EI – Expected Improvement
EGBB – Ensemble Grey-Black-Box
EMPS – Electro-Mechanical Positioning System
F – Scale Factor
GA – Genetic Algorithm
GBM – Grey-Box Model
GP – Gaussian Process
HI – Health Indicator
HSI – Habitat Suitability Index
IDIM – Inverse Dynamic Identification Model
IDM – Inverse Dynamic Model

IGA – Improved Genetic Algorithm
IV – Instrumental Variable
LCB – Lower Confidence Bound
LLS – Logarithmic Least Squares
LS – Least-Squares
ML – Machine Learning
MLE – Maximum Likelihood Estimation
MPC – Model Predictive Control
NARX – Nonlinear Auto Regressive with Exogenous Inputs
NLS – Nonlinear Least-Squares
ODE – Ordinary Differential Equations
OE – Output Error
OSA – One-Step-Ahead
PI – Probability of Improvement
PIC – Proportional Integral Control
PID – Proportional Integral Derivative
PRS – Pure Random Search
PSO – Particle Swarm Optimization
RMSE – Root Mean Squared Error
RNN – Recurrent Neural Network
RP – Recursive Partitioning
SAS – Semi-active Suspension
SGA – Standard Genetic Algorithm
SI – System Identification
SIVs – Suitability Index Variables
SSE – Sum of Squared Errors
TSP – Traveling Salesman Problem
UCB – Upper Confidence Bound
WLS – Weighted Least-Squares

1
Introduction

High-precision motion control plays a key part in positioning and handling applications, it is used in computer numerically controlled machines, robot manipulators, XY plant, microelectronics assembly units, and electromechanical devices. Some specific applications are in semiconductor manufacturing, laser and materials processing, optical inspection systems, additive manufacturing, or industrial digital printing [1]. The compensation of nonlinearities plays an important role in high-precision actuators [2]. Friction is one of these nonlinearities, which is often responsible for the poor performance of their components [3]. Modeling and identification are key aspects to understand the friction and design controller so that the influence of friction can be overcome [1].

The concept of a system can be defined in such a broad way that most things in our environment will become systems: the solar system, a vehicle moving on the road, human speech generation. That is why the term system is used not only in engineering, but also in science, economics, sociology, and politics. It can be defined as a collection of components that act together to perform a certain objective, whose properties we want to study, i.e., it is a part of the universe that is of our interest [4, 5]. As part of the universe, a system interacts with its environment according to the variables that originate outside the system and are not directly dependent on what happens in it. On the other hand, some variables are generated by the system as it interacts with the environment. The former and the latter variables are referred to as inputs and output variables.

Often, we have many questions about the system or its subsystems (such as the quarter of the vehicle cited earlier) properties. In this context, by the analysis of the system, one can predict how it will respond to various inputs and how that response changes with different values of the system parameters. Many questions concerning a system can be answered by experimentation, but it has limitations such as cost, danger, or the system does not yet exist, which makes experimentation impracticable sometimes. Simplified versions of the systems, known as models, can be used to answer questions about them.

Models are present in our everyday life all the time and play an important

role. We can use a model to explain a variety of phenomena in our surroundings. It is a tool used to answer questions about a system without the need to experiment. In [4] models are classified into four different categories: mental, verbal, physical, and mathematical. When one is learning how to drive a car, he/she develops a mental model of the car's driving properties. A verbal system is qualitative, and it is used to describe the reaction of the system according to certain given stimuli. For instance, if one does not eat anything, then he/she will get hungry. A more formal example of verbal models is experts systems.

The remaining models try to imitate the system. Physical models, such as a small scale of a ship, aircraft, house are built to test and/or investigate some properties of the system under realistic conditions. In this work, the focus is on the mathematical models, which give the notion of how the variables of the system relate to each other.

1.1 System Identification

The aim of system identification (SI) is the development of an appropriate model (mathematical description) of a system based on observed data. SI plays a crucial role in the field of control system engineering since a controller will be designed based on a known mathematical or an estimated model so that a precise output response of the system can be satisfied [6]. The resulting model may have many applications such as simulations, optimization, predictions, controller design, fault detection and diagnosis [7]. Due to the importance and usefulness of SI many reading options on this subject are available [7–11].

An important part of the system identification procedure is concerned with the choice of the model structure. It should be used prior knowledge and physical insights about the system for the task of selecting the model structure [12]. According to the amount of prior knowledge, models can be classified as black, grey, and white-box. In the former model, one assumes an identification problem with an unknown model structure of a system, this model does not have any physical interpretation. It is built based on observed data alone, it means that no physical insight is available or used. In this case, the model describes the experimental data without any physical interpretation of its parameters [8, 12]. On the other hand, if one uses a priori knowledge in the model development process, assuming that the model structure is known, it is called a grey-box model, which is physically meaningful, i.e., it has some correspondence to the physics of the process. In this case, parameters must be determined from observed data though [13]. In the latter, the model is perfectly known, and its structure and parameters are entirely determined based on first

principles.

After the selection of a candidate model, the next step is its estimation, which is the parameter estimation phase. It is essentially an optimization problem that uses the given data to estimate the parameters of the model. There are a variety of estimation methods available: the Method of Moments, Least-Squares (LS), Maximum Likelihood Estimation (MLE), Bayesian methods, and instrumental variable methods. Some of them are commonly used for standard parameter estimation, but they lack robustness in the search for global optimum if the search space is not differentiable or linear in the parameters, since they are essentially local search techniques that search for the optimum by using some gradient-based method. It means that they may get stuck in a local optimum without finding a global optimum [14]. Different classes of algorithms have been also used for the task of system identification, such as artificial neural networks (ANN) [15–19], Evolutionary Algorithms (EAs) [20–24], swarm intelligence [25–29].

As mentioned before, SI is used for a variety of purposes, including studies concerning systems with friction, which is an overly complicated phenomenon that has been studied over the centuries, but still, there is no universally accepted theory to clarify. Friction is associated with the resistance to the relative motion between two surfaces in contact and it is a complex nonlinear phenomenon [30]. Friction occurs in all mechanical systems and has a strong influence on their performance and behavior. The studies of friction models have been driven because of engineering needs since friction is an especially important variable in the design of drive systems, high-precision servo mechanisms, robots, pneumatic and hydraulic systems [31].

1.2 Literature Review

In the modeling procedure, the models to be determined traditionally range from black-box to white-box models. In the former, experimental data of the process are used to obtain an appropriate model without taking particular account of the physical system. It means that the parameters of the model are viewed as vehicles for adjusting the fit to the data, not reflecting physical considerations in the system. In the latter, the structure and parameters are entirely determined based on first principles. In this case, the model is based on physical laws and additional relationships with corresponding physical parameters. There is an intermediate model category called grey-box in which the object of modeling is predefined and a priori information is used to determine unknown parameters. In this case, the models have adjustable

parameters with physical interpretation, since some of them are uncertain or not well known and if one desires to obtain realistic predictions the parameters can be estimated from the data [9, 11, 13].

In SI grey-box models have a wide field of applications. They can be used for modeling building thermal behavior [32], describing pipe (or duct) temperature [33], modeling of residential heating, ventilation and air conditioning system [34], and modeling of homogenous charge compression ignition engines [35]. Several contributions of grey-box identification applied to mechanical systems can be found in the literature.

In [36] the authors proposed an identification procedure that uses intermediate local models. The work addresses the problem of estimating unknown parameters in a nonlinear grey-box model. Two standard parameter estimators, namely nonlinear Least-Squares and logarithmic Least-Squares (NLS and LLS), as well as the selection of weights in the estimators, have been studied. The procedure was successfully applied to the problem of identifying elasticity parameters in a six-axis industrial robot since the identified model gives a good global description in the frequency range of interest, moreover, the obtained model should be useful for many future purposes. In the work of [37] a model predictive control (MPC) for the force tracking control of an electro-hydraulic servo system (EHSS), which is highly nonlinear, is presented. The authors built a nonlinear mathematical model, which parameters are identified using the trust region reflective Newton method of nonlinear Least-Squares. A linearized version of the nonlinear state-space model acquired via the grey-box system identification method at the initial operation point is adopted for the MPC control of force tracking. The control performance was compared between and among proportional integral control (PIC), MPC, and hybrid MPC–PIC controller, which achieved the most robust performance in the simulation and experiment.

A very popular technique used for identifying robot parameters is the Inverse Dynamic Identification Model (IDIM), which provides the joint force in terms of the joint position, velocity and acceleration. The authors of [38] performed the identification of an industrial collaborative robot based on the classical IDIM Least-Squares (IDIM-LS) method. They have noticed that the identified inertial and friction parameters using the Least-Squares method had unrealistic values. To deal with this issue the authors performed a numerical optimization based on the quadratic error between the measured and estimated torques. Another criterion was added because the inertia matrix of the robot must be strictly positive. In general, the parameters were identified with sufficient accuracy, but the inaccuracy of some parameters raised by the fact

that the authors used a simplified friction model used for nonzero velocities that considers dry and viscous friction in every joint.

In the work of [39], the IDIM formulation was used for the identification of a 1 Degree of Freedom (DOF) robot. Least-Squares optimization, Instrumental Variable (IV) optimization, and an Automated Instrumental Variable were applied for the task of estimating the parameters of an Electro-Mechanical Positioning System. The IDIM-LS approach is generally used for purposes of robot identification but it may lack robustness because of the closed-loop structure required for robot operation. The second approach tries to deal with this issue, but both rely on a priori knowledge or require tuning parameters of filters. In the last approach, the identification of the additive noise characteristics was included so that correct and lower variances of the IDIM parameters can be achieved in an automated way. The third method seems to be more appropriate to identify the dynamic parameters than the two others, showing to be robust against noises. In the work of [40], the authors identified the dynamic parameters of a 6 DOF industrial robot. They used a nonlinear weighted Least-Squares formulation (WLS) under nonlinear constraints in the dynamic parameters for the identification problem, this technique aims to prevent the effects of inaccurate data. The identification method was efficient since the parameters lead to a positive definite inertia matrix. There are other works on the subject of identifying dynamic parameters of robots, such as [41–43].

There are also many applications for the black-box approach. In [44] the authors used a nonlinear autoregressive with exogenous inputs (NARX) neural network to enhance cardiovascular rehabilitation therapies. The focus of the authors was the relationship between the required exercise (machine resistance) and the patient's heartbeat for an optimal training configuration. The model was efficient to reproduce the evolution of the heart rate in controlled cardiovascular aerobic training. In the work of [45], a NARX ANN model for forecasting groundwater levels of three different sorts of aquifers was applied. Forecasts of lead times up to half a year were conducted. The results of their studies have shown that the NARX ANN models are well suited to perform groundwater predictions for uninfluenced observation wells in all aquifer types studied. In [46] the authors introduced a time-dependent functional NARX methodology for the development of aircraft virtual sensors, the authors focused on the angle-of-attack for the main flight regimes. The main flight regimes (landing, take-off, clean flight) of a small commercial aircraft were considered for three different virtual sensor designs. Through a nonlinear 6 DOF simulation environment the performance of the developed

virtual sensors was tested, they showed to meet the design requirements since they achieved simulation errors lower than the required. In the work of [47] a hybrid of conventional backpropagation training algorithm for the NARX network and multi-objective differential evolution algorithm for identification of the nonlinear dynamics of an un manned small-scale helicopter from experimental flight data was proposed. This approach was able to yield a set of Pareto-solutions with an optimal compromise between model accuracy and model complexity. In the work of [48] different physics-based and black-box approaches, such as Maxwell-Slip models and neural networks, were used to model the nonlinear dependence of pre-sliding and sliding friction forces on displacement and velocity. The models have shown a suitable capability of friction prediction. An ensemble of the best prediction models was also built, bringing performance improvement.

Employed in various fields of knowledge, population-based, nature-inspired or evolutionary algorithms (EAs) are mainly used to solve optimization tasks, which also include the SI parameter estimation procedure. These algorithms are a class of metaheuristic algorithms [49]. They are global search methods, which means that they do not search for a point that is only locally optimal. Differently from local optimization methods an initial estimate for the optimization variable, which can considerably influence the objective value of the local solution obtained, is not required [50].

These algorithms are useful in a variety of applications, such as nonlinear system identification [51], to detect structural damage [52], on the identification of hysteretic system [53], and others [54–56]. In the work of [57] ant colony algorithms, which are motivated by the pheromone-depositing behavior of ants, were successfully used to automatically search and optimize the parameters of a stochastic resonance method, which is employed for the diagnosis of failures in planetary gearboxes. An ant colony algorithm was also successfully addressed to the task of modal parameter estimation of dynamical systems in the work of [58].

In the work [59] a population-based algorithm has been used for the selection of the optimal health indicator (HI) that is applied to measure the amount of equipment degradation. A multidimensional binary differential evolution algorithm (DE) was chosen for this purpose. This approach was used for the prediction of the remaining useful life of a fleet of turbofan engines working under variable operating conditions being able to provide more satisfactory HIs compared to those found in the literature. The authors of [60] utilized a DE strategy for parameters estimation of an 8-DOF and a 20-DOF structural system. No prior knowledge of mass, damping, or stiffness

is considered and the effectiveness and applicability of the DE algorithm for structural parameters estimation have been investigated. Comparative studies show that the DE algorithm is superior compared to a particle swarm optimization algorithm (PSO) for hard unknown mass problems and almost as good as the modified genetic algorithm.

In the work of [61] a Genetic Algorithm (GA) with an automatically adjustable probability of crossover and mutation is used to search for the optimal PID controller parameters for a robotic excavator trajectory control. This strategy was used to avoid premature convergence and stagnation in looking for the global optimal solution of the standard GA (SGA). The improved GA (IGA) based PID tuning method has shown to be superior when compared with a standard GA and a Ziegler–Nichols-based PID controller method. Moreover, the IGA algorithm was effective for improving tracking accuracy. In the work of [62], an improved GA to optimize the design of a fuzzy PID controller was proposed. The method proposed by the authors aims to regulate the opening of a throttle valve. The authors have improved the initial population generation, adaptive selection, and genetic operators of the SGA. The proposed controller performance was superior in terms of rising time, adjustment time, overshoot, and steady-state error when compared with the traditional GA-PID, fuzzy adaptive PID, PSO-PID, and a PID controller based on Back-Propagation neural network. The IGA controller was able to significantly improve the performance of the wellhead back pressure control system.

Biogeography-Based Optimization (BBO) is a heuristic inspired by the science of biogeography used for optimization. It has been used for classification problems [63, 64], optimal design problems [65, 66], robot path planning [67], ANN hyper-parameter and structure optimization [68], and mainly for scheduling problems [69–73]. However, BBO has been rarely used for parameter estimation tasks [74].

Another common optimization algorithm that has been used for parameter identification is the Particle Swarm Optimization, which is inspired by the social behavior of bird flocks. Improvements to the standard PSO were proposed after its development [28, 75, 76], however, its standard version is still in use [77]. In the work of [78] the parameters of a Bouc–Wen hysteretic system are identified by two variants of the PSO algorithm. They were compared with other identification methods being able to outperform them, showing to be very accurate concerning all Bouc–Wen model parameters. The authors of [25] have applied a social-emotional PSO algorithm for the identification of the parameters of a planar 2-link manipulator. The proposed PSO algorithm has

proven to be more successful in terms of parameter accuracy than a GA and a standard PSO algorithm.

Surrogate models can also be used for optimization tasks. In this approach, a model acts as a model of a more expensive model by approximating its input-output responses [79]. Bayesian optimization and tree-based regression are examples of such a method. The former method is commonly used for hyperparameter optimization/tuning [80–83], such as in deep neural networks (DNN) [84]. Bayesian optimization is also used for autonomous process setup [85, 86], for system identification in biological systems [87], it is also applied to real-time control [88]. However, this method has been little explored in parameter identification tasks [89–91]. The authors of [92] applied a Bayesian probabilistic approach to a finite element model of a bridge to estimate the mechanical properties of its constituent materials and foundation soil stiffness. The tree-based algorithm is another surrogate model that can be used for control applications [93], forecasting [94], classification [95], and optimization [96] and prediction problems [97].

Ensemble systems have proven to be highly effective and versatile in real-world problem domains and applications [98]. Some examples of the application of the ensemble approach are load forecasting [99–101], output power forecast [102,103], and fault detection [104]. In the work of [105], a tribometer was used as an experimental rig for the modeling of dry friction using a NARX type shunting ANN model. They also constructed an ensemble model by combining the shunting neural network model with the physics-based dynamic nonlinear regression Maxwell slip model, which achieved good accuracy in modeling the overall dynamical behavior. The authors of [106] used three real data sets to explore a hybrid methodology that combines both ARIMA and ANN models, the aim was to take advantage of the unique strength of each model. The results indicate that the combined model can be an effective way to improve forecasting accuracy achieved by either of the models used separately.

1.3 Motivation

Factors such as the need for models in process analysis and automation and the practical limitations of the first-principles approach in developing models motivate the need for identification. The benefits provided by models are enormous, as they are applicable for a variety of process systems engineering, namely, design, estimation, control, and monitoring. As stated previously, because of safety and/or economic reasons, experimentation might be impracticable or even impossible. That is why models are used especially for simulation

and predictions, which are the main incentives of model development, as they offer a cost and time-effective, safe alternative to experiments.

Conventional search techniques are useful, but they are not robust in the task of estimating the parameters of a model. They are point-to-point methods, moving from a single point in the search space to the next using some transition rule to determine the next point. Their iteration is restricted on each datum received since they need new data to direct the search [14]. As an alternative, algorithms that search for the global optimum by employing mechanisms to search larger parts of the search space might be employed. EAs are examples of such an approach. EAs use a population of individuals, i.e., they search from a population, not a single point. It allows EAs to simultaneously evaluate many points in the search space and the assumption that the search space is differentiable or continuous is not necessary. But both local and global optimization techniques can be combined to form hybrid algorithms [107,108].

Over the past two decades, the idea of combining models/algorithms has received increasing attention from the computational intelligence and machine learning community. Ensemble systems have proven to be highly effective and versatile in real-world problem domains and applications, which motivates the use of such a method.

1.4 Research Objectives and Contributions

The general objective of this work is to develop different classes of models capable of accurately simulating the output variable of a system and to evaluate the efficiency of optimization algorithms used in the parameter estimation task. Besides, this study aims to assess which friction model is most appropriate to describe this phenomenon in a positioning system. The specific objectives are:

(i) Estimate the dynamic parameters of an electromechanical positioning system (EMPS) adopting different friction models in their asymmetric and symmetric versions;

(ii) Combine models from different classes, for example, grey-box and black-box, to more accurately simulate the output variable of the system;

(iii) Estimate the dynamic parameters of a robotic manipulator considering a nonlinear friction model with Stribeck effect;

(iv) Evaluate the efficiency of different algorithms in the task of estimating the dynamic parameters of the EMPS and the robotic manipulator;

(v) Evaluate different friction models, investigating which of their versions (asymmetric or symmetric) can better describe the friction behavior of the EMPS.

The contributions of this work are the following:

(i) Identification of the dynamic parameters of a positioning system using local and global search algorithms;

(ii) Development of a hybrid model capable of predicting the position of the load in a positioning system;

(iii) Combination of different models capable of predicting the output variable of a positioning system more accurately than individual models;

(iv) Performance evaluation of five evolutionary algorithms on the task of estimating the parameters of a positioning system;

(v) Comparison of four friction models in their asymmetric and symmetric versions;

(vi) Identification of the dynamic parameters of a TX40 robot considering a nonlinear friction model with Stribeck effect.

1.5 Dissertation Outline

The rest of the dissertation is structured as follows. Chapter 2 presents the general procedure of a system identification task, while Chapter 3 introduces basic concepts of optimization and some algorithms used to estimate parameters in a system identification process. Chapter 4 includes an overview of different friction models, indicating the difference between asymmetric and symmetric approaches. Two case studies are presented in Chapter 5. Chapter 6 is dedicated to presenting the contributions of this dissertation. The conclusions and suggestions for a future work are in Chapter 7.

2
System Identification Procedure

The aim of SI is the development of an appropriate model (mathematical description) of a system based on observed data from the system [7]. SI can be seen as the interface between the real world of applications and the mathematical world of model abstractions [109]. This makes techniques of system identification wide applicable since dynamic systems are abundant in our environment [9]. Figure 2.1 describes schematically a task in System Identification.

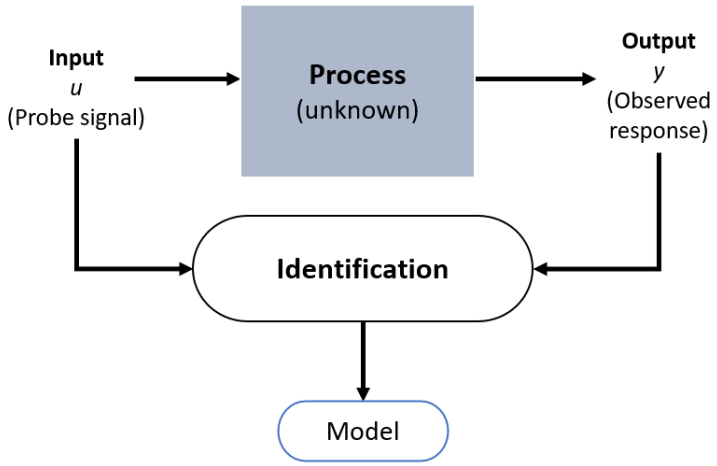

Figure 2.1: System identification is concerned with the development of mathematical models using the measured output of a system to a given input. The model should be a suitable (mathematical) description of that system. Adapted from [11].

Three general representations of an open system are shown in Figure 2.2. The notion of a system is a broad concept, a system is an object in which different kinds of variables interact at time and space scales and that produces observable signals [9, 11]. Both representations of [11] and [7] add a sensor box as a static element to emphasize the need of monitoring the systems to produce observable signals, i.e., this representation considers the sensor as being part of the dynamic system. [7] portrays a system as actuators associated with a process.

According to Figure 2.2, we may distinguish the variables of the system:

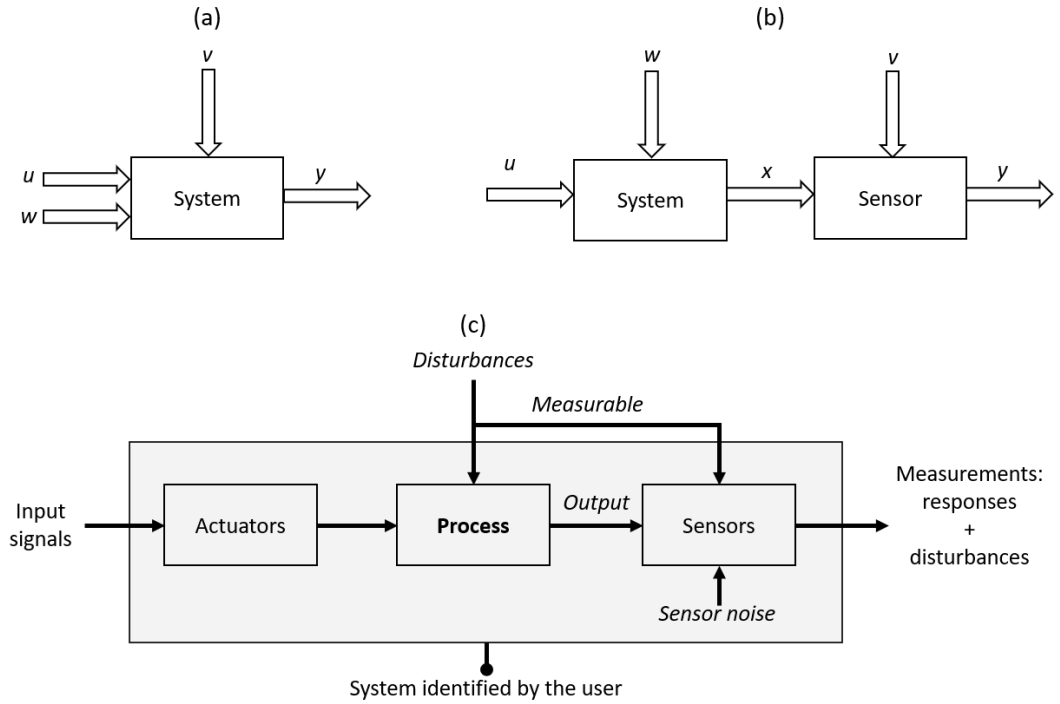

Figure 2.2: General representations of a system. A system has an input u, actuators, possible measured disturbance w, disturbance v, the system state x, and the output y of the system. These are signals that are measured and which one desires to predict, and the process. In *(b)* and *(c)* a sensor box acts as a static element to emphasize the need of monitoring the system. In *(c)* the system is even portrayed as actuators associated with a process. Adapted, respectively, from [9], [11], and [7].

(i) Input u: is an exogenous (external), measurable signal that can be manipulated directly by the observer;

(ii) Output y: observable signals that are of interest to the user, which are, in general, modeled as a function of the other signals;

(iii) State x: it summarizes all the effects of the past inputs u and disturbances w to the system;

(iv) Disturbance w: this disturbance is an exogenous, possibly measurable signal. It originates from the environment and cannot be manipulated, acting as external stimuli directly affecting the behavior of the system;

(v) Disturbance v: it is also an exogenous signal, which cannot be manipulated and represents the uncertainty (noise) measured;

(vi) Actuators: final control elements, which are responsible for exciting the process;

(vii) Process: It is what one wishes to identify.

The task of identification is the process of finding a mathematical model of a physical object, given a class of tentative models, and given a set of response data from experiments [110]. In the next few sections, we will discuss the identification procedure and some details of the different kinds of models, which are the relationship between observed quantities. They allow for the prediction of properties or behaviors of the object [109].

2.1 Procedure for Identification

System identification is approximate modeling for a specific application based on observed data and prior system knowledge [11]. That is why in a task of SI some points, such as *(i)* the data set recorded, *(ii)* the modeling phase, which is the construction of a class of tentative models, and *(iii)* establishing the best model are crucial [9, 110].

The procedure of system identification occurs iteratively, according to a natural logic flow. It can be resumed by data collection, choice of a model set, and the selection of the most appropriate one [9]. Figure 2.3 gives some details about this procedure, which is more detailed in Figure 2.4. As one may observe, prior knowledge plays a fundamental role in SI, since all processes are directly or indirectly dependent on it, as we can see by Figures 2.3 and 2.4.

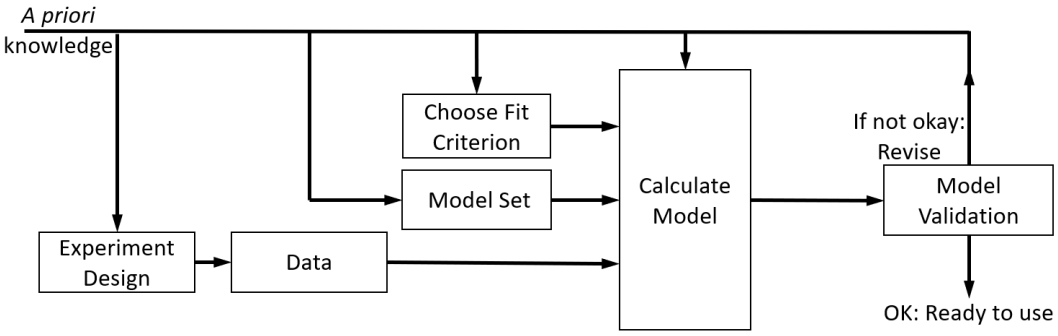

Figure 2.3: The system identification loop. Adapted of [9] and [11].

2.1.1 Data Generation and Acquisition

Input design is an identification-specific issue, it is all about the question of what kind of excitation is the best for a given identification problem. In SI a basic requirement is that the effect of the input in the measured output is larger than those caused by sensor noise/unmeasured disturbances [7]. It is justified because the accuracy of the estimated parameters will depend upon the signal input, that is why instead of choosing an input

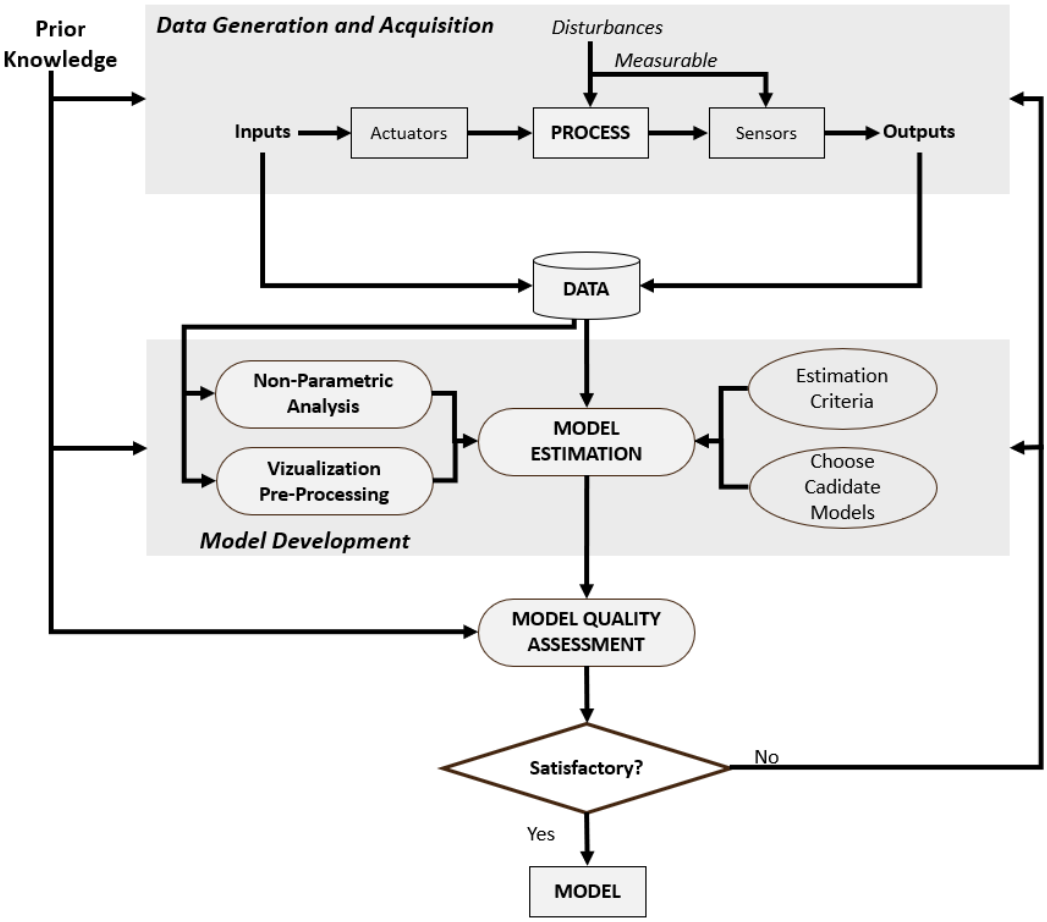

Figure 2.4: Flowchart of the system identification procedure. Adapted of [7].

arbitrarily one should choose an optimal input. It maximizes the sensitivity of the system output to the unknown parameters, consequently maximizing the parameter estimation accuracy [111]. Therefore, the objective of the experiment design/data generation and acquisition procedure is to make the right choices so that the data become maximally informative [9].

2.1.2 Data Pre-Processing

Pre-processing of data is required as a prior condition for the estimation phase since the data that has been collected, i.e., in its raw form, is usually not ready to be used for model development. That is why often the data must be submitted to quality checks and a pre-processing step before presenting it for use by the identification algorithms [7, 9]. Many factors are responsible for data deficiency:

(i) High-frequency disturbances in the data record;

(ii) Occasional bursts and outliers, missing data and non-continuous data records;

(iii) Drift and offsets, low-frequency disturbances, possibly of periodic character.

Checking the stationarity of data is an important step in the Pre-Processing phase since most identification methods require the statistical properties of data to remain invariant with time. Low-frequency disturbances, offsets, trends, drift, and periodic variations can be caused by external sources. The disturbances can be removed by explicitly pretreating the data or letting the noise model deal with them. Pre-filtering data is one approach for handling a variety of data characteristics such as drifts and noise. It can be also used to obtain preferentially accurate fits in selected frequency ranges.

The issue of missing data is common in several applications, it is due to a malfunction in the sensors, power disruptions, non-uniform sampling. On other hand, outliers may be also present, they do not conform to other parts of the data and can be detected in a residual plot. To deal with these issues one can cut out segments of the data sequence to avoid portions with bad data, but it may be quite difficult to merge the remaining segments. One can also replace the missing data appropriately by using the existing data.

2.1.3 Data Visualization

Data visualization should be considered as a key step in information extraction and signal analysis due to the immense value of information obtained from visual inspection of data at each stage of identification. In offline applications, it is crucial to first plot the data to inspect them for deficiencies cited in the previous subsection. By visual examination one may be able to identify the presence of drifts, outliers, and other peculiarities. It also provides an opportunity for the user to verify if the system was sufficiently excited by the input signal. One can also examine the input and output in the frequency domain, it is a tool to clarify the user about the spectral content and presence of periodicities in the signal. It is useful since it is possible to obtain a first-hand feel of the level of input excitation and the filtering nature of the system [7,9].

2.1.4 Model Development

When one interacts with a system the question is how its variables relate to each other, one might thus assume some relationship among the observed

signals, which is called a model of the system. A model is a map between the set of explanatory variables (inputs) and the set of predicted variables (outputs). The model is fundamentally a mathematical abstraction of the physical process [7, 9]. Figure 2.5 shows some details about the difference between a process of a system and a model.

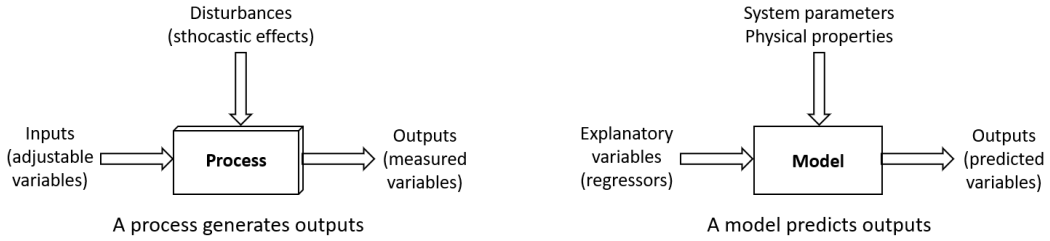

Figure 2.5: Variables that participate in the prediction are inputs to a model, while outputs of a model constitute those variables that we wish to predict or explain. Adapted of [7].

Mathematical models have different properties and ways of representation, they can take quite different forms depending on the system under study. One can have, for example, a discrete-time, first-principles, nonlinear model. Some aspects we might use to distinguish models are:

(i) Approach to modeling: first principles vs. empirical;

(ii) System characteristics, such as linear or nonlinear, time-varying or time-invariant;

(iii) The knowledge available to the user, e.g., deterministic or stochastic, black-box or grey-box;

(iv) The domain of modeling, such as continuous-time or discrete-time, time-domain or frequency-domain;

(v) Response characteristics, such as static or dynamic, lumped or distributed.

White-box models are developed from fundamentals using basic laws and constitutive relationships resulting in causal, continuous, nonlinear differential-algebraic equations. These models are highly effective and reliable, but simulations of these models require good numerical ODE and algebraic solvers. On the other hand, empirical models are built using measured data. Compared to the former models, these have the benefits of requiring in their development only a minimal knowledge of the process and that they offer flexibility in model structure, which is useful in several applications [7].

Empirical models can be divided into some categories. The first category makes a distinction based on whether the model possesses a specific structure chosen by the user or not, distinguishing them into parametric and non-parametric models. The former possesses a specific structure and order and is characterized by fewer parameters, while the latter do not possess any specific structure or order but is characterized by many unknowns. While non-parametric models can be estimated with minimal a priori knowledge, parametric models demand from the user some a priori knowledge [7].

The general model structure of a linear, time-invariant system [9] can be formulated as:

$$\mathbf{A}(z^{-1})\mathbf{y}(k) = \frac{\mathbf{B}(z^{-1})}{\mathbf{F}(z^{-1})}\mathbf{u}(k) + \frac{\mathbf{C}(z^{-1})}{\mathbf{D}(z^{-1})}\mathbf{e}(k) \tag{2-1}$$

where

$$\begin{aligned}
\mathbf{A}(z^{-1}) &= 1 + a_1 z^{-1} + \cdots + a_{nc} z^{-na} \\
\mathbf{B}(z^{-1}) &= b_1 z^{-1} + \cdots + b_{nb} z^{-nb} \\
\mathbf{C}(z^{-1}) &= 1 + c_1 z^{-1} + \cdots + c_{nc} z^{-nc} \\
\mathbf{D}(z^{-1}) &= 1 + d_1 z^{-1} + \cdots + d_{nd} z^{-nd} \\
\mathbf{F}(z^{-1}) &= 1 + f_1 z^{-1} + \cdots + a_{nf} z^{-nf}
\end{aligned} \tag{2-2}$$

$\mathbf{y}(k)$ and $\mathbf{u}(k)(k = 1, 2, \dots)$ are the system input and output signals, respectively, *na*, *nb*, *nc*, *nd*, and *nf* are non-negative integers indicating the orders of the relevant polynomials, they are usually referred to as the model orders; $\mathbf{e}(k)$ is a noise sequence which is assumed to be independently and identically distributed with zero mean and finite variance and z^{-1} is the backward shift operator. $\mathbf{A}(z^{-1})\mathbf{y}(k)$ is the autoregressive part (AR); $\mathbf{B}(z^{-1})\mathbf{u}(k)$ is the exogenous (X) variable ($\mathbf{B}(z^{-1})$ represents how the input interacts with the system); the factor $\mathbf{F}(z^{-1})$ accounts for the dynamics of the plant that is unique to it; $\mathbf{C}(z^{-1})$ of the noise model accounts for the moving average (MA) characteristics of the random process $\mathbf{e}(k)$; $\mathbf{D}(z^{-1})$ characterizes the auto-regressive behavior of $\mathbf{e}(k)$. The symbol z^{-1} denotes the backward shift operator, defined as $z^{-1}x(k) = x(k-1)$, where x can be any signal.

The model structure presented above is a parametric description of an input-output system, it is known as Autoregressive Moving Average with Exogenous Input Model. One or more of the five polynomials would be fixed to unit in some applications since this structure is too general for most application purposes. Some special cases of Equation (2-1) are summarized in Table 2.1. They are ARX (autoregressive with exogenous input) model, ARMAX (autoregressive moving average with exogenous input) model, ARMA (autoregressive moving average) model, FIR (finite impulse response) model, BJ (Box-Jenkins) model, and OE (output error) model.

Table 2.1: Classes of parametric models as special cases of Equation (2-1).

Polynomials used	Name of Model Structure	Polynomials values
AB	ARX	$C(z^{-1}) = D(z^{-1}) = F(z^{-1}) = 1$
ABC	ARMAX	$D(z^{-1}) = F(z^{-1}) = 1$
AC	ARMA	$B(z^{-1}) = 0;\ D(z^{-1}) = 1$
BD	FIR	$A(z^{-1}) = F(z^{-1}) = 1;\ C(z^{-1}) = 0$
BFCD	BJ (Box-Jenkins)	$A(z^{-1}) = 1$
BF	OE (output error)	$A(z^{-1}) = C(z^{-1}) = D(z^{-1}) = 1$

A nonlinear relationship between the input-output sequences gives much richer possibilities to describe systems [9]. A nonlinear black-box structure, such as neural networks, for a dynamical system, is a model structure that is prepared to describe virtually any nonlinear dynamics [12]. They can be used effectively for the identification and control of nonlinear dynamical systems since they can approximate large classes of nonlinear functions with sufficient precision. This makes them relevant candidates for use in dynamic models for the representation of nonlinear plants [112].

A neural network is a massively parallel distributed processor constituted of simple processing units, also referred to as neurons, which can store experiential knowledge and making it available for use [113]. Layers composed of such units are arranged so that data is entered at the input layer and passes through either one or several intermediate layers, which are also called hidden layers, before reaching the output layer, Figure 2.6 shows the schematic of a single-layer neural network with one hidden layer. The difference between the actual output and the target output of the network, the prediction error, is used to change the connection strengths between the nodes, this is the way the network is trained. By iterating, the weights are modified until the output error reaches an acceptable level, i.e., the network adjusts the weights so that the output pattern is reproduced [114].

The basic model with a single hidden layer network expresses the prediction as a weighted sum of non-linearly transformed regressors, it is expressed by Equation (2-3). Another class of a neural network, namely, recurrent neural network (RNN) is sensitive to historical data, enabling this network to have a dynamic memory function. Its schematic diagram is shown in Figure 2.7, which is a nonlinear ARX (NARX) neural network structure.

$$\widehat{y}[k] = \sum_{j=1}^{H} \beta_{1j} f_j \left(\sum_{i=1}^{p} w_{ji} x_i[k] \right) \tag{2-3}$$

where j = 1,..., H are the nodes in the hidden layer; x_i, i=1,..., p are the p regressors and w_{ji} is the synaptic weight of the respective neuron. The nonlinear function $f_j(\cdot)$ is known as the activation function.

The most important phase in a system identification procedure is the development of an appropriate model set since the success of an identification

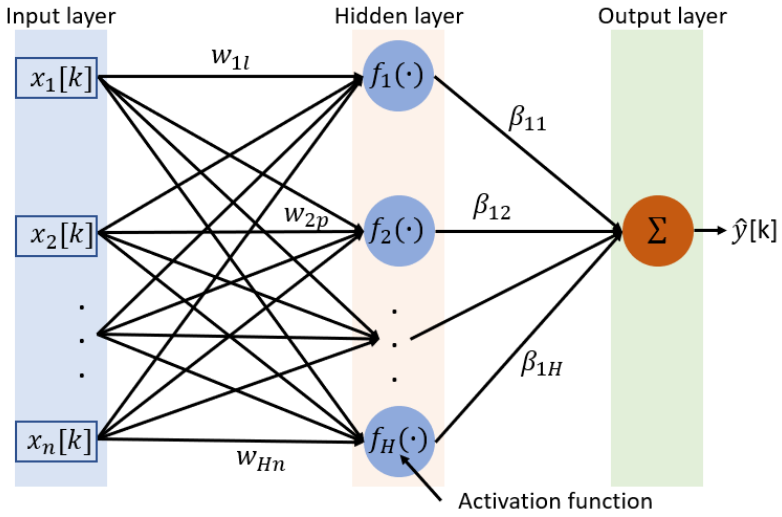

Figure 2.6: Schematic of a neural network with a single hidden layer.

application is directly dependent on the model. This phase is guided by our prior knowledge of the system, as we can see in Figures 2.3 and 2.4. The model development phase comprises two steps: (i) specifying a model structure and order or the set of models, and (ii) estimating the parameters of that model by solving the associated optimization problem [7].

One should give attention to some points for the choice of the candidate models:

(i) Quality of the model: It is related with the accuracy and precision of the model. One model should be selected so that both bias and variance are kept small, which are usually conflicting requirements. In other words, the model should be flexible enough offering good capabilities of describing different possible systems and, not using unnecessarily many parameters (the model should be parsimonious);

(ii) Price of the model: It is directly associated with the second step, i.e., with the price to calculate the model. In the second step, a criterion function must be specified as well as the identification method to solve the estimation problem. Two aspects influence this step, namely: *(i)* the algorithm complexity and *(ii)* the properties of the criterion function;

(iii) End-use of the model: The end application might impose requirements. According to the intent of using one can choose a white, grey, or black-box model. For instance, if the intended use is in control applications it usually suffices to use linear models. In this case, the model should be as simple as possible.

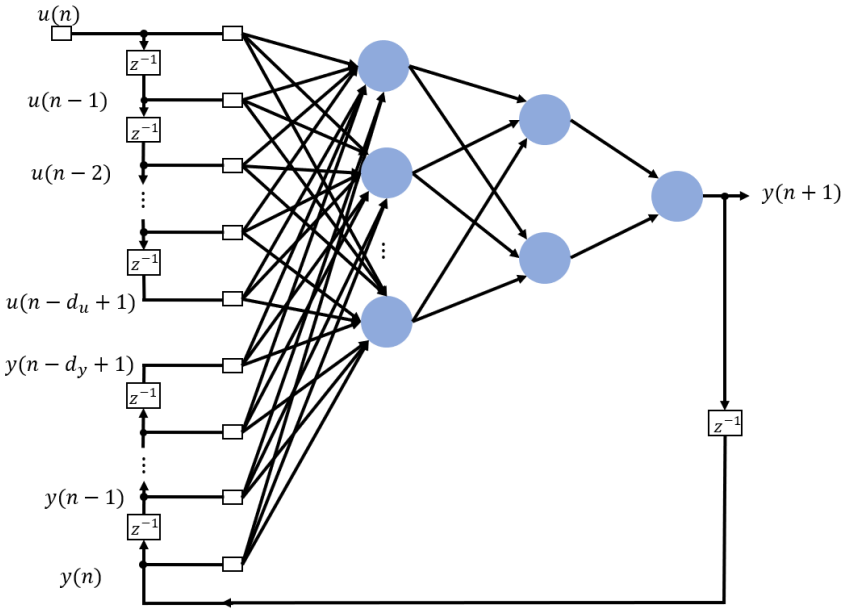

Figure 2.7: Typical NARX neural network structure. Adapted from [115].

After the selection of a candidate model, the next step is the parameter estimation phase. It is essentially an optimization problem that uses the given data, which is called estimation or training data, to estimate the parameters of the model [7]. There are a variety of estimation methods available: the Method of Moments, Least-Squares, Maximum Likelihood estimation (MLE), Bayesian methods, and instrumental variable methods. The Least-Squares method is particularly interesting because of its versatility.

Suppose we have a linear regression model structure (2-4), which is especially useful in describing basic linear and nonlinear systems [9].

$$y(k) = \phi_1(k)\theta_1 + \cdots \phi_p(k)\theta_p + e(t) \tag{2-4}$$

According to this model, the variable y is explained in terms of the variables $\theta_1, \cdots \theta_p$, called regressors, plus an unobserved error term e. It is assumed that the model has one unknown parameter ϕ_i for each explanatory variable, which may be known in advance, or which has been measured. The model (2-4) can be written in matrix notation (2-5), where $\mathbf{\Phi}$ is an $N \times p$ matrix with elements $\mathbf{\Phi} = \phi_j(k), j = 1, \cdots, p$:

$$\mathbf{y} = \mathbf{\Phi}\theta + \mathbf{e} \tag{2-5}$$

A reasonable way to estimate the unknowns from given data is by demanding that the prediction errors (residuals) are small, i.e., the predictions are collectively at a minimum distance from the observations of a variable:

$$\varepsilon(k) = y(k) - \phi(k)^T\theta \tag{2-6}$$

The estimation of the unknowns can be formally stated as the Least-Squares objective function $J(\theta)$. One will choose the parameter vector θ such that the sum of squared prediction errors is minimal:

$$\min \ J(\theta) = \sum_{k=1}^{N} \varepsilon^2(k) = \sum_{k=1}^{N} (y(k) - \phi(k)^T\theta)^2 = \sum_{k=1}^{N} (y(k) - \widehat{y}(k))^2 \qquad (2\text{-}7)$$

Equation (2-7) can be rewritten in matrix notation as:

$$\min \ J(\theta) = \varepsilon^T\varepsilon = (\mathbf{y}^T - \theta^T\mathbf{\Phi}^T)(y - \mathbf{\Phi}\theta) = (\mathbf{y} - \widehat{\mathbf{y}})^T(\mathbf{y} - \widehat{\mathbf{y}}) \qquad (2\text{-}8)$$

Solving 2-8 one may find the most efficient estimator of the parameters that minimize the sum of squared prediction errors, which is:

$$\widehat{\theta}_{\mathbf{LS}} = (\mathbf{\Phi}^T\mathbf{\Phi}^{-1})\mathbf{\Phi}^T\mathbf{y} \qquad (2\text{-}9)$$

The methods mentioned above, such as LS and MLE, are commonly used for standard parameter estimation, but they lack robustness in the search for global optimum if the search space is not differentiable or linear in the parameters, since they are essentially local search techniques that search for the optimum by using some gradient-based method. It means that they may get stuck in a local optimum without finding a global optimum [14].

2.1.5 Model Assessment and Validation

The model development phase is followed by the model quality assessment, which is a fundamental part of any model development exercise. It can be divided into model assessment and validation. From the previous procedure, one picks out a model within the model candidates to elucidate whether it has good agreement with the estimation data and whether it is good enough for its intended use. Positive feedback of this phase means that the model is considered appropriate and can be used, otherwise, the procedure must be repeated [9, 11, 109].

The first part of the analysis is concerned with how effectively the model has explained the output variations in the training data, i.e., if the model has captured the characteristics of training data with reasonable accuracy and reliability [7]. This procedure can be performed by two different tests, namely: *(i)* statistical analysis of residuals (part of the data that the model could not reproduce) and *(ii)* error analysis of estimates.

The second part of the procedure, known as the cross-validation test (or simply validation), assesses the predictive abilities of the model on a fresh

data set [7]. In practice, this process aims to ensure that the model is useful not only for the estimation data but also for other data sets of interest [109]. The purpose here is to evaluate the extrapolation capabilities of the model. Another common term for this process is call generalization.

There are short-term and long-term forecasts. The one-step-ahead (OSA) prediction, belongs to the former and is widely used in model estimation, while the infinite-step ahead prediction belongs to the latter, which is equivalent to simulation, it means that the actual past information is never used but the predictions themselves are used recursively to generate the long-range forecasts. One may use metrics such as the coefficient of determination R^2 (2-12) and its variants for determining the degree of fit of predictions, which is one of the central aims of modeling.

The prediction problem refers to the case when the information is available up to k and we are interested in estimating future values of the signal $y[k+1], y[k+2], \cdots$, given the dataset $\{\mathbf{Z}[0],\mathbf{Z}[1],\cdots,\mathbf{Z}[k]\}$. The predictions are denoted as:

$$\widehat{y}[k+1|k] \quad \text{One-step ahead predictor} \tag{2-10}$$

$$\widehat{y}[k+p|k] \quad p\text{-step ahead predictor} \tag{2-11}$$

The quality of the results can be also assessed by computing the relative error ϵ (2-13) of the simulation using the estimated parameters. The optimal result is found when the relative error is equal to zero. Any improvement of new solutions can be checked by calculating the improvement percentage $\mathbf{\Delta}(\%)$ (2-14). Positive values indicate an improvement of the new model. Additional metrics are the standard deviation s, and the root mean squared error (RMSE). The optimal result is found when the standard deviation and the root mean squared error are equal to zero and the coefficient of determination is equal to one.

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - \widehat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \overline{y})^2} \tag{2-12}$$

$$\epsilon(\%) = 100 \sum_{i=1}^{N} \frac{\mid y_i - \widehat{y}_i \mid}{\mid y_i \mid} \tag{2-13}$$

$$\mathbf{\Delta}(\%) = 100 \left(1 - \left(\frac{\widehat{\epsilon}}{\epsilon}\right)\right) \tag{2-14}$$

$$s = \sqrt{\frac{\sum_{i=1}^{N}(y_i - \overline{y})^2}{N-1}} \tag{2-15}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(\widehat{y}_i - y_i)^2}{N}} \tag{2-16}$$

where y_i is the measured data and $\widehat{y_i}$ is that predicted, $\widehat{\epsilon}$ is the relative error of the proposed new model, $\overline{y}$ is the mean of the measurements and N is the number of samples or measurements.

There are two specific evaluation metrics to monitor the optimization process performed by metaheuristic algorithms. The first one is the minimum value of the cost function $f_{min}(G)$ (2-17) in every generation G. Its choice is justified since the aim of the iterative generated set of solutions is simply to find better solutions and not generating an entire population of higher quality solutions. The second is the mean population searching distance $dist_{mean}(G)$ (2-18), which can provide a direct measure of an EA convergence behavior. On the search process, high values of $dist_{mean}(G)$ mean high population diversity, while a series of decreasing of its values occurs when the algorithm converges [116].

$$f_{min}(G) = min\{f(\mathbf{S})\} \tag{2-17}$$

$$dist_{mean}(G) = \frac{2}{m(m-1)} \sum_{k=1}^{m} \sum_{l=k+1}^{m} dist(S_k(t), S_l(t)) \tag{2-18}$$

where $2/m(m-1)$ indicates the number of unique pairs that exist in a population of size m and $\mathbf{S}$ is the set of all candidates in generation G.

2.1.6 Combination of Models

The idea of ensemble methodology is to build a predictive model by combining multiple models. It has been considered by researchers from various disciplines such as statistics and Artificial Intelligence (AI) since this approach aims to improve prediction performance [117]. An ensemble approach is composed of a set of combined models that act together to predict a response variable [118]. Over the past two decades, this approach has received increasing attention from the computational intelligence and machine learning community because ensemble systems have proven to be highly effective and versatile in many real-world problem domains and applications [98]. Figure 2.8 illustrates a classification problem, its result is obtained from two base classifiers and a single combiner. The combiner has the role of weighing the individual classifiers and combining them to reach a final decision.

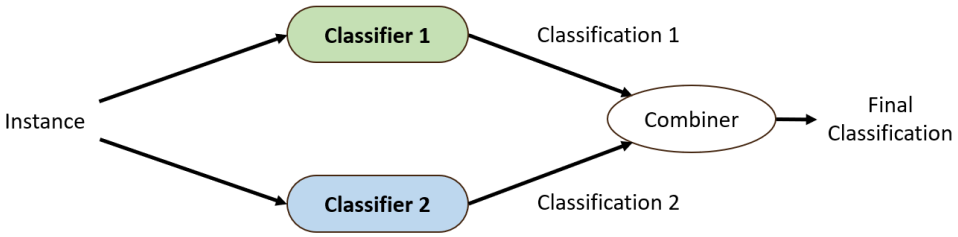

Figure 2.8: Basic classification problem where a combination method uses two classifiers to output the final classification. Adapted of [117].

3
Optimization Algorithms

Optimization is a systematic procedure of seeking the best choices [119] to accomplish either the least or the most extreme yield for any recognized issue [120]. In engineering, it is the process of maximizing or minimizing a desired objective function [121], which is the numerical measure of how good the decisions chosen are [122].

Optimization is applicable to solve many engineering problems, some applications in engineering are (*i*) design of aircraft and aerospace structures for minimum weight; (*ii*) design of pumps, turbines, and heat transfer equipment for maximum efficiency, (*iii*) optimum design of electrical networks and (*iv*) optimum design of control systems. This indicates how optimization is widely used [123].

The classification of optimization can be carried out in terms of the number of objectives, the number of constraints, function forms, the landscape of the objective functions, type of design variables, uncertainty in values, and computational effort. There are also different types of optimization problems that are solved by different optimization techniques since some methods are more suitable for certain types of optimization problems than others [124]. Figures 3.1 and 3.2 show, respectively, how optimization problems can be classified as well as optimization algorithms.

3.1
Classical optimization

Classical optimization (CO), also known as mathematical programming, comprises both analytical and numerical methods, which are useful in finding the optimum solution of continuous and differentiable functions, i.e., classical methods of optimization make use of the techniques of differential calculus in locating the optimum points. Because of this, the scope of use of CO is limited in practical applications since some of the practical problems involve objective functions that are not continuous and/or differentiable [123, 125]. This section introduces some of the basic concepts of optimization.

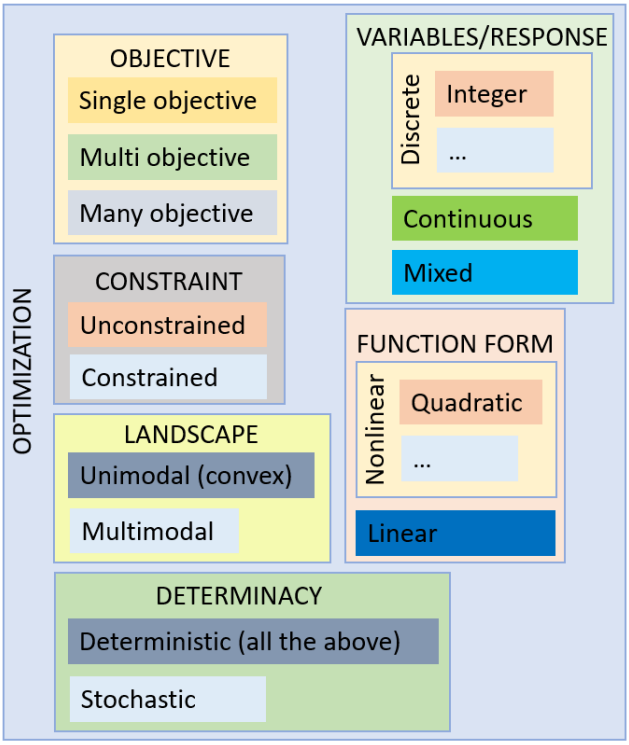

Figure 3.1: Classification of optimization problems. Adapted of [124].

3.1.1 Statement of an Optimization Application

Optimization is an iterative progression, a structured search through the space formed by the decision variables from an initial trial solution toward the optimum (commonly referred to as $\mathbf{X}^*$) [119]. A general optimization problem can be stated as follows:

$$\text{Find} \quad \mathbf{X} = \begin{Bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{Bmatrix} \text{which minimizes } f(\mathbf{X}) \tag{3-1}$$

subject to the constrains:

$$g_i(\mathbf{X}) \leq 0, i = 1, 2, ..., m \tag{3-2}$$

$$l_j(\mathbf{X}) = 0, j = 1, 2, ..., p \tag{3-3}$$

The optimization problem was stated being constituted by an objective function $f(\mathbf{X})$, by a n-dimensional design vector $\mathbf{X}$ (it contains n decision

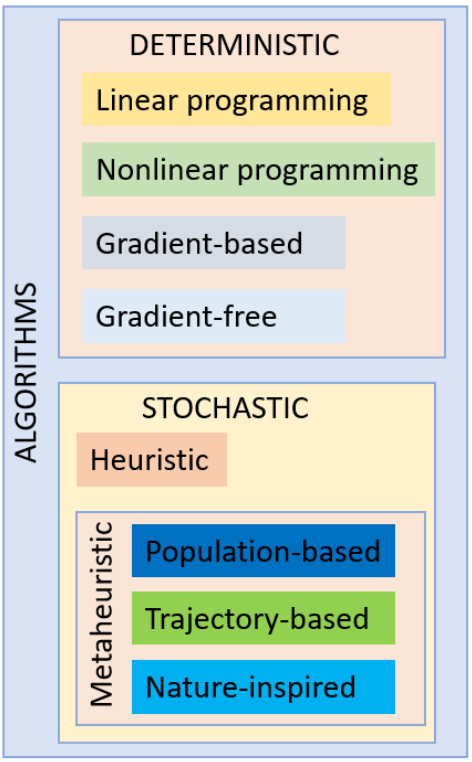

Figure 3.2: Classification of algorithms. Adapted of [124].

variables) and by $g_i(\mathbf{X})$ and $l_j(\mathbf{X})$, which are known as inequality and equality constraints, respectively. The above-stated problem is called a constrained optimization problem. However, some optimization problems do not involve any constraints, in this case, they are called unconstrained optimization problems.

3.1.2 The Notion of Local and Global Optima

Let us consider a single-variable optimization problem. A function of one variable $f(x)$ is said to have a relative or local minimum at $x = x^*$ if $f(x^*) \leq f(x^* + \delta)$ for all sufficiently small positive and negative values of δ. If $f(x^*) \geq f(x^* + \delta)$ for all values of δ sufficiently close to zero, a point x^* is denoted as a relative or local maximum. A function $f(x)$ is said to have a global or absolute minimum at x^* if $f(x^*) \leq f(x)$ for all x, and not just for all x close to x^*, in the domain over which $f(x)$ is defined. Similarly, a point x^* will be a global maximum of $f(x)$ if $f(x^*) \geq f(x)$ for all x in the domain. Figure 3.3 shows the difference between local and global optimum points for a single-variable optimization problem, in this case, the value of $x = x^*$ is to be found in the interval $[a, b]$ such that x^* minimizes or maximizes $f(x)$. An optimization problem can be written as a minimization problem or as a

maximization problem since min *f(x)* corresponds to max *-f(x)*, this is shown in Figure 3.4.

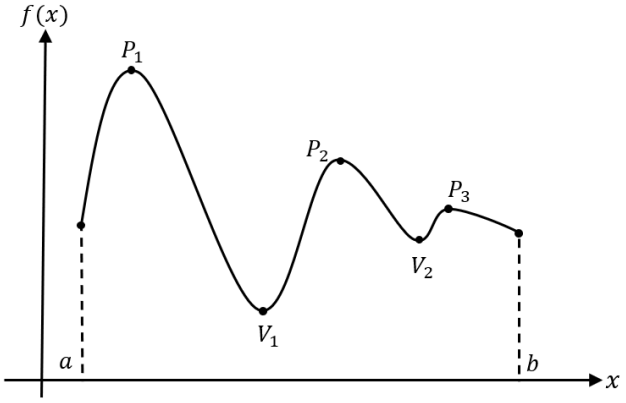

Figure 3.3: Relative and global minima and maxima of a single-variable function. P_1, P_2, and P_3 are relative maxima and P_1 is the global maximum. Similarly, V_1 and V_2 are relative minima and V_1 is the global minimum.

There is a variety of possible classification of optimization problems. One case is when there are no constraints involved, i.e., in Equations (3-2) and (3-3) i=0 and j=0, respectively. Then, Equation (3-1) is reduced to Equation (3-4).

$$\text{Find} \quad \mathbf{X} = \begin{Bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{Bmatrix} \text{which minimizes } f(\mathbf{X}) \tag{3-4}$$

Iterative algorithms have been developed for the solution of such problems. Beginning at a start point $\mathbf{X}_0$ the algorithm gradually converges towards a local optimum by generating a sequence of iterations $\mathbf{X}_j$, each new iteration can be described by Equation (3-5):

$$\mathbf{X}_{j+1} = \mathbf{X}_j + \eta_j \mathbf{d}_j \tag{3-5}$$

where $\mathbf{d}_j$ is the search direction at the point $\mathbf{X}_j$, and η_j is the step length.

The search direction is chosen according to each algorithm and the step length is typically adaptive, depending on $\mathbf{X}$. We may cite Newton's method and Gradient Descent method, often referred to as Steepest Descent, as options for the task of solving Equation (3-4). In the latter method, gradient information is used to select a suitable search direction, more specifically, the negative gradient at $\mathbf{X}_0$ is used since it gives the direction in which $f(\mathbf{X})$ decreases most [125]. Thus, the search direction can be written as:

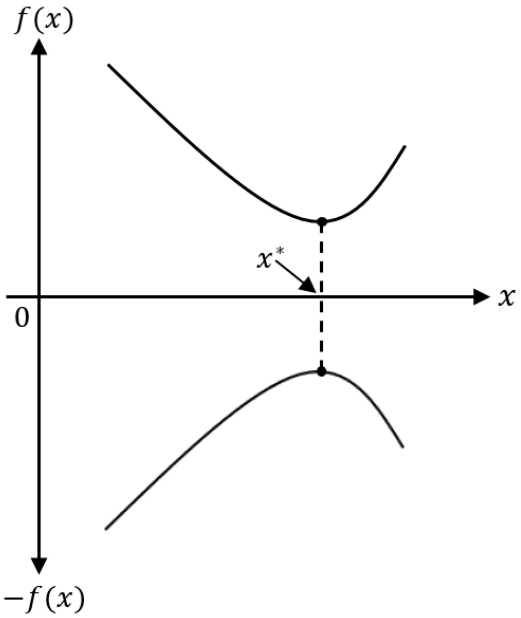

Figure 3.4: Maximization vs. Minimization. Minimization of $f(x)$ is equivalent to maximization of $-f(x)$.

$$\mathbf{d}_0 = -\nabla f(\mathbf{X}_0) \tag{3-6}$$

Equation (3-5) can be rewritten changing $\mathbf{d}_j$ for $\mathbf{d}_0$, so that the iteration now takes the form of:

$$\mathbf{X}_{j+1} = \mathbf{X}_j - \eta_j \nabla f(\mathbf{X}_0) \tag{3-7}$$

3.2 Metaheuristic Algorithms

This section briefly discusses metaheuristic algorithms, which are powerful and popular methods for solving complex engineering optimization problems [123]. Most metaheuristic algorithms are nature-inspired since they have been developed based on some abstraction of nature. We focus on swarm intelligence, population-based algorithms, and those that rely on evolutionary principles. They are often referred to as evolutionary algorithms (EAs), but there are a variety of possible terms to refer to, such as evolutionary algorithms, population-based algorithms, computer intelligence, and nature-inspired computing [126]. Although there are many terminologies, these algorithms evolve a population of promising solutions by using a mechanism that comprises two operators: selection and variation [127].

Figure 3.5 shows a general framework of an EA. The basic idea of EAs is that a population of individuals is exposed to environmental pressure, by this a natural selection process occurs, leading to the survival of the fittest [128], which increases the population average aptitude to survive and reproduce in a particular environment, i.e., its fitness [129]. The main evolutionary mechanisms used by an evolutionary optimization problem are reproduction, mutation, recombination, crossover, selection, and survival of the fittest. Each individual of the population represents a candidate solution, and the cost or objective function is analogous to the environment, deciding the fitness of each individual of the population. The next few subsections briefly discuss some EAs employed in this work.

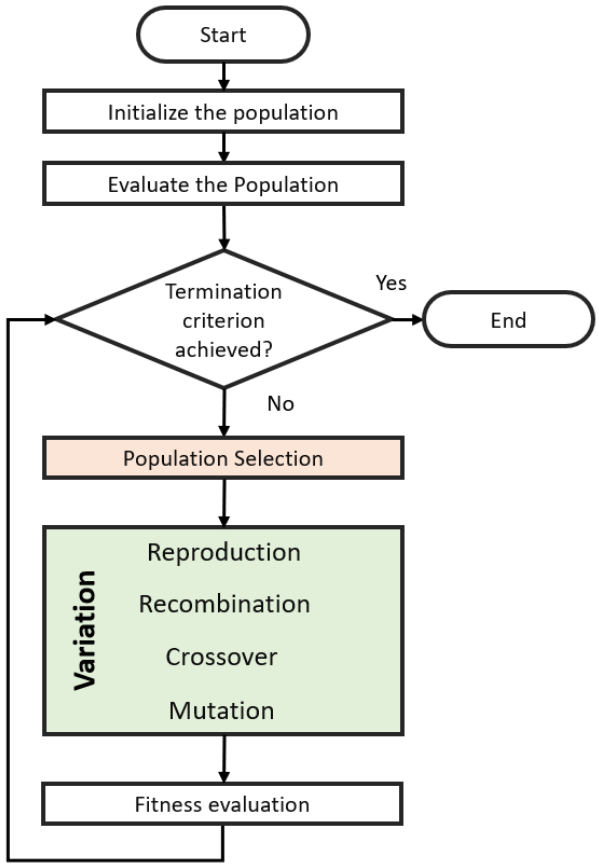

Figure 3.5: General framework of Evolutionary Algorithms. The main mechanisms used in this process are population selection, reproduction, recombination, crossover, and mutation.

3.2.1 Genetic Algorithm

Genetic algorithm (GA) [130] was one of the first evolutionary algorithms. Nowadays GAs remain popular, having a variety of implementations, this

is due to their performance on a range of problems and that they are also easily implementable. GAs are optimization algorithms that use principles of the mechanics of natural selection and natural genetics [108]. Features of an individual are inherited by its offspring and they will be kept according to their fitness in the environment. Adaptation plays also an important role in the optimization process since individuals are progressively eliminated if they are unable to adapt to the environment. Figure 3.6 shows a flowchart of a standard GA, the first step is the initialization of the initial set of individuals. In GA, individuals are represented as chromosomes, which consist of a set of genes, and each gene is a decision variable. The solutions are evaluated according to their fitness, which is the chromosome's fitness to the environment. The standard GA is composed of three operators, namely selection, crossover, and mutation [131].

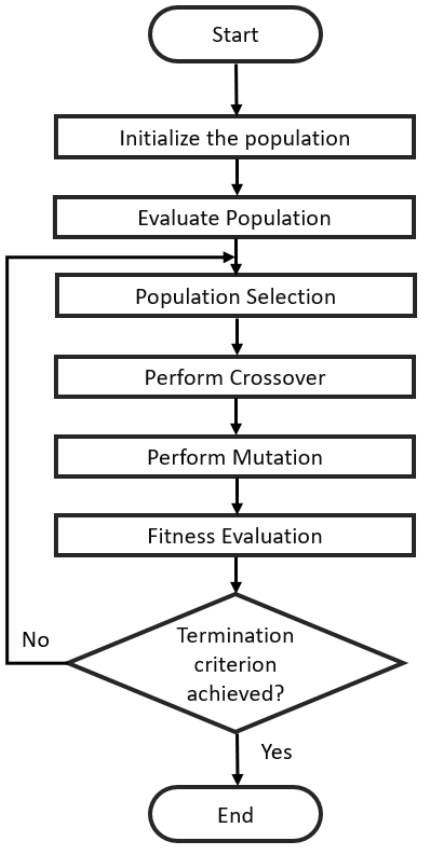

Figure 3.6: General framework of a GA.

In its simplest form, GA involves two operators: selection and single-point crossover. A third operator was later introduced: mutation. First, a set of initial candidates must be initialized. The next step is performed by first evaluating the initial population to select the parents for the next generation. Parents generate offspring through genetic recombination (crossover). A mutation

process can occur in the new generation, it improves the search capabilities of GAs. The fitness of the new population is finally evaluated, this process endures until the termination criterion is achieved.

Selection is performed to choose individuals that will share their genetic information to produce children for the next generation. We may cite some selection methods, such as roulette-wheel (also called fitness proportional selection), rank-based, and tournament selection [132]. The former method is the most popular, in this approach a roulette wheel with slot areas corresponding to the fitness of the individuals are created. This way, an individual i in the population of N individuals will be selected for the crossover with a probability p_i, which is calculated by Equation (3-8). The selection is performed by randomly rotating the wheel to get a sector.

$$p_i = \frac{f_i}{\sum_{j=1}^{N} f_j} \tag{3-8}$$

After selecting two parents, two children will be generated, each one receiving some genetic information from one parent. The parents die and their children give rise to a new generation to continue the evolutionary process. Low-fitness individuals are more likely of dying in their generation, which means they will be removed from the GA simulation. By this, high-fitness individuals can survive to cross over and give rise to a new generation of individuals. Crossover can be performed in a generation j as shown by Figure 3.7(a), which describes the single-point crossover. In this case, a position is randomly selected and two parents share their genes to reproduce two children. This process might be also executed as shown in Figure 3.7(b), which is known as multi-point crossover; this is an extension of the former process. It is important to mention that a uniform crossover process can be also executed, in this case, each gene can be probabilistic choose for crossover.

The last step of the GA algorithm is a mutation, which is a random change on one or more genes of some chromosomes, it is performed so that the solution diversity can be increased. A mutation is important because it helps the evolutionary process by adding some genetic information into the population in case any is missing [126]. Algorithm 1 describes a standard GA after a brief introduction to its three operators.

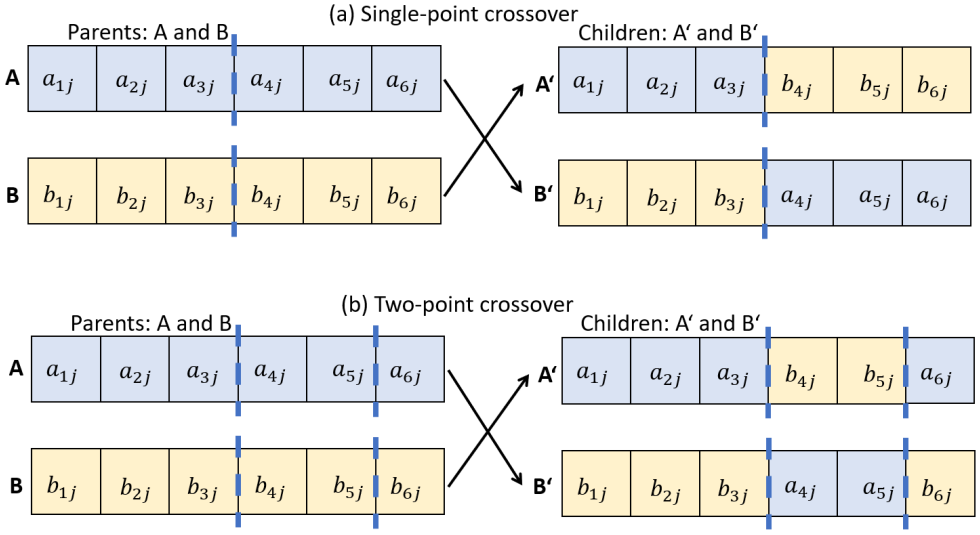

Figure 3.7: Single and two-point crossover representation. The parents can be selected by using methods such as roulette-wheel, rank-based, and tournament selection. In this process, each individual shares some of his genetic information with his offspring. The crossover point is chosen randomly and the parents mate to generate individuals for the new generation.

Algorithm 1 Genetic Algorithm

```
Initialize a population of N Parents
while termination criterion is not satisfied do
    Evaluate the fitness of each parent in the population
    Empty Offspring
    while |Offspring|< |Parents| do
        Use fitness to probabilistic select two parents P1 and P2 from Parents
        With a probability of pc, conduct crossover on P1 and P2
        Add P1 and P2 to Offspring
        With a probability of pm, conduct mutation on P1 and P2
    end while
    Parents ← Offspring
end while
Next generation
```

3.2.2 Ant Colony Optimization

The first ant colony optimization algorithm (ACO) was the ant system (AS) [133], it is a population-based optimization algorithm motivated by the pheromone-deposition behavior of ants. The basic idea underlying this algorithm is the positive feedback mechanism [134], which is based on the trail-laying trail-following behavior of some species of ants [135]. When ants

travel searching for a food source, they leave a trail of pheromone along the path. If they are successful on their journey, they will deposit more pheromone along the path. Other ants will smell the pheromone and bring more food to the colony further enhancing the pheromone density on the path. If the ants do not find a food source they will not deposit more pheromone when coming back to the colony. Other ants will not follow this path since pheromone trails evaporate with time, reducing their density indicating a path that will not lead them to a food source. Figure 3.8 shows a simple experimental setting that demonstrates how ants find the shortest path between their colonies and the source of food.

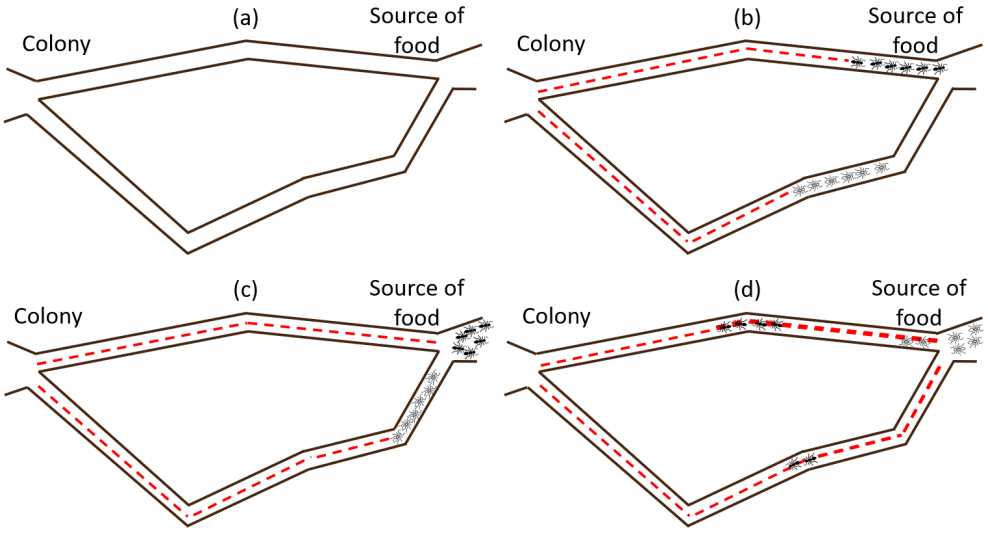

Figure 3.8: How ants can find the shortest path between food sources and their colony. Red dashed lines represent the pheromone deposition, and their thickness is the density of deposition. (a) There is no pheromone in the environment and the ants are not searching for food. (b) As ants go outside to search for any source of food, they choose with a probability of 50% the short or the long way and they deposit pheromone along the path. (c) One group of ants have arrived first since they took the short path, and when returning its members are more likely to choose the short path again. (d) The probability of taking the short path increases since the pheromone trail on it is more reinforced. As time goes, the colony will probably use the short path because of the pheromone evaporation on the long path.

The first ACO algorithm was applied to solve different combinatorial optimization problems as the Traveling Salesman Problem (TSP), this procedure is shown by Algorithm 2. The specific parameters of the ACO algorithm are the evaporation rate (ρ), deposition constant (Q), initial pheromone between cities (τ_0), pheromone between cities (τ_{ij}), and distance between cities (d_{ij}).

Algorithm 2 ACO Algorithm

Place N ants on n cities
while termination criterion is not satisfied **do**
  for k= 1 to N **do**
    for $i = 1$ to n-1 **do**
      Let i be the current city where ant k is placed
      Let C_j be the group of adjacent cities of i
      for each city j in C_j **do**

$$p_{ij}(k) = \frac{(\tau_{ij})^{\alpha}/(d_{ij})^{\beta}}{\sum_{l \in C_j} (\tau_{il})^{\alpha}/(d_{il})^{\beta}} \tag{3-9}$$

      end for
    end for
    let ant k go to city j with probability $p_{ij}(k)$
  end for
  $L_k \leftarrow$ Total path covered by ant k
  for each city i **do**
    for each city j **do**
      for each ant k **do**
        if ant k visited city j from city i **then**
          $\Delta\tau_{ij}(k) \leftarrow Q/L_k$
        else
          $\Delta\tau_{ij}(k) \leftarrow 0$
        end if
      end for
      $\tau_{ij} \leftarrow (1-\rho)\tau_{ij} + \Delta\tau_{ij}(k)$
    end for
  end for
end while
Next generation

3.2.3 Biogeography-Based Optimization

Inspired by the study of the speciation, extinction, and geographical distribution of biological species, Simon [136] proposed a new metaheuristic algorithm called biogeography-based optimization (BBO). BBO is a population-based evolutionary algorithm in which each solution is analogous to an island (habitat) with an immigration rate and an emigration rate. Islands are evaluated according to their conditions for species to live, if the conditions are suitable these islands have a high habitat suitability index (HSI). Rainfall, vegetative diversity, land area, and temperature are some of the variables associated with habitability, they are called suitability index variables (SIVs). The island with a high HSI value has a low immigration rate because it tends to support many species. On the other hand, the immigration rate is high for

islands with low HSI values since they contain small populations [68].

Figure 3.9 represents a simple linear species migration model of an island [137], although it might be more complicated in general. Immigration λ and emigration µ rates are dependent on the number of species in the habitat. When there are zero species on the island the maximal possible immigration rate to the habit is I, which decreases with the increase of the number of species, as the island becomes more crowded because fewer species can survive immigration successfully. The immigration rate is zero as the maximal possible number of species that the habitat can support (S_{max}) is achieved. Considering now the emigration curve, the emigration rate is zero in the case there are no species on the island. As the number of species increases the habitat becomes more crowded so that more species leave the habitat, then the emigration rate increases achieving its maximum value E. S_0 is known as the equilibrium number of species and M_0 is its correspondent rate of equilibrium, i.e., the intersection point of λ and μ.

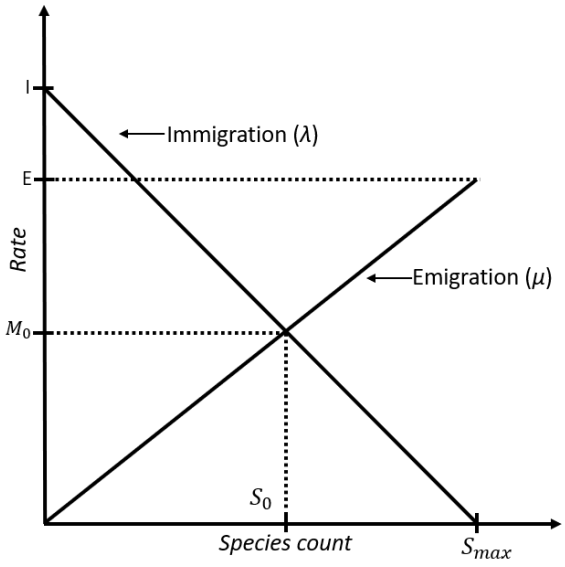

Figure 3.9: Simple linear species migration model of an island. The maximal possible immigration and emigration rates are, respectively, I and E. S_0 is the equilibrium species count and M_0 is its correspondent rate of equilibrium ($\mu = \lambda$). A linear migration curve is assumed for convenience but in biogeography migration curves assume a nonlinear behavior. Another representation possible could be an S-shaped curve for migration rate.

In a BBO problem, a candidate from the set of solutions is considered as a habitat, its HSI is analog to the fitness value in other population-based optimization algorithms and each candidate component is an SIV. Immigration and emigration rates are used to perform migration in each generation so that the population can continually evolve [126]. Mutation can be also introduced

as an operator in the optimization problem. P_m is the mutation probability and l_b and u_b are, respectively, the lower and upper search bounds of the ith SIV. Based on the migration and mutation operators, a framework of the basic BBO can be given by Algorithm 3, which is known as partial immigration-based BBO [138], since it is probabilistic to decide whether to replace a solution feature.

Algorithm 3 The basic framework of the BBO algorithm

```
Initialize a set of N candidate solutions called habitats: X
while termination criterion is not satisfied do
    for each habitat x_k, k ∈ [1, N], x_k ∈ X do
        set emigration rate μ_k proportional to HSI: μ_k ∈ [0,1]
        set immigration rate as λ_k = 1 − μ_k
    end for
    Set a temporary population of habitats: Z ← X
    if rand(0,1) < λ_k then
        for each habitat x_k, k ∈ [1, N], x_k ∈ X do
            for each SIV do
                perform migration on x_k using μ_j to probabilistic select the
emigrating habitat x_j:
                x_k(SIV) ← x_j (SIV)
            end for
        end for
    end if
    for each SIV in habitat x_i do
        if rand(0, 1) < P_m then
            x_i(SIV) ← rand(l_b, u_b)
        end if
    end for
    X ← Z
end while
Next generation
```


3.2.4 Differential Evolution

Differential Evolution (DE) [139] is not a biologically-motivated, but a population-based algorithm [140], which searches for a global optimum point in an n-dimensional continuous domain [141]. Each solution $\mathbf{x_i}$ is an n-dimensional vector, it is represented by (3-10), which can be considered as the chromosomes of a GA [142].

$$\mathbf{x_i} = \{x_{1,i}, x_{2,i}, ..., x_{n,i}\} \tag{3-10}$$

DE uses a scaled version of the difference between two distinct individuals and adds it to a third individual to obtain a new candidate solution. This

process can be resumed in three stages: mutation, crossover, and selection. Figure 3.10 describes some steps of the basic idea behind differential evolution in a two-dimensional search space.

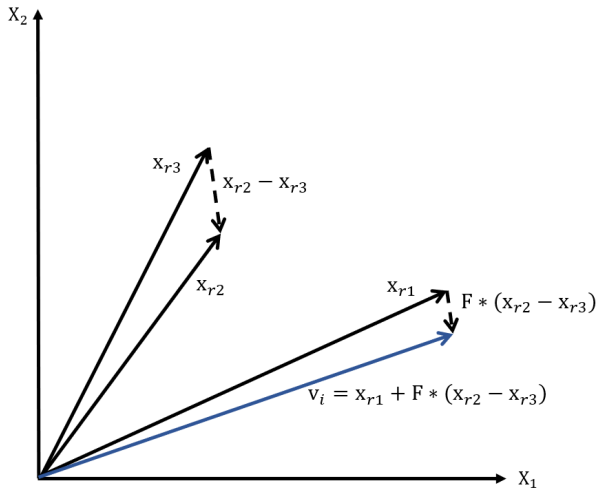

Figure 3.10: The basic idea of differential evolution mutation scheme for a two-dimensional search space. Three distinct randomly chosen vectors ($\mathbf{x_{r1}}$, $\mathbf{x_{r2}}$, and $\mathbf{x_{r3}}$) are used to create a mutant vector $\mathbf{v_i}$.

First, two distinct individuals $\mathbf{x_{r2}}$ and $\mathbf{x_{r3}}$ are randomly chosen to obtain a mutant vector $\mathbf{v_i}$, which is built by adding to $\mathbf{x_{r1}}$ a scaled version of the difference between $\mathbf{x_{r2}}$ and $\mathbf{x_{r3}}$. A crossover process is performed so that a trial vector $\mathbf{u_i}$ is generated, the process is finally finished by selecting the fittest individual among $\mathbf{x_i}$ and $\mathbf{u_i}$. Algorithm 4 summarizes the basic DE algorithm for an n-dimensional problem.

Algorithm 4 DE Algorithm

Set the step size parameter (F)
Set the crossover rate (c)
Initialize a population of N candidate solutions: $\mathbf{X}$
while termination criterion is not satisfied **do**
  for each$\mathbf{x_i} \in \mathbf{X}$ **do**
    randomly chose three distinct individuals $\mathbf{x_{r1}}$, $\mathbf{x_{r2}}$ and $\mathbf{x_{r3}}$ from $\mathbf{X}$: $\mathbf{x_i} \notin \{\mathbf{x_{r1}}, \mathbf{x_{r2}}, \mathbf{x_{r3}}\}$
    generate a mutant vector:
    $\mathbf{v_i} \leftarrow \mathbf{x_{r1}} + \mathrm{F} * (\mathbf{x_{r2}} - \mathbf{x_{r3}})$
    Perform crossover:
    $J_r \leftarrow$ rand(1,n)
    for each dimension j $\in [1, n]$ **do**
      $r_{cj} \leftarrow$ rand(0,1)
      if $r_{cj} <$ c or $j = J_r$ **then**
        $u_{ij} \leftarrow v_{ij}$;

```
            else
                u_ij ← x_ij;
            end if
        end for
    end for
    for each population index i ∈ [1, N] do
        select the fitter one:
        if f(u_i) < f(x_i) then
            x_i ← u_i
        end if
    end for
end while
Next generation
```

3.2.5 Particle Swarm Optimization

Particle Swarm Optimization (PSO) [143] was inspired by the social behavior of bird flocks, it belongs to the group denominated as Swarm Intelligence algorithms, which are inspired by the collective behavior of social insects as well as from other animal societies for the design of intelligent multi-agent systems [144, 145]. Candidate solutions are analogous to particles of a swarm, they move through the problem search space with some velocity $\mathbf{v_i}$ searching for an optimal solution. PSO was first designed for n-dimensional minimization problems defined over a continuous domain. Although PSO is a population-based algorithm it does not make use of crossover and mutation, instead of that the position and velocity of each individual are adjusted in every generation [68], which means that PSO can model the dynamics of the movement of the particles through the search space [126]. The position $\mathbf{x_i}$ of each particle is adjusted generation by generation according to its velocity and taking into account its own best position so far as well as from its neighbors [144]. It means that the social interaction between the particles is used to perform the position adjustment of the particles since the experience of the neighboring particles in the search space influences their movement. The velocity and position of each particle component can be modeled, respectively, by Equations (3-11) and (3-12).

$$v_{ij}^{t+1} = w v_{ij}^{t} + \phi_1(b_{ij}^{t} - x_{ij}^{t}) + \phi_2(h_{ij}^{t} - x_{ij}^{t}) \tag{3-11}$$

$$x_{ij}^{t+1} = x_{ij}^{t} + v_{ij}^{t+1} \tag{3-12}$$

where w is a scalar parameter called inertia weight. It can be a fixed value $w \in (0,1)$ or adjustable over the interactions. If t_{max} is the maximum number of interactions, w_{min} and w_{max} are defined as the minimum and maximum

values of w, the inertia weight term can be modeled as in Equation (3-13). ϕ_1 and ϕ_2 are learning rates or acceleration constants. The difference terms $(b_{ij}^t - x_{ij}^t)$ and $(h_{ij}^t - x_{ij}^t)$ are the cognitive and social terms, respectively. It is important to keep in mind that an arbitrary growth of the velocity shall be avoided [132]. Finally, the pseudocode for PSO is presented by Algorithm 5.

$$w = w_{max} - \frac{t}{t_{max}}(w_{max} - w_{min}) \tag{3-13}$$

Algorithm 5 PSO Algorithm

Initialize a swarm $\mathbf{X}$ of N particles
Initialize the velocity vector $\mathbf{v_i}$ of each particle
Initialize the best position of each individual $\mathbf{b_i}$
Define the neighborhood size
Set the maximum learning rates $\phi_{1,max}$ and $\phi_{2,max}$
Define the maximum velocity v_{max}
Define the lower and upper borders of the inertia weight w: w_{min}, w_{max}
while termination criterion is not satisfied **do**
    for each individual $\mathbf{x_i}$ of the swarm **do**
        Set $\mathbf{H_i}$ as the set of the P nearest neighbors of x_i
        From $\mathbf{H_i}$ select the fittest neighbor $\mathbf{h_i}$ of $\mathbf{x_i}$
        $\mathbf{h_i} \leftarrow \arg\min_x\{f(x) : x \in \mathbf{H_i}\}$
        Generate the learning rate ϕ_1 for each dimension j of $\mathbf{x_i}$: ϕ_1 [0,$\phi_{1,max}$]
        Generate the learning rate ϕ_2 for each dimension j of $\mathbf{x_i}$: ϕ_2 [0,$\phi_{2,max}$]
        Update the velocity of $\mathbf{x_i}$ according to Equation (3-11)
        if $|\mathbf{v_i}| > v_{max}$ **then**
            $\mathbf{v_i} \leftarrow \mathbf{v_i} v_{max} / \mid \mathbf{v_i} \mid$
        end if
        Update the position of $\mathbf{x_i}$ according to Equation (3-12)
        bi $\leftarrow \arg\min\{f(\mathbf{x_i}), f(\mathbf{b_i})\}$
    end for
    Update the inertia weight w according to Equation (3-13)
end while
Next generation

3.3 Surrogate Models Used for Optimization

Surrogate models, also known as metamodels or reduced-order models, are an approximation of the input-output that seek to understand input and output relationships that are either unknown or complex via reasonably accurate simpler functions [146, 147].

3.3.1 Bayesian Optimization Using Gaussian Process

Bayes' theorem plays a central role in statistical analysis, it describes the probability of an event based on prior knowledge of the conditions that might be related to the event [92, 148]. Suppose that $\mathbf{y} = (y_1, ..., y_n)$ is a vector of n observations with a probability distribution $p(y \mid \theta)$ depending on the values of k parameters $\theta = (\theta_1, ..., \theta_k)$. θ also has $p(\theta)$ probability distribution [149]. The conditional distribution of θ, given the observed data $\mathbf{y}$, is expressed as follows:

$$p(\theta \mid \mathbf{y}) = \frac{p(y \mid \theta)p(\theta)}{p(\mathbf{y})} \tag{3-14}$$

where $p(\theta)$ is called the *prior* distribution of θ, which tells what is known about θ without knowledge of the data. $p(\theta \mid \mathbf{y})$ is called the *posterior* distribution of θ given $\mathbf{y}$, which tell us what is known about θ given the knowledge of the data.

Bayesian Optimization (BO) was first introduced by [150] and then by [151] and [152]. It is composed of two main components: a Bayesian statistical model for modeling the objective function and an acquisition function for deciding where to sample next. BO usually models the objective function by a Gaussian process (GP), and iteratively samples the next data point by maximizing an acquisition function. BO is a popular approach for expensive black-box optimization problems, some applications are parameter tuning, experimental design, and robotics [153, 154]. A simple framework of a BO problem using Gaussian Process with N function evaluations is shown by Algorithm 6.

An objective function $f(x_n)$ is sequentially optimized with assumptions on a prior distribution, which is a probabilistic model, over $f(x_n)$. In each iteration a point x is chosen by maximizing an acquisition function ($acq(\cdot)$), the objective value $f(x_n)$ is evaluated, and the prior distribution is updated with the new data point.

The task of an acquisition function is to measure the value that would be generated by the evaluation of the objective function at a new point x, based on the current posterior distribution over $f(x_n)$. Some examples of acquisition functions are the probability of improvement (PI) [150], the expected improvement (EI) [155], and the upper confidence bound (UCB) [156].

Algorithm 6 Basic pseudo-code of a Bayesian optimization problem

1: Place a Gaussian process on f
2: Observe f at n_0 points according to an initial space-filling experimental design.
3: Set $n = n_0$.
4: **while** $n \leq N$ **do**
5: Update the posterior probability distribution on f using all available data
6: Let x_{n+1} be a maximizer of the acquisition function over x, where the acquisition function is computed using the current posterior distribution.

$$\mathbf{x}_{n+1} = \arg\max_{\mathbf{x}\in X} \; acq(\mathbf{x}) \tag{3-15}$$

7: Evaluate f at $\mathbf{x}_{n+1}$ to obtain y_{n+1}
8: Increment n
9: **end while**
10: Return a solution: either the point evaluated with the largest $f(x_n)$, or the point with the largest posterior mean.

3.3.2 Sequential Optimization Using Decision Trees

In sequential optimization using decision trees (DT) a tree-based regression model is used to sequentially evaluate the expensive function at the next point. Thereby finding the minimum of the function with as few evaluations as possible.

In supervised learning problems, one tries to use a set of inputs, or independent variables, to predict an output or dependent variable. In classification problems, the output is a categorical variable, on the other hand, if the output is a continuous variable this case is referred to as a regression problem. Tree-based models have a set of useful tools for supervised learning tasks. They approach these problems by recursively partitioning a learning sample over its input variable space and fitting a simple function to each resulting subgroup of cases [157]. Recursive partitioning (RP) is a non-parametric technique for prediction and classification. RP provides in its standard way, trees that display the succession of rules that need to be followed to derive a predicted value or class [158].

RP is the key to the non-parametric statistical method of classification and regression trees (CART) [159]. A decision tree is constructed by either splitting or not splitting each node on the tree into two daughter nodes, it is done step-by-step by a recursive partitioning process [160]. Basic regression trees partition the data into smaller groups that are more homogeneous concerning the response. Outcome homogeneity is achieved in regression trees by determining [161]:

(i) The predictor to split on and value of the split;

(ii) The depth or complexity of the tree;

(iii) The prediction equation in the terminal nodes.

For regression problems, the model begins with the entire data set, S, and searches every distinct value of every predictor to find the predictor and split value that partitions the data into two groups (S_1 and S_2). This process is done in such a way that the overall sum of squared errors (SSE) is minimized:

$$SSE = \sum_{i \in S_1} (y_i - \bar{y}_1)^2 + \sum_{i \in S_2} (y_i - \bar{y}_2)^2 \tag{3-16}$$

where $\bar{y}_1$ and $\bar{y}_2$ are the averages of the training set outcomes within groups S_1 and S_2, respectively.

Suppose we have a simple example of a recursive partitioning involving two input variables, X_1 and X_2. Suppose the tree diagram is given in the left panel of Figure 3.11.

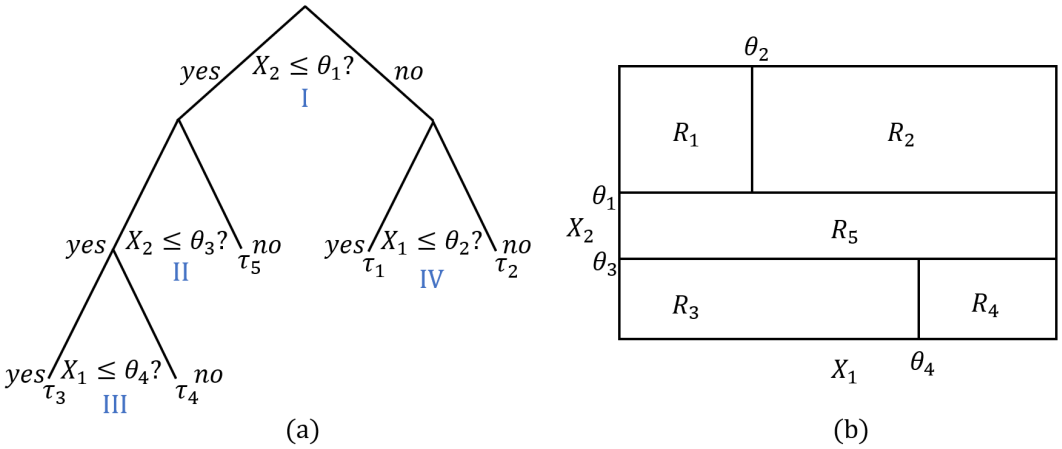

Figure 3.11: Example of recursive partitioning with two input variables X_1 and X_2. The left panel shows a decision tree with five terminal nodes, τ_1-τ_5, and four splits. The right panel shows the partitioning of two into five regions, R_1-R_5, corresponding to the five terminal nodes.

The possible stages of this tree are as follows: (I) Is $X_2 \le \theta_1$? If yes, follow the left branch; if no, follow the right branch. For a positive answer to (I) we ask the next question: Is $X_2 \le \theta_3$? An answer of no yields terminal node τ_5 with corresponding region $R_5 = \{\theta_1 \le X_2 < \theta_3\}$; For a positive answer to (II) we ask the next question (III): Is $X_1 \le \theta_4$? An answer of no yields terminal node τ_4 with corresponding region $R_4 = \{X_2 \le \theta_3, X_1 > \theta_4\}$; otherwise, the corresponding region is $R_3 = \{X_2 \le \theta_3, X_1 \le \theta_4\}$. A negative answer to (I) leads to questions IV: Is $X_1 \le \theta_2$? An answer of yes results in terminal node τ_1

with corresponding region $R_1 = \{X_2 > \theta_1, X_1 \leq \theta_2\}$; An answer of no yields terminal node τ_2 with corresponding region $R_2 = \{X_2 > \theta_1, X_1 > \theta_2\}$. For this example, $\theta_4 > \theta_2$ and $\theta_1 > \theta_3$.

4
Friction Models

Studies concerning friction have been carried out over the centuries. Da Vinci, Amontons [162], and Coulomb [163] have made the first contributions on this topic, the latter, however, has brought the most significant advances in this area developing the first mathematical friction model [164, 165]. In Coulomb's friction model a constant friction component is assumed, it is independent of the magnitude of the velocity, however, at zero velocity the friction force is a multi-valued function (4-1).

$$\mathbf{F} = \begin{cases} F_C sign(\mathbf{v}) & \text{if } \|\mathbf{v}\| \neq 0 \\ min(\|\mathbf{F_e}\|, \mathbf{F_C}) sign(\mathbf{F_e}) & \text{if } \|\mathbf{v}\| = 0 \end{cases} \tag{4-1}$$

$\mathbf{F_e}$ represents the resultant of the external forces acting on the reference body in the tangential direction of the contact, $\mathbf{v}$ is the relative tangential velocity of the body concerning the other contacting surface. F_C denotes the magnitude of Coulomb's friction (4-2); μ and $\mathbf{F_N}$ are, respectively, the coefficient of friction and the normal force of contact.

$$F_C = \mu \|\mathbf{F_N}\| \tag{4-2}$$

One of the most common modifications of Coulomb's friction law is the introduction of a viscous friction component F_v proportional to the velocity (4-3) corresponding to a well-lubricated situation. This combination is also the most often employed model as well [165, 166], it is given by Equation (4-4).

$$\mathbf{F} = F_v \mathbf{v} \tag{4-3}$$

$$\mathbf{F_C^s} = \begin{cases} F_C sign(\mathbf{v}) + F_v \mathbf{v} & \text{if } \|\mathbf{v}\| \neq 0 \\ min(\|\mathbf{F_e}\|, \mathbf{F_C}) sign(\mathbf{F_e}) & \text{if } \|\mathbf{v}\| = 0 \end{cases} \tag{4-4}$$

Equation (4-4) is the symmetric version of Coulomb's friction law with viscous friction, but in practice, friction may be asymmetric. It can be handled by using different values of the parameters for positive and negative values of the velocity [167]. This assumption leads us to rewrite equation (4-4) in its asymmetric version (4-5), here we call it $\mathbf{F_C^a}$.

$$\mathbf{F_C^a} = \begin{cases} F_C^+ sign(0^+(\mathbf{v})) + F_v^+ 0^+(\mathbf{v}) + F_C^+ sign(0^-(\mathbf{v})) + F_v^+ 0^-(\mathbf{v}) & \text{if } \|\mathbf{v}\| \neq 0 \\ min(\|\mathbf{F_e}\|, F_C^+ sign(0^+(\mathbf{v})), F_C^- sign(0^-(\mathbf{v})))sign(\mathbf{F_e}) & \text{if } \|\mathbf{v}\| = 0 \end{cases} \tag{4-5}$$

F_C^+, F_C^-, F_V^+, F_V^- represent, respectively, the magnitude of Coulomb's friction and the viscous friction coefficient for positive and negative velocities. Two mathematical operators denoted as $0^+(\mathbf{v})$ and $0^-(\mathbf{v})$, as given by Equation (4-6) and (4-7) respectively, are used to compute different values of the parameters for positive and negative values of the velocity. $0^+(\mathbf{v})$ returns $\mathbf{v}$ if $\mathbf{v} > 0$ and 0 otherwise and $0^-(\mathbf{v})$ returns -$\mathbf{v}$ if $\mathbf{v} < 0$ and 0 otherwise.

$$0^+(\mathbf{v}) = \mathbf{v}\,(1 + sign\,(\mathbf{v}))\,/2 \tag{4-6}$$

$$0^-(\mathbf{v}) = \mathbf{v}\,(1 - sign\,(\mathbf{v}))\,/2 \tag{4-7}$$

As shown by Figure 4.1, Coulomb's friction model has a discontinuity at zero velocity (blue line). To soften its discontinuity the model of Equation (4-8) was proposed by [168], its representation is shown by Figure 4.1 (orange line). In this present work, however, we will include the viscous friction component in Equation (4-8) and take into account the modification proposed by [165] since it has a higher resemblance with Coulomb friction law. Equation (4-9) shows the proposed model.

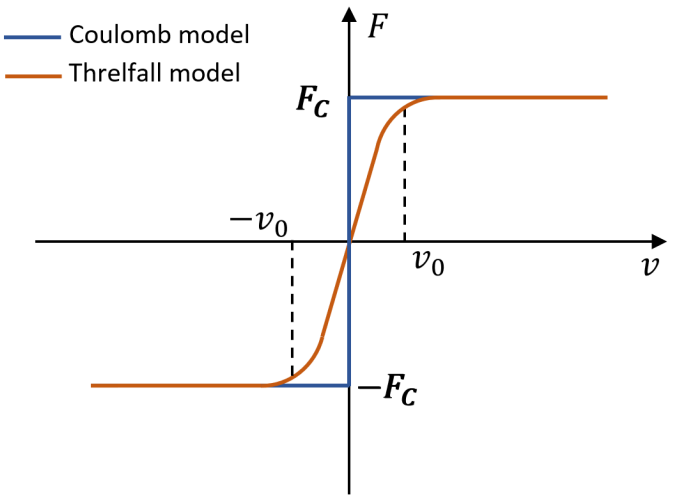

Figure 4.1: Comparison between the model proposed by [168] for a 1D case and Coulomb's friction model. The former model softens the discontinuity of the latter model for velocities close to zero through an exponential term.

$$\mathbf{F} = \begin{cases} F_C \left(1 - e^{\frac{3\|\mathbf{v}\|}{v_0}}\right) sign(\mathbf{v}) & \text{if} \quad \|\mathbf{v}\| \leq v_0 \\ 0.95 F_C sign(\mathbf{v}) & \text{if} \quad \|\mathbf{v}\| > v_0 \end{cases} \tag{4-8}$$

where v_0 is a tolerance velocity.

$$\mathbf{F_F^s} = \begin{cases} F_V\mathbf{v} + F_C\left(\frac{1-e^{\frac{3\|\mathbf{v}\|}{v_0}}}{1-e^3}\right) sign(\mathbf{v}) & \text{if} \quad \|\mathbf{v}\| \leq v_0 \\ F_V\mathbf{v} + F_C sign(\mathbf{v}) & \text{if} \quad \|\mathbf{v}\| > v_0 \end{cases} \tag{4-9}$$

Due to its importance and complexity, further friction models have been formulated. A model also including an exponential decaying term in the friction was proposed by Tustin [169], which is well suitable to describe friction force at a velocity close to zero [170]. This model is given by Equation (4-10).

$$\mathbf{F} = F_S - F_{SK}(1 - e^{-\frac{\mathbf{v}}{v_c}}) + F_v\mathbf{v} \tag{4-10}$$

where F_S is the static friction coefficient; F_{SK} is the difference between static friction and kinetic friction; and v_c is a characteristic velocity at which the system transits to kinetic friction.

It is common to find Equation (4-10) rewritten as (4-11) [170–172], in this case v_s can be the Stribeck velocity. Modified versions of this model were also proposed by [173] and [174].

$$\mathbf{F_T^s} = F_C sign(\mathbf{v}) + (F_S - F_C)e^{-\frac{\mathbf{v}}{v_s}} + F_v\mathbf{v} \tag{4-11}$$

Another exponential model that considers Coulomb's model with viscous friction, stiction, and the Stribeck effect, which manifests as a nonlinear dependence of friction on sliding speed, was introduced by [175]. This model is given by Equation (4-12).

$$\mathbf{F_V^s} = F_v\mathbf{v} + \left(F_C + (F_S - F_C)e^{-\left(\frac{\|\mathbf{v}\|}{v_s}\right)^{\delta_\sigma}}\right) sign(\mathbf{v}) \tag{4-12}$$

where δ_σ is a factor that relies on the geometry of the contacting surfaces.

5
Case Studies - EMPS and TX40 Robot

In the next sections, the two case studies covered in this work will be presented.

5.1
First Case Study: EMPS

Nowadays most of the positioning and motion control systems are involved in processes of manufacturing, transferring, or testing. Positioning systems can range from pre-engineered linear servo motor tables to rotary tables designed for motor-driven indexing. Figure 5.1 shows an Electro-Mechanical Positioning System (EMPS), which is a standard configuration of a drive system for prismatic joint, it is based on a linear shaft driven by a rotational electromagnetic motor via ballscrew. This is the first case study, which is described in [176]. The benchmark is challenging because of the unknown friction behavior and the marginally stable system dynamics. Figure 5.2 shows the joint force, position, velocity, and acceleration.

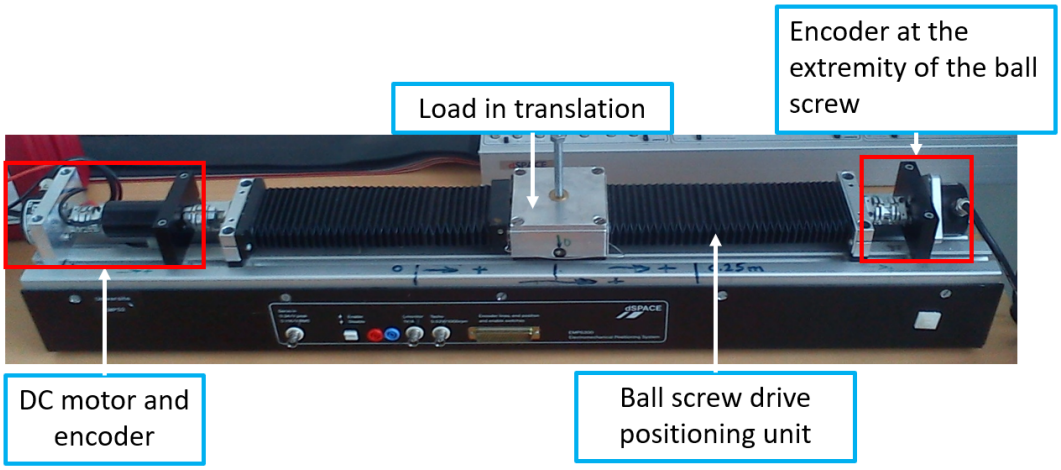

Figure 5.1: Electro-Mechanical Positioning System (EMPS). The image shows some of the components of the EMPS. For more details see [176].

In the benchmark, the motor force (F) is the system input and the measured output is the prismatic joint position ($\mathbf{q_m}$). The joint velocities and accelerations were calculated from $\mathbf{q_m}$, which is firstly filtered and then the derivatives ($\dot{\mathbf{q}}$,$\ddot{\mathbf{q}}$) are calculated with finite differences. The estimation and

validation datasets were constructed from closed-loop experiments performed with the same reference position trajectory. The two datasets are acquired during approximately 25 seconds and are collected at a sampling frequency of 1 kHz. A force disturbance is acting on the system to generate the validation dataset.

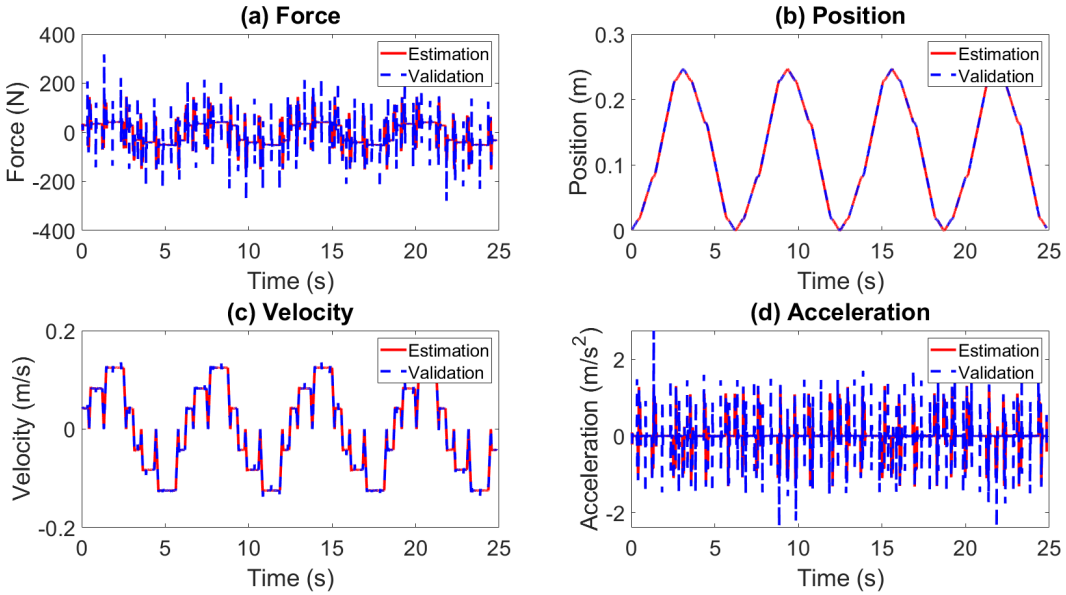

Figure 5.2: Estimation and validation dataset provided for the identification task of the dynamic parameters of the EMPS. For more details see [176].

5.1.1 Problem Formulation

The system identification problem is addressed as a minimization problem with cost function defined as the squared error between the measured data (y_i) and that predicted by the model ($\widehat{y}_i$):

$$minf(\mathbf{X}) = \sum_{i=1}^{N} (y_i - \widehat{y}_i)^2 \tag{5-1}$$

Equation (5-1) can be written in more detail by (5-2). $\mathbf{\Gamma}$ is the $(N \times 1)$ vector of measurements that expresses the joint force, and $\widehat{\mathbf{F}}_{Fric}$ denotes the $(N \times 1)$ vector of estimated friction force, $\widehat{M}$ is the estimated mass and $\ddot{\boldsymbol{q}}$ is the acceleration vector.

$$minf(\mathbf{X}) = \sum \left[\mathbf{\Gamma} - (\widehat{M}\ddot{\mathbf{q}} + \widehat{\mathbf{F}}_{Fric})\right]^2 \tag{5-2}$$

The second term of Equation (5-2) is known as the inverse dynamic model (IDM), which provides, in this case, the joint force in terms of the joint velocities and accelerations (5-3). This model is used to identify the dynamic parameters that are necessary for both control and simulation applications

[166]. We have, therefore, a cost function for each friction model, i. e., we model the frictional effect using the friction models of Chapter 4. Like (4-4), models $\mathbf{F_F^s}$, $\mathbf{F_T^s}$ and $\mathbf{F_V^s}$ will also be evaluated in their asymmetric versions. They will be named as $\mathbf{F_F^a}$, $\mathbf{F_T^a}$ and $\mathbf{F_V^a}$, respectively.

$$\tau_{\mathbf{idm}} = \mathbf{IDM}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \tag{5-3}$$

Some recent works have also addressed the EMPS as one of their case studies [177,178]. The authors used several approaches to simulate the position of the prismatic joint. In their work, the authors reached an R^2 higher than 0.99, a RMSE of 2.64×10^{-3} and a fit of 96.8%. These evaluation metrics were not the same as those used in studies in which the position of the prismatic joint was simulated (Chapter 6: Sections 6.2 and 6.3).

5.2 Second Case Study: TX40 Robot

The Stäubli TX40 6-axis robot is an articulated arm with 6 axes with spherical work envelope. It can be mounted on the floor, wall, or ceiling. The TX40 6-axis robot has a maximum payload of 2.3 kg and a 515 mm reach. It is used mainly in electronics and pharmaceutical companies, where precision and very low cycle times are a priority.

The TX40 robot is one of the case studies of [179]. The author kindly provided some data for a preliminary study presented in Chapter 6, Section 6.4, for the identification of the dynamic parameters of the TX40 robot. In the preliminary study, the data were used only to identify the dynamic parameters of the case study since another database has not yet been provided for the validation of the model. Figure 5.3 shows the torque, position, velocity, and acceleration of the second joint.

The measurements of the position of the joint and the vector of the control signal, calculated according to the control law, are generally the available signals from the robot controller. The IDM method (Equation 5-3) uses a filtered position obtained by filtering measurements of the position through a low-pass Butterworth filter in both the forward and reverse directions using the *filtfilt* MATLAB® function. The derivatives ($\dot{\mathbf{q}}$,$\ddot{\mathbf{q}}$) are calculated offline without phase shift, using a central difference algorithm of the lowpass filtered position.

5.2.1 Problem Formulation

Here the formulation follows the same principle of Subsection 5.1.1:

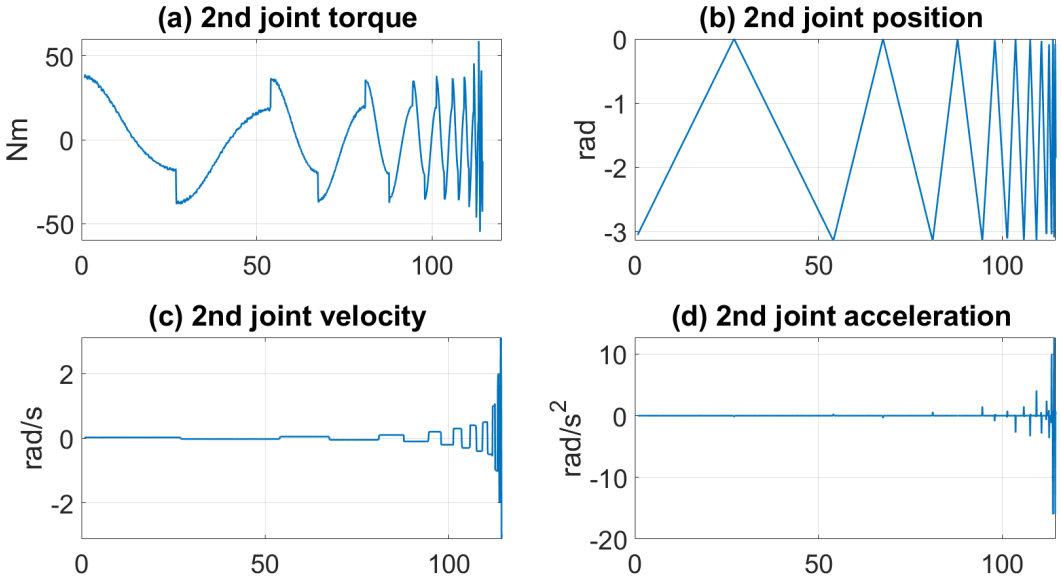

Figure 5.3: Data provided for the identification task of the TX40 robot.

$$minf(\mathbf{X}) = \sum [\tau - \tau_{idm}]^2 \tag{5-4}$$

The IDM of the Stäubli TX40 robot is given by Equation 5-5:

$$\tau_{idm} = ZZ\ddot{\mathbf{q}} + \tau_{Grav}(\mathbf{q}) + \tau_{Fric}(\dot{\mathbf{q}}) + offset \tag{5-5}$$

where τ is the $(N \times 1)$ vector of measurements that expresses the second joint torque; τ_{Grav} is the gravity torque of the second link; τ_{Fric} is the friction torque; ZZ is the total inertia of the second link and $offset$ is an offset parameter.

The gravity torque of the second link is expressed by:

$$\tau_{Grav} = -gMXcos(\mathbf{q}) + gMYsin(\mathbf{q}) \tag{5-6}$$

where g is the gravity constant; MX and MY are the components of the gravity effect and $\mathbf{q}$ is the position of the second link.

6
Parameter Identification of Robotic Manipulators

This chapter is dedicated to the presentation of the contributions of this dissertation. The first case study is covered in the first three sections, while the last section covers the second case study.

6.1
Comparison of Friction Models for Grey-box Identification of an Electromechanical Positioning System

This section covers the contributions presented at the Brazilian Symposium on Intelligent Automation that took place in Ouro Preto in 2019.

6.1.1
Contributions

The contributions of this section are:

(i) Identification of the dynamic parameters of a positioning system;

(ii) Comparison of four friction models in their asymmetric and symmetric versions.

6.1.2
Methods

The data for the identification of the EMPS parameters were made available by [176]. It includes the joint position and the control signal resulting from the control law. First, a lowpass filter is applied to remove any torque ripples because the motor torque τ presents high-frequency disturbances. Besides, a decimation procedure is performed since there is no information on high frequencies because the data is lowpass filtered.

The second term of Equation (5-2), known as the inverse dynamic model (IDM), is rewritten here by Equation (6-1), according to [176, 179]:

$$\tau_{\mathbf{idm}} = M\ddot{\mathbf{q}} + \mathbf{F_f}(\dot{\mathbf{q}}) + offset \tag{6-1}$$

where $\tau_{\mathbf{idm}}$ is the joint torque/force; M is the mass; $\mathbf{F_f}$ is the friction model, which is one of those discussed in Chapter 4; $offset$ is an offset of measurements; $\dot{\boldsymbol{q}}$ and $\ddot{\boldsymbol{q}}$ are, respectively, the velocity and acceleration vectors.

Equations (4-4) and (4-5) are linear in relation to the parameters of the model, for this case a QR factorization was used to solve the linear Least-Squares problem. The Levenberg–Marquardt algorithm [180, 181] was applied to estimate the free parameters of the other friction models, which are given in Table 6.1. This approach has been validated on several industrial robots and prototypes [182–184].

Table 6.1: Models and their parameters.

Friction Model	Free parameters
F_F^s	M, F_V, F_C, of
F_F^a	M, F_V^+, F_C^+, F_V^-, F_C^-
F_C^s	M, F_V, F_C, of
F_C^a	M, F_V^+, F_C^+, F_V^-, F_C^-
F_T^s	M, F_V, F_K, F_S, k, of
F_T^a	M, F_V^+, F_K^+, F_S^+, F_V^-, F_K^-, F_S^-, k^+, k^-
F_V^s	M, F_V, F_C, F_S, of
F_V^a	M, F_V^+, F_C^+, F_S^+, F_V^-, F_C^-, F_S^-

For this contribution the $\mathbf{F_T^s}$ friction model formulated by [173] was adopted, see Equation (6-2). Since some of the specific parameters are extensively used as specified in the literature [165,185,186], they will be not estimated in the optimization process, they are: Stribeck velocity $v_S = 0.001m/s$, Geometry factor $\delta_\sigma = 2$ and Tolerance velocity (Threlfall) $v_0 = 0.001m/s$. Though they must not necessarily be fixed since one could try to find the δ_σ for a specific study case.

$$\mathbf{F_T^s} = F_V\dot{\mathbf{q}} + \left[F_K + (F_S - F_K)\, e^{\frac{|\dot{\mathbf{q}}|}{k}}\right] sign(\dot{\mathbf{q}}) \tag{6-2}$$

6.1.3 Results

In the present section, the results concerning the grey-box approach for the identification of the dynamic parameters of the EMPS using a local search method are presented, as well as the comparison of four friction models in their asymmetric and symmetric versions.

The software MATLAB® and the function *lsqnonlin* were chosen for the optimization process.Table 6.2 shows the estimated dynamic parameters of all models. It can be noticed that there is no significant difference in the estimated values of some parameters of the asymmetric models, such as F_C^+ and F_C^-, F_S^+ and F_S^-. However, F_V^+ and F_V^-, F_K^+ and F_K^- are significantly different from each other, which indicates an asymmetric behavior of the friction. The error of the models with the highest and lowest accuracy are available in Figure 6.1. It shows that F_T^a model has a much lower error amplitude compared to the F_F^s model.

Table 6.2: Estimated parameters of the symmetric and asymmetric models.

Parameters	$\mathbf{F^s_C}$	$\mathbf{F^s_V}$	$\mathbf{F^s_T}$	$\mathbf{F^s_F}$	$\mathbf{F^a_C}$	$\mathbf{F^a_V}$	$\mathbf{F^a_T}$	$\mathbf{F^a_F}$
M (kg)	95.1089	95.108	95.109	95.121	95.1540	95.152	95.149	95.302
$F_V (N/ms^{-1})$	203.5034	203.38	203.54	206.22	-	-	-	-
F_C (N)	20.3935	20.406	-	9.8066	-	-	-	-
F_S (N)	-	19.945	20.393	-	-	-	-	-
F_K (N)	-	-	2222.6	-	-	-	-	-
k (rad/s)	-	-	-62187	-	-	-	-	-
Of (N)	-3.1648	-3.1648	-3.165	-3.1621	-	-	-	-
$F_V^+ (N/ms^{-1})$	-	-	-	-	166.7061	166.04	245.14	168.95
$F_V^- (N/ms^{-1})$	-	-	-	-	240.4236	240.17	240.51	243.03
F_C^+ (N)	-	-	-	-	20.1440	20.209	-	9.7088
F_C^- (N)	-	-	-	-	20.6277	20.653	-	9.9268
F_S^+ (N)	-	-	-	-	-	17.882	18.903	-
F_S^- (N)	-	-	-	-	-	19.691	20.624	-
F_K^+ (N)	-	-	-	-	-	-	22.572	-
F_K^- (N)	-	-	-	-	-	-	-6277.3	-
k^+ (rad/s)	-	-	-	-	-	-	-0.100	-
k^- (rad/s)	-	-	-	-	-	-	1.344e+05	-

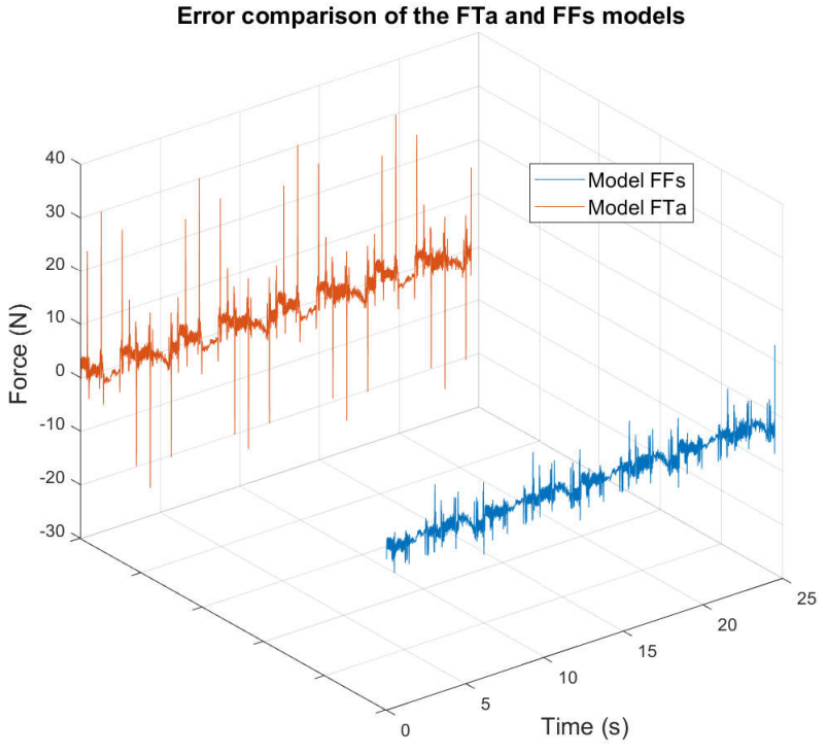

Figure 6.1: Error comparison between the best and worst friction model when the force is predicted by an OSA procedure. F_T^a model has the lowest error and F_F^s the highest.

Figure 6.2 illustrates the behavior of the simulation error of the force of the joint when all friction models are considered. The behavior of the error is, in general, the same for all models, the difference lies, basically, in the error amplitude of each one. A direct comparison between the reconstructed and measured force of the best model performed by an OSA procedure is show by Figure 6.3(a). There is a reasonable resemblance between reconstructed

and measured forces. The relation between measured and estimated forces are shown in Figure 6.3(b), it clearly shows that the estimated parameters of the F_T^a model can describe the dynamic behavior of the EMPS adequately.

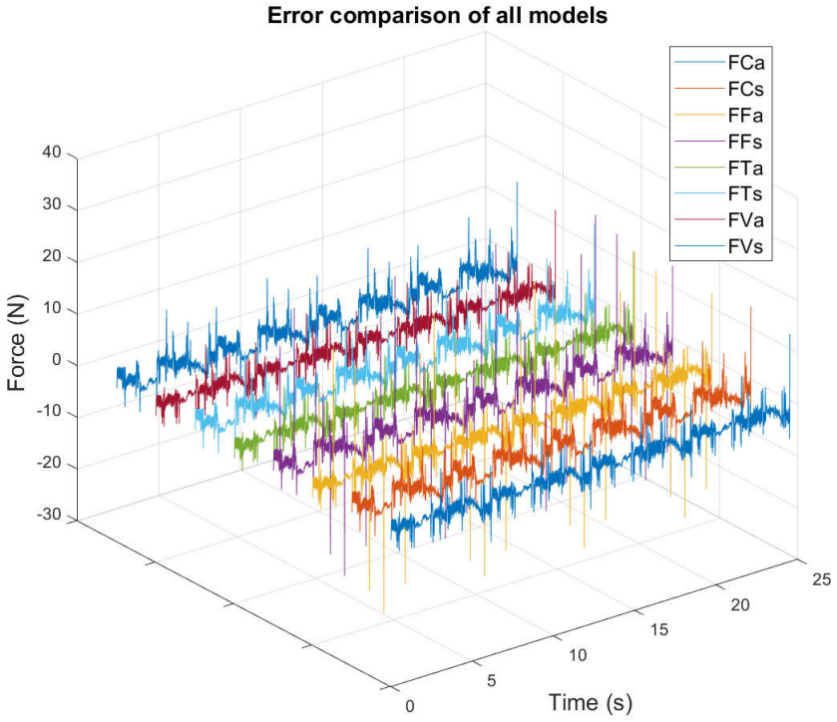

Figure 6.2: Error comparison of all models. The error between the predicted and measured forces has a similar behavior when all friction models are compared, they differ from each other on the amplitude.

Table 6.1.3 shows the results of all evaluation metrics described in Equations (2-13) to (2-16). In this table, the improvement described in percentage was based on the relative error, where each model is compared with the one that has the highest relative error. In Table 6.1.3 we can see that the model with the best metrics is the F_T^a model, as its R^2 approaches unity and the standard deviation is relatively small. F_T^a is the best model, which presents an improvement of 99% in terms of the relative error compared to F_F^s model. This shows the importance of adopting more accurate models in grey-box system identification.

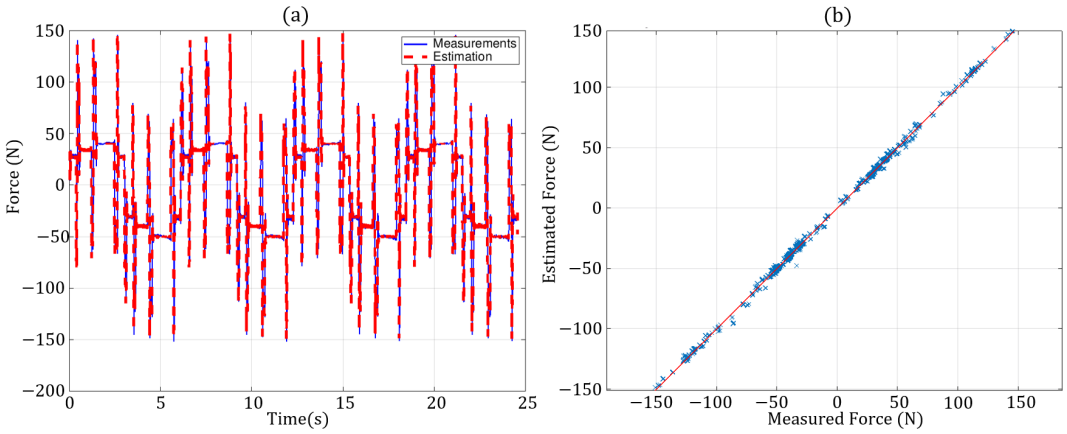

Figure 6.3: (a) Direct comparison of the measured and estimated data of the best model. There is a reasonable resemblance between reconstructed and measured forces when the best friction model is selected for the prediction; (b) Measured force and estimated force of the F_T^a model. It clearly shows that the estimated parameters of the F_T^a model can describe the dynamic behavior of the EMPS adequately. This can be seen also by the evaluation metrics.

Table 6.3: Evaluation of the proposed models.

Model	$\mathbf{R^2}$	δ(N)	ϵ (%)	s (N)	$\mathbf{\Delta}$(%)
F_F^s	0.996	3.2249	6.0299	3.2256	-
F_F^a	0.997	2.8658	5.3584	2.8664	11.14
F_C^s	0.998	2.1839	4.0834	2.1843	32.30
F_T^s	0.998	2.1839	4.0834	2.1843	32.30
F_V^s	0.998	2.1837	4.0831	2.1842	32.30
F_C^a	0.999	1.6642	3.1117	1.6646	48.40
F_V^a	0.999	1.6611	3.1058	1.6614	94.15
F_T^a	0.999	1.6194	3.0276	1.6194	99.16

6.2 Ensemble Grey and Black-box Nonlinear System Identification of a Positioning System

This section covers the contributions presented at the Brazilian Congress of Automation in 2020, a virtual event due to the restrictions imposed on events due to the COVID-19 pandemic.

6.2.1 Contributions

The contributions of this section are:

(i) Identification of the dynamic parameters of a positioning system;

(ii) Development of a hybrid model, consisting of grey and black-box models, used to simulate the output variable of a positioning system;

(iii) Combination of models of different classes to simulate the output variable of a positioning system;

(iv) Comparison of four friction models in their asymmetric and symmetric versions.

6.2.2 Methods

Figure 6.4 shows a schematic of the three approaches used for the simulation of the load's position, namely: (*i*) grey-box model (GBM); (*ii*) black-grey-box model (BGBM); (*iii*) ensemble grey-black-box (EGBB). First, the parameters of a grey-box model are estimated and used for simulating the position of the load of the EMPS ($\widehat{\mathbf{y}}_\mathbf{1}$). Next, The error $\mathbf{e_1}$ between the true position $\mathbf{y}$ of the load and $\widehat{\mathbf{y}}_\mathbf{1}$ is modeled by a black-box approach using a NARX ANN model. By summing the modeled error ($\widehat{\mathbf{e}}_\mathbf{1}$) and the estimated position $\widehat{\mathbf{y}}_\mathbf{1}$, the second estimated position ($\widehat{\mathbf{y}}_\mathbf{2}$) is generated. Finally, an ensemble (EGBB) is built by the combination of both GBM and BGBM models. It is given a weight to every single model so that a more accurate estimation of the load's position can be achieved. Next, each of these three approaches are depict separately.

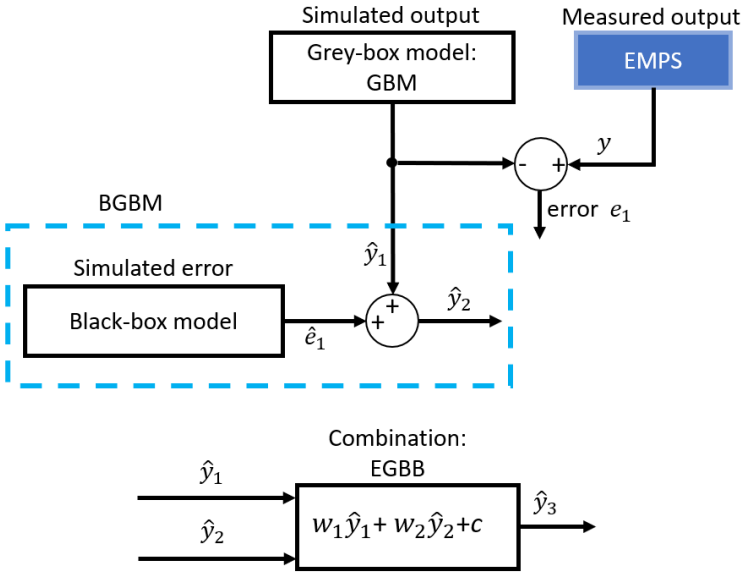

Figure 6.4: Formulation of the three approaches. GBM is the first approach, it generates $\widehat{\mathbf{y}}_\mathbf{1}$. The error of the GBM approach ($\mathbf{e_1}$) is modeled to generate the estimated position of the second approach (BGBM), which is a combination of a grey and black-box models. Finally, an ensemble of $\widehat{\mathbf{y}}_\mathbf{1}$ and $\widehat{\mathbf{y}}_\mathbf{2}$ is built by using a simple linear regression.

The parameters of the GBM approach were estimated according to the methodology of Subsection 6.1.2, but here the $\mathbf{F^s_T}$ friction model formulated by [174] was adopted and in the optimization process v_S, δ_σ, and v_0 were also estimated. All necessary parameters for the simulation of the position of the joint of the EMPS can be found in Tables 6.4 and 6.5.

Table 6.4: Parameters of the asymmetric friction models estimated by the optimization process.

Parameter	$\mathbf{F^a_C}$	$\mathbf{F^a_F}$	$\mathbf{F^a_T}$	$\mathbf{F^a_V}$
M (kg)	95.1540	95.154	95.547	95.153
$F_V^+(N/ms^{-1})$	166.7061	166.71	178.48	166.1
$F_V^-(N/ms^{-1})$	240.4236	240.42	218.67	240.19
F_C^+ (N)	20.1440	20.144	25.014	20.204
F_C^- (N)	20.6277	20.628	28.028	20.651
F_S^+ (N)	-	-	25.014	18.708
F_S^- (N)	-	-	21.578	20.058
δ_σ	-	-	-	1.8258
v_0 (m/s)	-	0.0032239	-	-
v_S (m/s)	-	-	0.004156	0.00063898

Table 6.5: Parameters of the symmetric friction models estimated by the optimization process.

Parameter	$\mathbf{F^s_C}$	$\mathbf{F^s_F}$	$\mathbf{F^s_T}$	$\mathbf{F^s_V}$
M (kg)	95.109	95.109	95.681	95.105
$F_V(N/ms^{-1})$	203.503	203.5	203.36	200.85
$F_C(N)$	20.393	20.393	17.023	20.639
$F_S(N)$	-	-	17.023	18.968
$F_K(N)$	-	-	-	-
$Of(N)$	-3.165	-3.165	-4.123	-3.146
δ_σ	-	-	-	1.236
v_0 (m/s)	-	5.413E-3	-	-
v_S (m/s)	-	-	6.464E-3	5.483E-3

Since the true output of the EMPS can be described by $\hat{\mathbf{y}}_\mathbf{1}$ and $\mathbf{e_1}$, as shown by Figure 6.4, the aim of the first part of BGBM approach is to model $\mathbf{e_1}$ using a NARX model (6-3), and sum the modeled error with $\hat{\mathbf{y}}_\mathbf{1}$ to obtain a more accurate response than that found by the first approach. The input ($\mathbf{u}$) and output ($\mathbf{y}$) considered are, respectively, the force and the error $\mathbf{e_1}$.

$$\begin{aligned} y(k) = F[&y(k-1), y(k-2), ..., y(k-n_y), \\ &u(k-d), u(k-d-1), ..., u(k-d-n_u)] \end{aligned} \tag{6-3}$$

where $y(k)$, $u(k)$ are the system output and input, respectively; n_y and n_u are, respectively, the maximum lags for the system output and input; F is some nonlinear function, and d is a time delay.

The model, based on the linear ARX model [114], is essentially an expansion of past inputs and outputs, but the nonlinear ARX model uses a nonlinear mapping function F between the input and output data. Here we use a NARX neural network structure, described in Section 2.1, for the formulation of the model.

Finally, the strategy adopted for the last approach is a linear regression, we may find the relationship between the true output of the EMPS and the two approaches $\widehat{\mathbf{y}}_\mathbf{1}$ and $\widehat{\mathbf{y}}_\mathbf{2}$ (predictors). A weight is attributed to each different solution so that the approximation given by $\widehat{\mathbf{y}}_\mathbf{3}$ can be better than $\widehat{\mathbf{y}}_\mathbf{1}$ and $\widehat{\mathbf{y}}_\mathbf{2}$ individually. The input arguments for the linear regression are $\widehat{\mathbf{y}}_\mathbf{1}$ and $\widehat{\mathbf{y}}_\mathbf{2}$ as predictors and $\mathbf{y}$ as the response variable.

6.2.3 Results

In this section, the results concerning the application of grey and black-box models as well as their combination for the simulation of the position of the joint of the EMPS are provided. The whole system was simulated in the software MATLAB® according to Figure 6.5, *ode45* was used as the solver for the differential equation to simulate the position of the joint with the parameters estimated in the optimization process. The software MATLAB® was used to define the parameters of the black-box model, which are its orders *na*, *nb* and the number of neurons M of the wavelet network nonlinearity estimator that is used for the mapping of the nonlinearity of the model. The values of *na* and *nb* ranged from 2 to 15 and the number of neurons from 10 to 30.

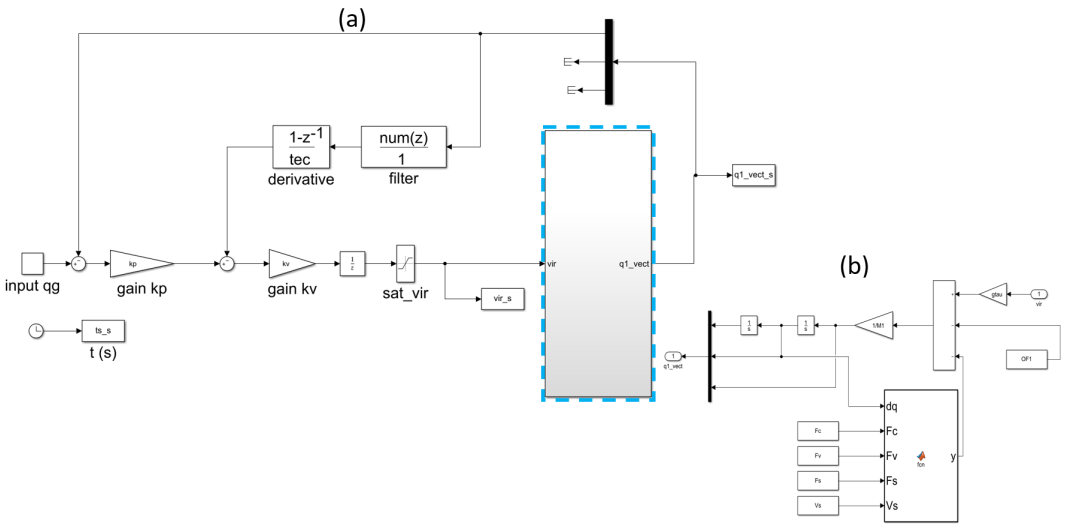

Figure 6.5: The positioning system was modeled and simulated on Simulink®. Figure 6.5(a) shows the whole system, while the blue dashed block is detailed by Figure 6.5(b), which represents the $\mathbf{F}^\mathbf{s}_\mathbf{V}$ friction model.

Figure 6.6 shows the comparison between the measured and estimated position using the BGBM approach ($\widehat{\mathbf{y}}_\mathbf{2}$). $\mathbf{F}^\mathbf{a}_\mathbf{C}$ and $\mathbf{F}^\mathbf{s}_\mathbf{T}$ are the friction models ($\widehat{y}_2 F^a_C$, $\widehat{y}_2 F^s_T$), which are considered as the best models of this approach since

they achieved higher relative error improvement. The estimated position is almost perfectly modeled by both models, showing their reasonable resemblance. The comparison was made taking into account the relative error of the friction model proposed in the benchmark (symmetric Coulomb model with viscous friction), and the models proposed here, see Table 6.9. In the benchmark, the relative errors are 0.013752% and 0.0080248% for the estimation and validation sets, respectively.

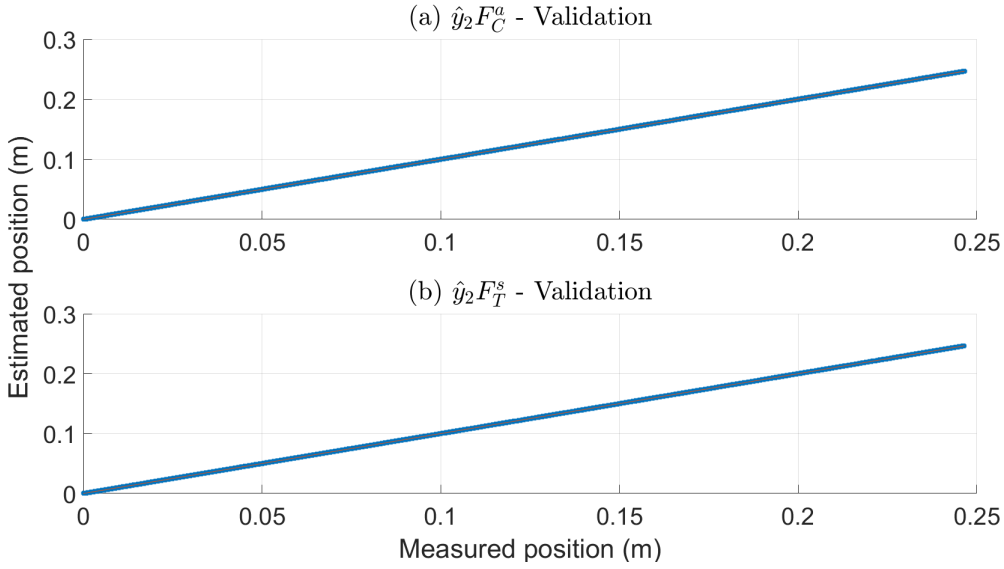

Figure 6.6: Comparison between the measured and estimated output given by models $\hat{y}_2F_C^a$ and $\hat{y}_2F_T^s$. It can be seen that measured and estimated position have high correspondence, confirming the low relative errors given by Table 6.7

Tables 6.6 - 6.8 show the parameters of all the models and their evaluation metrics. Considering the GBM approach ($\hat{\mathbf{y}}_\mathbf{1}$), from the six proposed models, only $\hat{y}_1F_T^s$ can bring some error improvement considering estimation and validation sets. Table 6.7 shows clearly that the BGBM approach is the most suitable for better modeling the position of the load of the EMPS. According to the metrics, $\hat{y}_2F_V^s$ and $\hat{y}_2F_C^s$ have, respectively, the lowest relative errors for estimation (ε_{Est}) with 0.0071464% and 0.0072538%. $\hat{y}_2F_T^s$ and $\hat{y}_2F_C^a$ with 0.0063456% and 0.0069405%, have respectively, the lowest relative errors for the validation set (ε_{Val}). For the estimation set, all models of $\hat{\mathbf{y}}_\mathbf{2}$ perform better than the model proposed by the benchmark. We can see the same behavior at the validation set, except by $\hat{y}_2F_F^a$ and $\hat{y}_2F_F^s$ models. Table 6.8 shows the results of the EGBB approach ($\hat{\mathbf{y}}_\mathbf{3}$). In this case, 50% of the models were able to perform better than the original friction model proposed in the benchmark for the validation set. The best friction models for this approach are $\hat{y}_3F_T^s$ and $\hat{y}_3F_C^s$.

Table 6.9 shows the percentage of improvement achieved by each approach and for every model. The model with the highest relative error improvement for the estimation set was $\widehat{y}_2F_V^s$ (48.03%), followed by $\widehat{y}_2F_C^s$ (47.25%) and $\widehat{y}_2F_T^s$ (46.68%). For the validation set, $\widehat{y}_2F_T^s$ had the highest error improvement percentage (20.92%), followed by $\widehat{y}_3F_T^s$ and $\widehat{y}_2F_C^a$ models with, 18.59% and 13.51% respectively.

Table 6.6: Parameters and metrics of the GBM approach. $\widehat{y}_1F_T^s$ and $\widehat{y}_1F_V^s$ show relative error improvement for the estimation set compared to $\widehat{y}_1F_C^s$, which is the friction model proposed by the benchmark. Except by $\widehat{y}_1F_F^a$, $\widehat{y}_1F_F^s$ and $\widehat{y}_1F_V^s$ all other models show relative error improvement.

Model	**Parameters**	$\varepsilon_{Est}(\%)$	$\varepsilon_{Val}(\%)$
$\widehat{y}_1F_C^a$	M=95.15540; F_V^+=166.7061; F_V^-=240.4236; F_C^+=20.1440; F_C^-=20.6277;	0.0137580	0.0079672
$\widehat{y}_1F_C^s$	M=95.1089; F_V=203.5034; F_C=20.3935; Of=-3.1648	0.0137522	0.0080248
$\widehat{y}_1F_F^a$	M=95.154; F_V^+=166.71; F_V^-=240.42; F_C^+=20.144; F_C^-=20.628; v_o=0.0032239	0.0139793	0.0120720
$\widehat{y}_1F_F^s$	M=95.109; F_V=203.5; F_C=20.393; Of=-3.1648 v_o=0.0054125	0.0139503	0.0121076
$\widehat{y}_1F_T^a$	M=95.547; F_V^+=178.48; F_V^-=218.67; F_C^+=25.014; F_C^-=28.028; F_S^+=25.014; F_S^-=21.578; v_S=0.004156;	0.0149361	0.0079732
$\widehat{y}_1F_T^s$	M=95.681; F_V=203.36; F_S=17.023; F_C=17.023; v_S=0.006464;Of=-4.1234	0.0125696	0.0079662
$\widehat{y}_1F_V^a$	M=95.153; F_V^+=166.1; F_V^-=240.19; F_C^+=20.204; F_C^-=20.651; F_S^+=18.708; F_S^-=20.058; v_S=0.00063898 δ_σ=1.8258	0.0137678	0.0079584
$\widehat{y}_1F_V^s$	M=95.105; F_V=200.85; F_C=20.639; F_S=18.968; Of=-3.1458; δ_σ=1.2375 v_S=0.0054831	0.0137384	0.0080252

Table 6.7: Parameters and metrics of the BGBM approach. It shows that $\widehat{y}_2F_V^s$ and $\widehat{y}_2F_T^s$ have the lowest relative errors for the estimation and validation sets respectively.

Model	**na**	**nb**	**M**	$\varepsilon_{Est}(\%)$	$\varepsilon_{Val}(\%)$
$\widehat{y}_2F_C^a$	5	13	10	0.0073630	**0.0069405**
$\widehat{y}_2F_C^s$	4	13	12	**0.0072538**	0.0070701
$\widehat{y}_2F_F^a$	3	13	14	0.0077180	0.0119043
$\widehat{y}_2F_F^s$	3	4	29	0.0081926	0.0108225
$\widehat{y}_2F_T^a$	6	3	10	0.0118427	0.0072983
$\widehat{y}_2F_T^s$	4	13	12	0.0073332	**0.0063456**
$\widehat{y}_2F_V^a$	4	6	15	0.0091737	0.0077414
$\widehat{y}_2F_V^s$	4	13	15	**0.0071464**	0.0073584

Table 6.8: Parameters and metrics of the third approach. Model $\widehat{y}_3F_T^s$ and $\widehat{y}_3F_C^s$ have lowest relative errors compared to the model proposed by the benchmark.

Model	**c**	**w1**	**w2**	$\varepsilon_{Val}(\%)$
$\widehat{y}_3F_C^a$	3.986E-06	0.0279	0.9721	0.0072700
$\widehat{y}_3F_C^s$	3.248E-07	0.0716	0.9284	**0.0071669**
$\widehat{y}_3F_F^a$	5.223E-06	0.1173	0.8826	0.0121304
$\widehat{y}_3F_F^s$	7.612E-06	-0.4233	1.4232	0.0118544
$\widehat{y}_3F_T^a$	1.866E-05	-0.0438	1.0438	0.0118841
$\widehat{y}_3F_T^s$	8.767E-08	0.1184	0.8816	**0.0065332**
$\widehat{y}_3F_V^a$	3.695E-06	-0.9112	1.9112	0.0085389
$\widehat{y}_3F_V^s$	8.294E-07	0.0683	0.9317	0.0073771

Table 6.9: General comparison between all approaches. It shows that $\widehat{y}_2F_V^s$ and $\widehat{y}_2F_T^s$ have the best performance for estimation and validation sets respectively.

Model	$\mathbf{\Delta}_{Est}(\%)$	$\mathbf{\Delta}_{Val}(\%)$	**Model**	$\mathbf{\Delta}_{Est}(\%)$	$\mathbf{\Delta}_{Val}(\%)$
$\widehat{y}_1\ F_C^a$	-0.04	0.72	$\widehat{y}_2F_T^a$	13.89	9.05
$\widehat{y}_2\ F_C^a$	46.46	13.51	$\widehat{y}_3F_T^a$	-	-22.76
$\widehat{y}_3\ F_C^a$	-	9.41	$\widehat{y}_1F_T^s$	8.60	0.73
$\widehat{y}_2\ F_C^s$	**47.25**	11.90	$\widehat{y}_2F_T^s$	46.68	**20.92**
$\widehat{y}_3\ F_C^s$	-	10.69	$\widehat{y}_3F_T^s$	-	**18.59**
$\widehat{y}_1\ F_F^a$	-1.65	-50.43	$\widehat{y}_1F_V^a$	-0.11	0.83
$\widehat{y}_2\ F_F^a$	43.88	-48.34	$\widehat{y}_2F_V^a$	33.29	3.53
$\widehat{y}_3\ F_F^a$	-	-51.16	$\widehat{y}_3F_V^a$	-	-6.4
$\widehat{y}_1\ F_F^s$	-1.44	-50.88	$\widehat{y}_1F_V^s$	0.10	0.00
$\widehat{y}_2\ F_F^s$	40.43	-34.86	$\widehat{y}_2F_V^s$	**48.03**	8.30
$\widehat{y}_3\ F_F^s$	-	-47.72	$\widehat{y}_3F_V^s$	-	8.07
$\widehat{y}_1\ F_T^a$	-8.61	0.64			

The friction models proposed by the GBM approach has the lowest performance among all approaches. It is due to the number of parameters being estimated and also to the initial conditions, which have a high influence on the performance of this approach. BGBM approach has shown to be efficient on the error modeling since only two of sixteen models were not able to perform better than the model proposed in the benchmark. Therefore, the EGBB approach has also proven to be effective, which achieved better performance on 50% of the models, among then $\widehat{y}_3F_T^s$ achieved almost 20% of relative error improvement. Finally, the models that smooth the discontinuity at zero velocity of the Coulomb friction model have better performance, and the symmetric models are also able to perform better than the asymmetric models.

6.3 Comparison of Metaheuristic Algorithms and Friction Models on the Identification of a Positioning System

This section compares metaheuristic algorithms used to estimate the dynamic parameters of EMPS as well as symmetric and asymmetric friction models chosen to compensate for this phenomenon in the case study.

6.3.1 Contributions

The contributions of this section are:

(i) Performance evaluation of five EAs in the task of estimating the dynamic parameters of a positioning system;

(ii) Investigation of the best friction model to describe the friction behavior of a positioning system.

6.3.2 Methods

The estimation of the dynamic parameters of the EMPS considers eight different friction models, as introduced in Chapter 4, four symmetric and four asymmetric. Here the most common version of the $\mathbf{F_T^s}$ friction model was adopted [170–172]. The specific parameters of each algorithm are set as suggested in the literature [126], they can be found in Tables 6.10 and 6.11. The average minimum cost per function evaluation and the mean population searching distance are presented to enable the analyses. The whole system was simulated in the software MATLAB®.

Table 6.10: Specific parameters of each EA and how they were tuned. They are: Initial pheromone amount (τ_0); Evaporation rate (ρ); Elite ants per generation (N_{ACO}); Scale Factor (F); Crossover constant (CR).

ACO	**BBO**	**DE**
Population size: 50	Population size: 50	Population size: 50
Generation limit: 100	Generation limit: 100	Generation limit: 100
τ_0: 1e-6	Sinusoidal migration	/bin variation
ρ: 0.9	Migration blend parameter: 0.50	F (constant): 0.4
N_{ACO}: 2		CR: 0.9

Table 6.11: Specific parameters of each EA and how they were tuned. They are: Elite individuals per generation (N_{GA}); Crossover type (X_{over}); Constriction coefficient (K_F); Maximum learning rates (μ_l).

GA	**PSO**
Population size: 50	Population size: 50
Generation limit: 100	Generation limit: 100
N_{GA}: 2	K_F: 0.25
ρ_c: 0.02	Neighborhood size: 6
X_{over}: single point	μ_l: 2.5

6.3.3 Results

Figures 6.7-6.9 show the results of the search quality metric and convergence measure when five EAs are applied to the task of identifying the dynamic parameters of the EMPS considering eight friction models. Each algorithm runs 20 times with different random numbers of seeds. By the analyses of the curves in Figures 6.7 and 6.8, it is possible to summarize an applicable pattern to all five EAs. The solution is rapidly improved during the early searching stage, followed by gradual enhancement with a slow rate. DE has the lowest minimum cost during the early stage, however, as the number of function evaluations increases, it was outperformed by GA and BBO considering the cases of F_F^a, F_V^a and F_V^s friction models. We can distinguish two different behaviors among the five EAs. The first group composed of DE and PSO is characterized by rapid improvement during the early searching stage followed by a steady-state condition (BBO seems to have a steady-state condition for F_T^a). The remaining EAs belong to the other group, which is also characterized by rapid improvement during the early searching stage, however, they can consistently improve their searching quality throughout the searching process. But, in later stages, the value of the cost function decreases slowly.

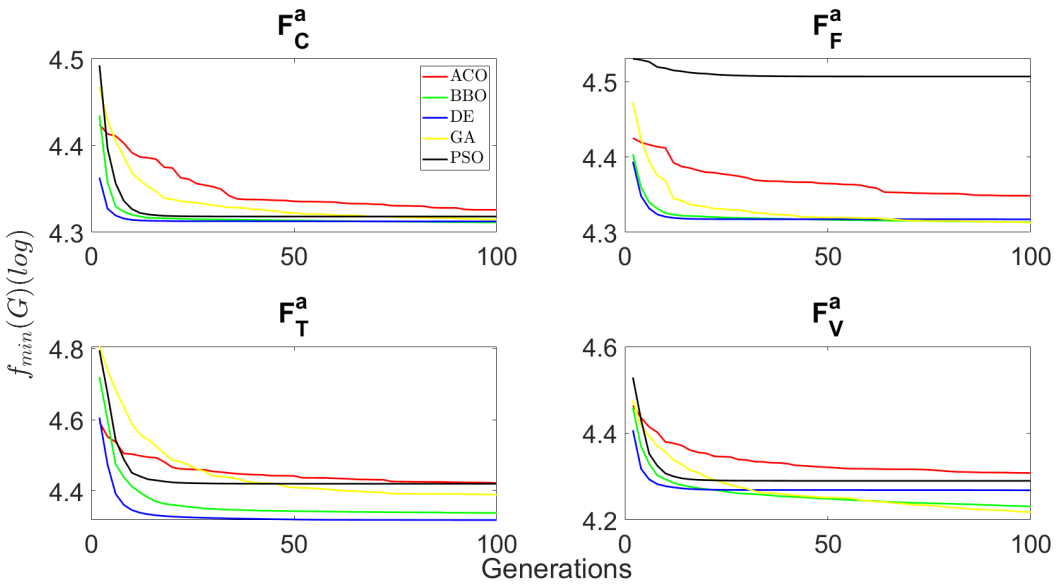

Figure 6.7: Minimum value of the cost function of the asymmetric friction models. It is possible to see that DE is the first to achieve steady-state condition compared to the other EAs (normally before 1000 evaluations/20 generations), it is followed by PSO. DE has on average lower minimum cost per function evaluation, but it was outperformed by BBO and GA for F_V^a case.

Figure 6.9 shows the convergence metric when ACO, BBO, DE, GA, and PSO are compared. As shown, the $dist_{mean}(G)$ values decrease at a faster rate

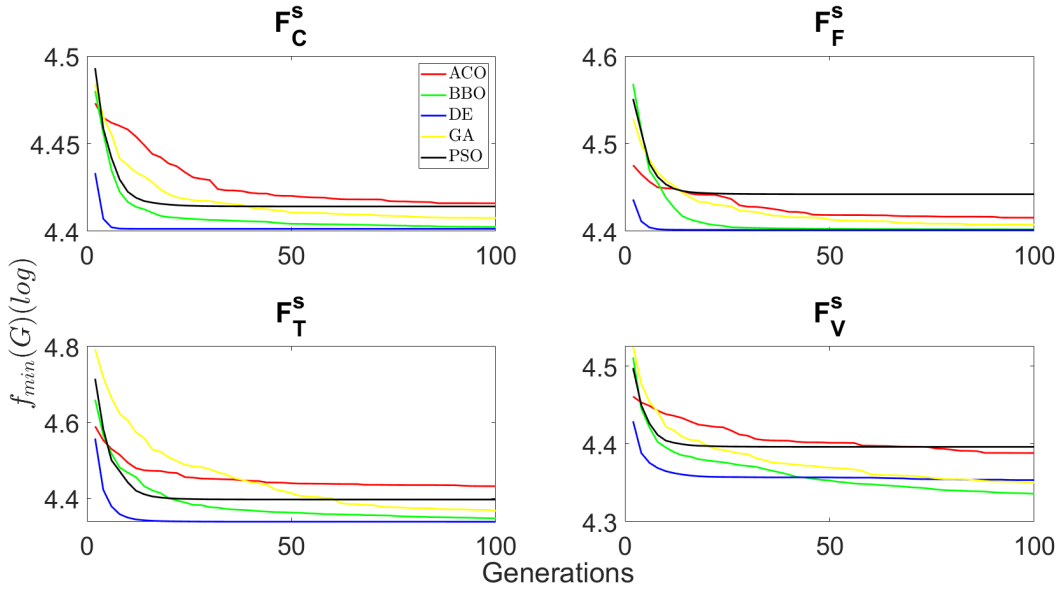

Figure 6.8: Minimum value of the cost function of the symmetric friction models. The figure shows that the behavior of the EAs for both symmetric and asymmetric friction models are the same. It can be seen clearly that from the five EAs, one group is composed of DE and PSO and the other by ACO, BBO, and GA. The former group achieved steady-state condition before 2000 cost function evaluations (40 generations) while the other keeps decreasing its cost slowly, but not necessarily achieving steady-state condition.

during the early stage, it is related to the rapid improvement of the quality of the solution shown by Figures 6.7 and 6.8. Concerning the convergence speed, ACO has the fastest convergence speed in the initial searching stage for all friction models, but DE, with low fluctuation though, and PSO converge quickly during the middle and later stages. ACO keeps fluctuating after achieving low $dist_{mean}(G)$, GA has also this fluctuation but with much higher $dist_{mean}(G)$. It means that GA maintains a high population diversity during the entire process. The convergence speed is not affected by the different friction models. Figure 6.9 also shows, in accordance with Figures 6.7 and 6.8, that the focus of DE and PSO lies on the searching in the surrounding regions of the identified best solution.

Tables A-1 and A-2 show the average and standard deviation of the estimated parameters when the asymmetric and symmetric versions of each friction model were considered. It can be noticed that, in general, there is no significant difference in the values estimated by each EA for every friction model. However, F_V^+ and F_V^-, coefficients for positive and negative velocities (asymmetric models), indicate an asymmetric behavior of the friction since they are significantly different from each other.

Tables A-3 and A-4 show, respectively, the relative errors of the simula-

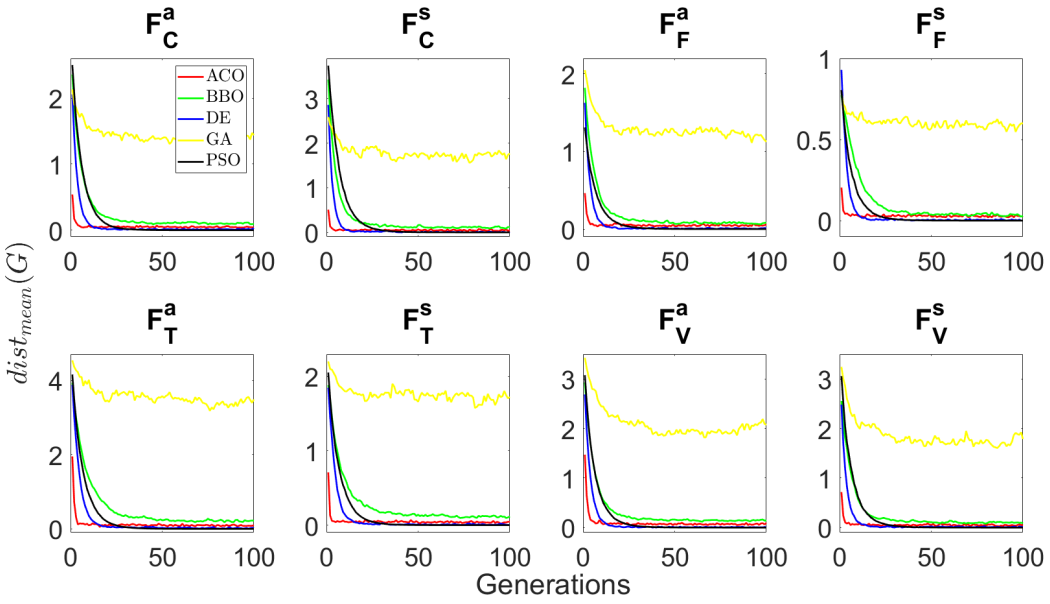

Figure 6.9: Mean population searching distance per generation $dist_{mean}(G)$. Except by GA, the distance between the members of the population is close to zero before 50 generations. The lower average distance indicates that the algorithm is searching for a solution under a tight search space, i.e, the algorithm is converging. ACO can achieve a small distance among the population members before all other EAs, but it fluctuates in the later search stage, while PSO does not show fluctuations, which is particularly high by GA.

tion of the position of the joint when both asymmetric and symmetric Coulomb friction models are considered. μ and σ are, respectively, the average and standard deviation of each column. The percentage of improvement achieved by each test over the parameter estimation approach using the standard Least-Squares inverse dynamic model (LS-IDIM) employed on the benchmark is represented by $\mathbf{\Delta_{Val}}(\%)$. The comparison is performed with the relative errors of [176], see Table 6.12. Considering the asymmetric model, the LS-IDM method can outperform the results achieved by the EAs, except by two tests: number 2 (GA) and number 11 (ACO) with 0.007876% and 0.007915% of relative error, respectively. When the symmetric friction models are compared, more tests (highlighted in boldface) were able to outperform the benchmark.

Table 6.12: Relative errors of the IDM approach.

Model	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	**Simulation performed with parameters of**
Assymetric	0.013769%	0.0079575%	[176]
Symmetric	0.013752%	0.0080248%	
Assymetric	0.013777%	0.0079494%	[179]
Symmetric	0.01376%	0.0080178%	

Tables A-5 to A-10 present the results of three others friction models in their symmetric and asymmetric versions, these models are not considered in [176] and [179] for the study of the frictional behavior of the EMPS. Their

relative errors were compared with those of [176] and [179] and are highlighted in boldface if any test can outperform at least one of the relative errors of [176] and [179]. In general, the average relative error for the estimation set is lower when compared to those of [176] and [179]. Considering estimation and validation sets, the relative error was lower for all EAs only when F_V^a was used as the friction model. The lowest relative errors are on average 0.013055% and 0.007930% for estimation and validation sets respectively, they were achieved by PSO and GA using F_F^a as the friction model.

Table 6.13 shows a general rank considering each test individually, $\mathbf{\Delta_{asym}}(\%)$ and $\mathbf{\Delta_{sym}}(\%)$ consider the improvement over the average of the relative errors of the simulations performed by [176] and the simulations using the parameters estimated by [179], see Table 6.12. DE and PSO were able to find the best solutions, the former achieved 26.86% and 27.48% of relative error improvement for the asymmetric and symmetric models respectively, while the latter achieved 10.44% and 11.20%. Considering the first 10 best tests, DE has five of them, followed by GA and ACO with two and PSO with one test. The best results were achieved by F_T^a and F_T^s friction models. According to Table 6.13, the best results were achieved when the friction models that smooth the discontinuity at zero velocity of Coulomb's friction model were considered.

Table 6.13: General rank of the 30 best friction models.

Test	**EA**	**Fric. Model**	$\varepsilon_{\mathbf{Val}}(\%)$	$\mathbf{\Delta_{asym}}(\%)$	$\mathbf{\Delta_{sym}}(\%)$
11	DE	F_T^a	5.817E-03	26.86	27.48
10	DE	F_T^a	5.971E-03	24.93	25.56
7	DE	F_T^a	7.040E-03	11.49	12.24
2	PSO	F_T^s	7.123E-03	10.44	11.20
12	GA	F_T^a	7.201E-03	9.45	10.22
19	DE	F_T^a	7.231E-03	9.09	9.86
10	ACO	F_T^a	7.271E-03	8.58	9.35
5	DE	F_T^a	7.419E-03	6.72	7.50
6	GA	F_T^a	7.467E-03	6.12	6.91
7	ACO	F_T^s	7.542E-03	5.17	5.97
10	GA	F_T^a	7.566E-03	4.88	5.68
13	ACO	F_T^a	7.614E-03	4.26	5.07
17	ACO	F_T^a	7.681E-03	3.42	4.24
19	ACO	F_T^a	7.688E-03	3.34	4.16
4	GA	F_F^a	7.717E-03	2.97	3.79
6	ACO	F_T^a	7.720E-03	2.94	3.76
12	GA	F_F^a	7.723E-03	2.90	3.72
10	ACO	F_V^s	7.756E-03	2.48	3.31
14	DE	F_T^a	7.767E-03	2.35	3.17
5	DE	F_F^a	7.770E-03	2.31	3.13
1	ACO	F_V^a	7.770E-03	2.31	3.13
4	GA	F_T^a	7.787E-03	2.09	2.92
5	ACO	F_T^s	7.789E-03	2.07	2.90
1	DE	F_F^a	7.800E-03	1.93	2.76
9	PSO	F_V^a	7.803E-03	1.89	2.72
14	PSO	F_T^s	7.803E-03	1.89	2.72
18	GA	F_F^s	7.804E-03	1.88	2.71
20	GA	F_T^a	7.811E-03	1.79	2.63
1	GA	F_F^s	7.818E-03	1.71	2.54
3	ACO	F_V^a	7.836E-03	1.48	2.31

Figure 6.10 shows the result of a multiple comparison procedure. Multiple

pairwise comparisons were performed to verify whether the results achieved by each EA and each friction model are significantly different among their groups. Since the distribution of the data (relative errors) is non-normal, the nonparametric Friedman's test was used to analyze the effect of the EAs and the friction models on the relative error. All comparison intervals for the EAs overlap, which means that they are not significantly different from one another on the task of searching for the optimal solution. Concerning the friction models, there is no significant difference under the estimated position of the EMPS if one uses any of them to model the friction term, except by F_T^s and F_V^a. These are the models that are significantly different from each other for the EMPS case study. Tables 6.14 and 6.15 show the probability values (p-Values) of the pairwise comparison. For 95% of confidence level, the p-Value of a pairwise comparison should be lower than 0.05 to assume a significant difference under the groups being compared. According to the results, also already shown in Figure 6.10, none of the groups being compared are significantly different from each other considering the EAs, but the p-Value for F_T^s and F_V^a comparison is 0.016487 for the validation set. This confirms the information given in Figure 6.10.

Table 6.14: Probability value (p-Value) for multiple comparisons between AEs.

EA	**EA**	**p-Value Estimation**	**p-Value Validation**
ACO	BBO	0.997	0.81948
ACO	DE	0.99451	0.95689
ACO	GA	0.97799	0.91201
ACO	PSO	1	0.99982
BBO	DE	1	0.99573
BBO	GA	0.99931	0.99951
BBO	PSO	0.9971	0.72526
DE	GA	99977	0.99985
DE	PSO	0.99467	0.90872
GA	PSO	0.97841	0.84269

Table 6.15: Probability value (p-Value) for multiple comparisons between friction models.

Groups	**p-Value Est.**	**p-Value Val.**	**Groups**	**p-Value Est.**	**p-Value Val.**
F_C^a - F_C^s	0.99999	0.98129	F_F^a - F_T^a	0.99744	0.88076
F_C^a - F_F^a	1	0.99883	F_F^a - F_T^s	0.85155	0.46403
F_C^a - F_F^s	1	0.99439	F_F^a - F_V^a	1	0.87692
F_C^a - F_T^a	0.98843	0.51501	F_F^a - F_V^s	0.98972	1
F_C^a - F_T^s	0.74879	0.14483	F_F^s - F_T^a	0.99837	0.9386
F_C^a - F_V^a	0.99996	0.99535	F_F^s - F_T^s	0.87256	0.58134
F_C^a - F_V^s	0.99782	0.9988	F_F^s - F_V^a	1	0.79351
F_C^a - F_F^a	0.99976	0.99997	F_F^s - F_V^s	0.98591	1
F_C^s - F_F^s	0.99955	1	F_T^a - F_T^s	0.99612	0.99754
F_C^s - F_T^a	0.94326	0.974	F_T^a - F_V^a	0.9996	0.12148
F_C^s - F_T^s	0.55835	0.69838	F_T^a - F_V^s	0.78624	0.88174
F_C^s - F_V^a	0.9982	0.68792	F_T^s - F_V^a	0.91909	**0.016487**
F_C^s - F_V^s	0.99995	0.99997	F_T^s - F_V^s	0.31081	0.46564
F_F^a - F_F^s	1	1	F_V^a - F_V^s	0.97062	0.87591

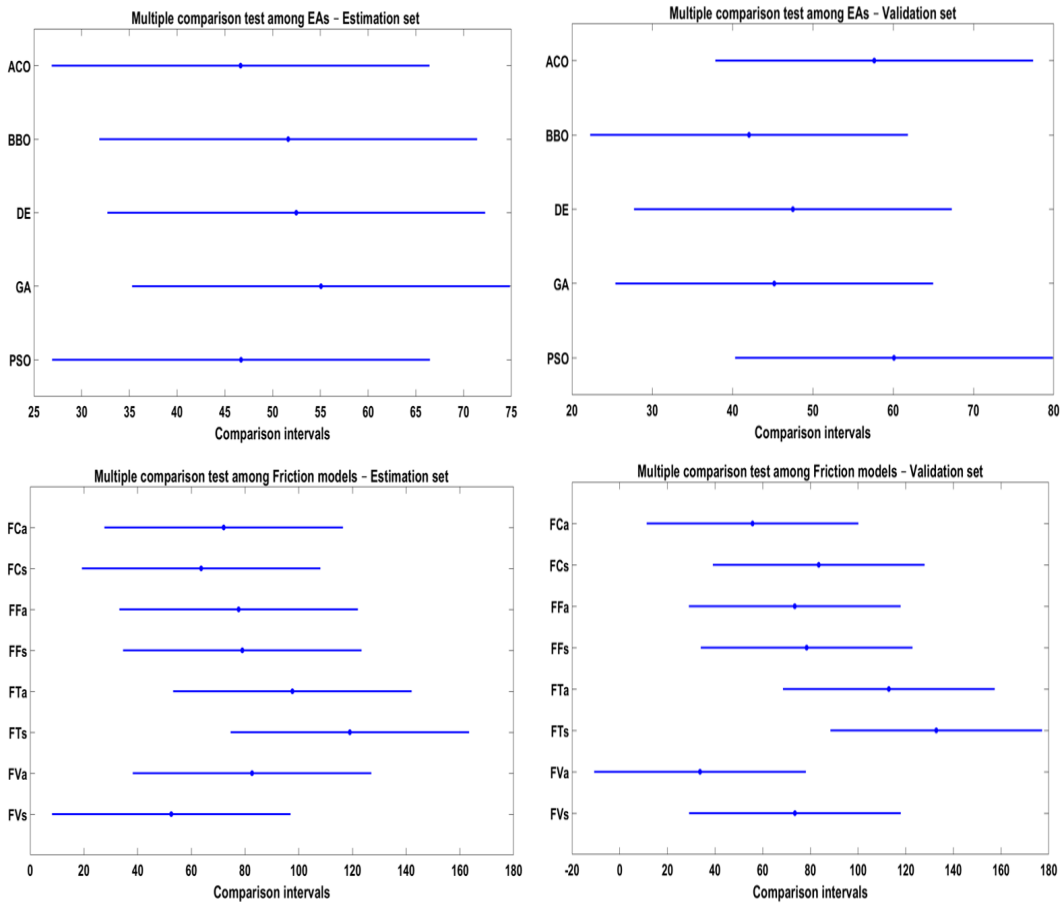

Figure 6.10: Multicomparison test. Comparison intervals are overlapping each other when EAs are compared. It indicates that the algorithms are not significantly different from one another on the task of searching for the optimal solution. Concerning the friction models, there is no significant difference under the estimated position of the EMPS if one uses any of them, except by F_T^s and F_V^a when they are tested on the validation set.

6.4 Identification of the TX40 Dynamic Parameters

In this section, different classes of algorithms used to estimate the dynamic parameters of a robotic manipulator that considers a nonlinear friction model are compared.

6.4.1 Contribution

The contribution of this section is:

(i) Identification of the dynamic parameters of a TX40 robot considering a nonlinear friction model with Stribeck effect.

6.4.2 Methods

The IDM used for the Stäubli TX40 robot is given by Equation 6-4:

$$\tau_{idm} = ZZ\ddot{\mathbf{q}} + \tau_{Grav}(\mathbf{q}) + \tau_{Fric}(\dot{\mathbf{q}}) + offset \tag{6-4}$$

The gravity torque of the second link and the friction term are, respectively, expressed by:

$$\tau_{Grav} = -gMXcos(\mathbf{q}) + gMYsin(\mathbf{q}) \tag{6-5}$$

$$\tau_{Fric}(\dot{\mathbf{q}}) = F_v\dot{\mathbf{q}} + \left(F_C + (F_S - F_C)e^{-\left(\frac{\|\dot{\mathbf{q}}\|}{v_s}\right)^{\delta\sigma}}\right) sign(\dot{\mathbf{q}}) \tag{6-6}$$

The estimation of the dynamic parameters of the TX40 robot, which considers a nonlinear friction model, will be performed by five algorithms. The specific parameters of each EA are set as suggested in the literature [126], they can be found in Table 6.16.

Table 6.16: Specific parameters of each EA and how they were tuned. They are: Scale Factor (F); Crossover constant (CR); Elite individuals per generation (N_{GA}); Crossover type (X_{over}); Mutation rate (ρ_c); Constriction coefficient (K_F); Maximum learning rates (μ_l).

DE	**GA**	**PSO**
Population size: 50	Population size: 50	Population size: 50
Generation limit: 100	Generation limit: 100	Generation limit: 100
/bin variation	N_{GA}: 2	K_F: 0.25
F (constant): 0.4	ρ_c: 0.02	Neighborhood size: 6
CR: 0.9	X_{over}: single point	μ_l: 2.5

The other algorithms used in this work are part of an open-source optimization library called Scikit-Optimize. The library has five minimization algorithms, two of which will be used in this work: *forest_minimize*, and *gp_minimize*. The former is a sequential optimization algorithm that uses decision trees and the latter is a Bayesian optimization algorithm that uses a Gaussian Process. They will be called here DT, and BO respectively. BO and DT algorithms start with 10 initial points. DT algorithm uses extra trees as its regressor. The cost function is called 100 times by BO and DT, which acquisition functions are the lower confidence bound (LCB) and negative probability of improvement (PI), respectively.

6.4.3 Results

Figure 6.11 shows the progress of the optimization process. All algorithms run 10 times. By the analyses of the curves in Figure 6.11 we notice that the tree EAs present the same behavior shown in the previous case study. DE and

PSO are characterized by a rapid improvement during the early searching stage followed by a steady-state condition. DE converges until the 30th generation and PSO until the 40th. GA is also characterized by a rapid improvement during the early searching stage; however, it can consistently improve its searching quality throughout the searching process, but in later stages, the value of the cost function decreases slowly. It seems that GA has converged until the 100th generation.

The cost function fluctuates in the initial stages and presents a sudden drop before 20 calls for BO and DT algorithms. DT algorithm converges up to 60 cost function calls and from there it presents only small fluctuations in its values. It is possible to observe that, for the BO algorithm, the cost function does not show any considerable fluctuation after 30 calls and that its value falls at a very low rate until reaching 100 function calls.

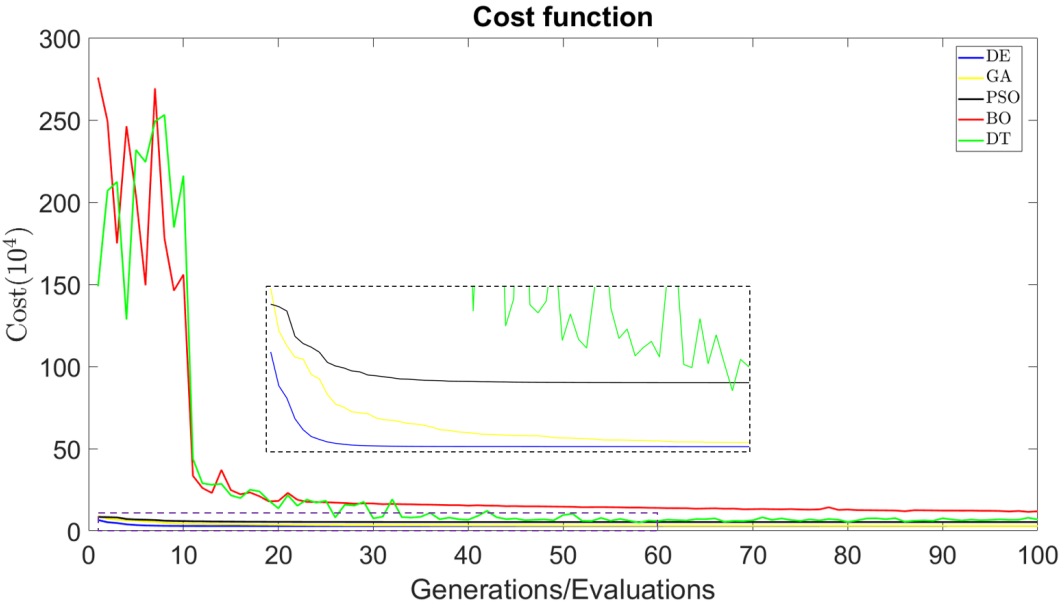

Figure 6.11: Minimum value of the cost function of the asymmetric friction models.

Figure 6.12 shows that the $dist_{mean}(G)$ values decrease at a faster rate during the early stage, which is related to the rapid improvement of the quality of the solution in Figure 6.11. GA has fluctuation in its $dist_{mean}(G)$, meaning that GA, compared to DE and PSO, maintains higher population diversity during the entire process.

Tables 6.17 and 6.18 show the dynamic parameters estimated by each algorithm, the total inertia, and gravity terms are following those found in [179], but the author has considered a linear friction model. Table 6.19 shows the average time spent by each algorithm. It is possible to notice that the DE algorithm demands, on average, lower computational cost than the other EAs

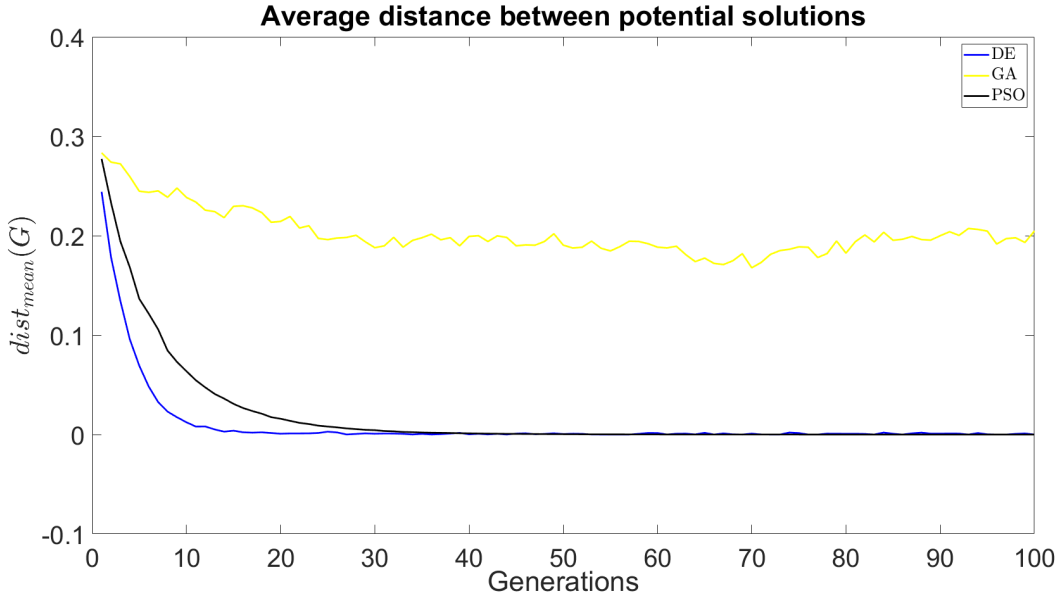

Figure 6.12: Mean population searching distance per generation $dist_{mean}(G)$.

in a total of 100 generations. BO is generally used in applications where the cost function is not known (black-box applications). In this case study, the BO algorithm is not a good choice since simpler algorithms can estimate the dynamic parameters of the system with lower computational cost presenting a lower relative error.

Table 6.17: Dynamic parameters of the TX40 robot estimated by the EAs.

Parameter	**DE**	**GA**	**PSO**
ZZ	1.5431 ± 0.099041	1.4625 ±0.11511	1.7759 ± 0.27671
MX	2.8351 ± 0.0045724	2.8305 ± 0.032262	2.7157 ± 0.084109
MY	0.011162 ± 0.040204	0.0087263 ± 0.06556	0.043832 ± 0.045052
F_v	6.1622 ± 0.43446	5.8579 ± 0.51648	7.6546 ± 1.3917
F_c	5.6851 ± 0.28933	5.8252 ± 0.30816	6.1318 ± 0.36578
F_s	9.2543 ± 1.4706	8.4545 ± 0.5738	8.0299 ± 0.89416
$Offset$	-0.069789 ± 0.29188	-0.059046 ± 0.41107	0.1021 ± 0.21679
v_s	0.043739 ± 0.018775	0.046754 ± 0.013601	0.057701 ± 0.015954
δ_σ	1.2634 ± 0.51774	1.7986 ± 0.68804	1.4888 ± 0.5117

The relative error of each algorithm was calculated after an OSA prediction, the results are in Table 6.20. The EAs have, on average, lower relative error when compared with the other algorithms. DE is the algorithm with the lowest relative error among all (7,2133%), it must be considered that DT presented a relative error (7,9176%) similar to those achieved by DE and GA, but with lower computational cost.

Figure 6.13 shows the results of the OSA prediction of the torque of the second joint. It is possible to notice that the algorithms can predict the torque in the second joint of the robot reasonably well. Through Equation (6-6) it is possible to reconstruct the shape of the friction in the second joint of the robot. Figure 6.14 shows the results of the reconstructed friction with the parameters

Table 6.18: Dynamic parameters of the TX40 robot estimated by BO and DT algorithms.

Parameter	BO	DT
ZZ	1.741209 ± 0.514135	1.638843 ± 0.453103
MX	2.716822 ± 0.186321	2.852588 ± 0.023148
MY	0.085575 ± 0.101936	0.038281 ± 0.077164
F_v	7.992918 ± 1.458288	5.995656 ± 1.027730
F_c	6.090332 ± 0.319006	5.911671 ± 0.511094
F_s	8.627831 ± 1.348596	8.718668 ± 1.386991
v_s	0.055201 ± 0.018365	0.051831 ± 0.022372
δ_σ	1.757256 ± 0.357630	1.598269 ± 0.832302
$Offset$	0.137157 ± 0.255402	0.237290 ± 0.517151

Table 6.19: Computational cost of each algorithm. 100 cost function evaluations were considered for the BO, and DT algorithms and 100 generations were used for the AEs.

Algorithm	Elapsed time (s)	Algorithm	Elapsed time (s)
DE	171.16 ± 3.3599	BO	61.439239 ± 1.309274
GA	174.91 ± 2.4036	DT	26.614644 ± 1.309274
PSO	179.58 ± 3.7509		

Table 6.20: Relative error of each algorithm.

Algorithm	Relative error (%)	Algorithm	Relative error (%)
DE	7.2133 ± 0.2903	BO	11.482 ± 3.7563
GA	7.4285 ± 0.44141	DT	7.9176 ± 0.73242
PSO	9.3014 ± 1.4671		

estimated by each algorithm. It is possible to notice that the reconstructed friction form is following the expected shape.

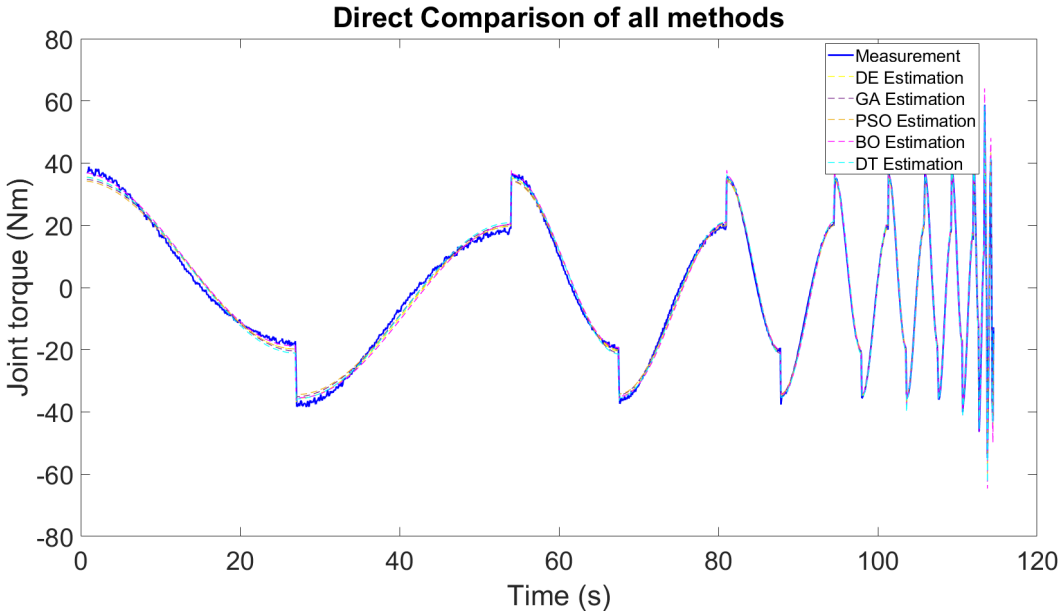

Figure 6.13: OSA prediction of the torque of the second joint.

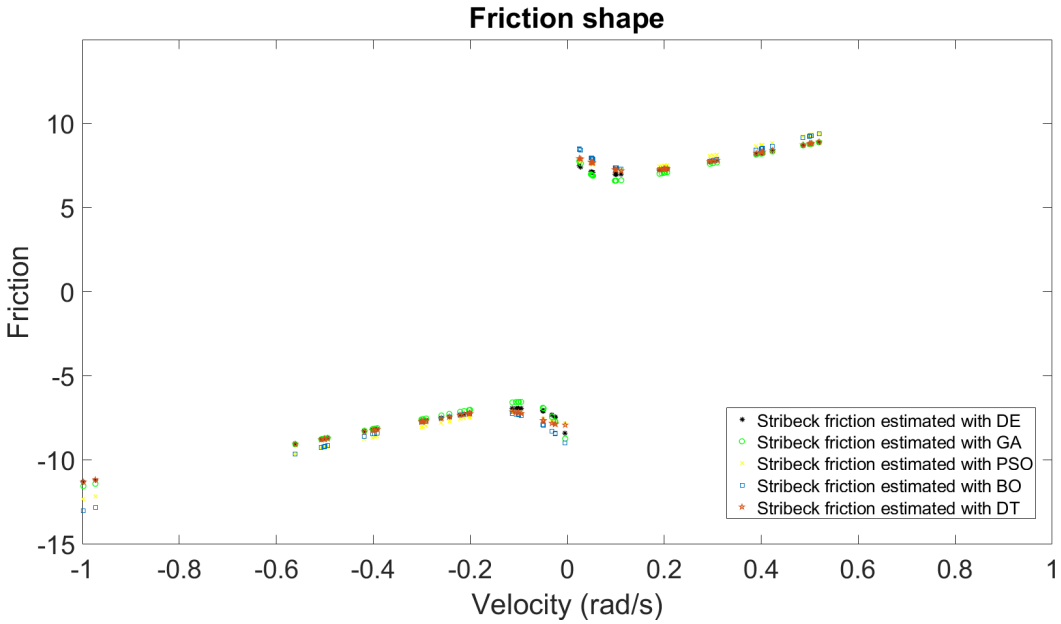

Figure 6.14: Friction estimated by all methods.

7
Conclusions

The general objective of this work was to develop different classes of models capable of accurately simulating the output variable of a system and to evaluate the efficiency of the optimization algorithms used in the parameter estimation task. Besides, this study aimed to assess which friction model would be the most appropriate to describe this phenomenon in a positioning system.

The results show that nonlinear friction models were more adequate to describe this phenomenon, which had an asymmetric behavior, in the positioning system. The simulations performed with parameters estimated by the evolutionary algorithms showed better results. The decision tree-based optimizer, used in the second case study, proved to be equally effective compared to evolutionary algorithms. Finally, the results show that combining models is an effective alternative to obtain more accurate simulation results.

In the first section of Chapter 6 different friction models have been considered in the task of estimating the dynamic parameters of the EMPS. The results have shown that a significant improvement can be achieved by using more complex friction models. According to the models adopted, the results suggest that the friction has an asymmetric behavior since terms like F_K^+ and F_K^-, F_V^+ and F_V^- are significantly different and the estimated mass has no significant variance when both symmetric and asymmetric models are compared. According to Figure 6.3(a), there is a high agreement between both measured and estimated forces, which is also evident when we see Figure 6.3(b). The best model has low standard deviation and relative error, lower than 2% and 4%, respectively. It has also an R^2 almost equal to unity.

In Section 6.2 of Chapter 6 grey and black-box models were used to estimate the position of the joint of the EMPS. According to the results, a significant improvement can be achieved by modeling the simulation error. An improvement higher than 20% was achieved by combining the modeled error with a grey-box model. The GBM approach has the lowest performance among all approaches, it is due to the influence of the number of parameters being estimated and the initial conditions. Because of this and since there is no significant difference between the relative errors of the friction models of the GBM approach, it is not possible to conclude whether asymmetric or

symmetric friction models are better to describe the behavior of the friction in the EMPS. The ensemble of $\widehat{\mathbf{y}}_{\mathbf{1}}$ and $\widehat{\mathbf{y}}_{\mathbf{2}}$ has also proven to be effective, achieved almost 20% of relative error improvement.

In the third section of Chapter 6, the performance of five metaheuristic algorithms used to estimate the dynamic parameters of the EMPS were compared, in addition to investigating which friction model is the most suitable to describe the friction behavior of the case study. The results of the simulations performed with the parameters estimated by the EAs were compared with the results of other studies.

The results have shown that DE and PSO have different behavior compared to ACO, BBO, and GA. The main difference lies in the fact that the second group of EAs can consistently improve their searching quality throughout the searching process, while the first one rapidly achieves a steady-state condition. ACO has the fastest convergence speed in the initial searching stage, but DE and PSO converge quickly during the middle and later stages. GA has the highest $dist_{mean}(G)$ and fluctuations amplitudes, meaning that it maintains a high population diversity in every search stage.

The estimated coefficients of the asymmetric friction models indicate asymmetric behavior of the friction since F_V^+ and F_V^- are significantly different for all friction models. Some tests of the asymmetric and symmetric Coulomb model performed with the estimated parameters were able to slightly outperform the LS-IDIM approach. However, the suggested friction models were able to bring higher relative error improvement, especially F_T^a and F_T^s friction models. These models smooth the discontinuity at zero velocity of Coulomb's friction model by an exponential term in their formulation. A non-parametric multiple comparison procedure among the EAs has stated that there is no significant difference among the EAs in terms of relative error improvement for the number of function evaluations tested. However, the PSO algorithm was able to converge faster than the others, as shown in Figure 6.9. Regarding the friction models, F_T^s and F_V^a can be considered significantly different from each other for the EMPS case study.

The last section of Chapter 6 addresses the second case study, five optimization algorithms were chosen to identify the dynamic parameters of a TX40 robot. The AEs had, on average, superior performance compared to the other algorithms when the relative error was considered. However, the optimization algorithm using decision trees achieved similar performance compared to DE and GA algorithms and superior to PSO. Among the EAs, DE presented the best performance, with the lowest computational cost and the lowest relative error. Among the other algorithms, BO has the highest

computational cost and relative error. DT, DE, and GA were the optimizers that presented the lowest relative errors and also the lowest computational cost.

The dynamic parameters estimated by the algorithms were used to predict the torque of the second joint and the shape of the friction torque. The results of the OSA prediction show that with the estimated parameters it was possible to predict the torque in the second joint with reasonable precision and that the friction shape is consistent with the expected.

7.1 Final Comments

It is important to test more case studies to verify more precisely the benefits of one optimization algorithm over another. More choices of the settings of the hyperparameters for the comparison of the EAs should be also considered.

7.2 Future Work

Recently, Physics-Informed neural networks have been used successfully to solve inverse problems involving differential equations [187–189]. This motivates the author to use this approach in future system identification works.

Bibliography

[1] MA, G.; WANG, Z. ; WANG, F.. **Friction measurement, identification, and compensation for servomechanisms**. In: Chen, R., editor, 2011 INTERNATIONAL CONFERENCE IN ELECTRICS, COMMUNICATION AND AUTOMATIC CONTROL PROCEEDINGS, p. 1469–1475, New York, NY, 2012. Springer New York.

[2] ISERMANN, R.. **Examples for the Design of Mechatronic Systems: Modeling, Control and Diagnosis**, p. 527–586. Springer London, London, 2005.

[3] MERZOUKI, R.; SAMANTARAY, A. K.; PATHAK, P. M. ; BOUAMAMA, B. O.. **Intelligent Transportation Systems**, p. 769–867. Springer London, London, 2013.

[4] LJUNG, L.; GLAD, T.. **Modeling of dynamic systems**. Prentice Hall, Upper Saddle River, NJ,USA, 2nd edition, 1994.

[5] KULAKOWSKI, B. T.; GARDNER, J. F. ; SHEARER, J. L.. **Dynamic Modeling and Control of Engineering Systems**. Cambridge University Press, New York,USA, 3rd edition, 2007.

[6] CHANG, W.-D.. **An improved real-coded genetic algorithm for parameters estimation of nonlinear systems**. Mechanical Systems and Signal Processing, 20(1):236 – 246, 2006.

[7] TANGIRALA, A. K.. **Principles of system identification:Theory and practice**. CRC Press, Boca Raton, FL, USA, 2015.

[8] KERSCHEN, G.; WORDEN, K.; VAKAKIS, A. F. ; GOLINVAL, J.-C.. **Past, present and future of nonlinear system identification in structural dynamics**. Mechanical Systems and Signal Processing, 20:505–592, 2006.

[9] LJUNG, L.. **System Identification - Theory for the user**. Prentice Hall, Upper Saddle River, NJ,USA, 2nd edition, 1999.

[10] LJUNG, L.. **Perspectives on system identification**. Annual Reviews in Control, 34(1):1 – 12, 2010.

[11] KEESMAN, K. J.. **System Identification: An Introduction**. Springer-Verlag London, London, UK, 1st edition, 2011.

[12] SJÖBERG, J.; ZHANG, Q.; LJUNG, L.; BENVENISTE, A.; DELYON, B.; GLORENNEC, P.-Y.; HJALMARSSONT, H. ; JUDITSKY, A.. **Nonlinear black-box modeling in system identification**. Automatica, 31:1691–1724, 1995.

[13] BOHLIN, T.. **A case study of grey box identification**. Automatica, 30:2:307–318, 1994.

[14] KRISTINSSON, K.; DUMONT, G. A.. **System identification and control using genetic algorithms**. IEEE Transactions on Systems, Man, and Cybernetics, 22(5):1033–1046, 1992.

[15] MISHRA, K. G. S. K.. **Nonlinear system identification using clonal particle swarm optimization-based functional link artificial neural network**. In: Sethi, I. K., editor, COMPUTATIONAL VISION AND ROBOTICS, p. 89–96, New Delhi, 2015. Springer India.

[16] JALANKO, M.; SANCHEZ, Y.; MAHALEC, V. ; MHASKAR, P.. **Adaptive system identification of industrial ethylene splitter: A comparison of subspace identification and artificial neural networks**. Computers and Chemical Engineering, 147:107240, 2021.

[17] BAGHERZADEH, S. A.. **Nonlinear aircraft system identification using artificial neural networks enhanced by empirical mode decomposition**. Aerospace Science and Technology, 75:155–171, 2018.

[18] RAHMOUNE, M. B.; HAFAIFA, A.; KOUZOU, A.; CHEN, X. ; CHAIBET, A.. **Gas turbine monitoring using neural network dynamic nonlinear autoregressive with external exogenous input modelling**. Mathematics and Computers in Simulation, 179:23–47, 2021.

[19] AYALA, H. V. H.; HABINEZA, D.; RAKOTONDRABE, M. ; DOS SANTOS COELHO, L.. **Nonlinear black-box system identification through coevolutionary algorithms and radial basis function artificial neural networks**. Applied Soft Computing, 87:105990, 2020.

[20] CHARALAMPAKIS, A.; KOUMOUSIS, V.. **Identification of bouc–wen hysteretic systems by a hybrid evolutionary algorithm**. Journal of Sound and Vibration, 314(3):571–585, 2008.

[21] ABBAS, H. M.; BAYOUMI, M. M.. **An adaptive evolutionary algorithm for volterra system identification**. Pattern Recognition Letters, 26(1):109–119, 2005.

[22] DEVIA, W.; AGBOSSOU, K. ; CARDENAS, A.. **An evolutionary approach to modeling and control of space heating and thermal storage systems**. Energy and Buildings, 234:110674, 2021.

[23] DE ASSIS, L. S.; DE P. JUNIOR, J. R.; TARRATACA, L. ; HADDAD, D. B.. **Efficient volterra systems identification using hierarchical genetic algorithms**. Applied Soft Computing, 85:105745, 2019.

[24] GOTMARE, A.; BHATTACHARJEE, S. S.; PATIDAR, R. ; GEORGE, N. V.. **Swarm and evolutionary computing algorithms for system identification and filter design: A comprehensive review**. Swarm and Evolutionary Computation, 32:68–84, 2017.

[25] XIN ZHENG, Y.; LIAO, Y.. **Parameter identification of nonlinear dynamic systems using an improved particle swarm optimization**. Optik, 127(19):7865–7874, 2016.

[26] WANG, Z.; LUO, X.. **Modeling study of nonlinear dynamic soft sensors and robust parameter identification using swarm intelligent optimization cs-nlj**. Journal of Process Control, 58:33–45, 2017.

[27] YOUSRI, D.; HASANIEN, H. M. ; FATHY, A.. **Parameters identification of solid oxide fuel cell for static and dynamic simulation using comprehensive learning dynamic multi-swarm marine predators algorithm**. Energy Conversion and Management, 228:113692, 2021.

[28] BIAN, Q.; ZHAO, K.; WANG, X. ; XIE, R.. **System identification method for small unmanned helicopter based on improved particle swarm optimization**. Journal of Bionic Engineering, 13(3):504–514, 2016.

[29] MAJHI, B.; PANDA, G.. **Development of efficient identification scheme for nonlinear dynamic systems using swarm intelligence techniques**. Expert Systems with Applications, 37(1):556–566, 2010.

[30] DUPONT, P. E.. **Friction modeling in dynamic robot simulation**. Proceedings of IEEE International Conference on Robotics and Automation, p. 1370–1376, 1990. Cincinnati, OH, USA.

[31] IURIAN, C.; IKHOUANE, F.; RODELLAR, J. ; GRINO, R.. **Identification of a system with dry friction.** Technical Report, Universitat Politecnica de Catalunya, 2005.

[32] O.M.BRASTEIN; D.W.U.PERERA; C.PFEIFER ; N.-O.SKEIE. **Parameter estimation for grey-box models of building thermal behaviour**. Energy and Buildings, 169:15:58–68, 2018.

[33] KICSINY, R.. **Grey-box model for pipe temperature based on linear regression**. International Journal of Heat and Mass Transfer, 107:13–20, 2017.

[34] AFRAM, A.; JANABI-SHARIFI, F.. **Black-box modeling of residential HVAC system and comparison of gray-box and black-box modeling methods**. Energy and Buildings, 94:121–149, 2015.

[35] M.BIDARVATAN; V.THAKKAR; M.SHAHBAKHTI; B.BAHRI ; AZIZ, A.. **Grey-box modeling of HCCI engines**. Applied Thermal Engineering, 70:1:397–409, 2014.

[36] WERNHOLT, E.; MOBERG, S.. **Nonlinear gray-box identification using local models applied to industrial robots**. Automatica, 47:4:650–660, 2011.

[37] YUAN, H.-B.; NA, H.-C. ; KIM, Y.-B.. **Robust MPC–PIC force control for an electro-hydraulic servo system with pure compressive elastic load**. Control Engineering Practice, 79:170–184, 2018.

[38] TAGHBALOUT, M.; ANTOINE, J. F. ; ABBA, G.. **Experimental dynamic identification of a yumi collaborative robot**. IFAC-PapersOnLine, 52(13):1168–1173, 2019. 9th IFAC Conference on Manufacturing Modelling, Management and Control MIM 2019, Berlin.

[39] BRUNOT, M.; JANOT, A. ; CARRILLO, F.. **An automated instrumental variable method for rigid industrial robot identification.** IFAC-PapersOnLine, 51:431–436, 2018. Stockholm, Sweden.

[40] BAHLOUL, A.; TLIBA, S. ; CHITOUR, Y.. **Dynamic parameters identification of an industrial robot with and without payload**. IFAC-PapersOnLine, 51(15):443–448, 2018. 18th IFAC Symposium on System Identification SYSID 2018.

[41] JUBIEN, A.; GAUTIER, M. ; JANOT, A.. **Dynamic identification of the Kuka LWR robot using motor torques and joint torque**

sensors data. IFAC Proceedings Volumes, 47(3):8391–8396, 2014. 19th IFAC World Congress, Cape Town.

[42] BRUNOT, M.; JANOT, A.. **A new recursive instrumental variables approach for robot identification**. IFAC-PapersOnLine, 51(15):132–137, 2018. 18th IFAC Symposium on System Identification SYSID 2018.

[43] BRUNOT, M.; JANOT, A. ; CARRILLO, F.. **State space estimation method for the identification of an industrial robot arm**. IFAC-PapersOnLine, 50(1):9815–9820, 2017. 20th IFAC World Congress, Toulouse.

[44] IRIGOYEN, E.; MINANO, G.. **A NARX neural network model for enhancing cardiovascular rehabilitation therapies**. Neurocomputing, 109:9–15, 2013.

[45] WUNSCH, A.; LIESCH, T. ; BRODA, S.. **Forecasting groundwater levels using nonlinear autoregressive networks with exogenous input (NARX)**. Journal of Hydrology, 567:743–758, 2018.

[46] SAMARA, P. A.; SAKELLARIOU, J. S.; FOUSKITAKIS, G. N.; HIOS, J. D. ; FASSOIS, S. D.. **Aircraft virtual sensor design via a time-dependent functional pooling NARX methodology**. Aerospace Science and Technology, 29:114–124, 2013.

[47] TIJANI, I. B.; AKMELIAWATI, R.; LEGOWO, A. ; BUDIYONO, A.. **Nonlinear identification of a small scale unmanned helicopter using optimized NARX network with multiobjective differential evolution**. Engineering Applications of Artificial Intelligence, 33:99–115, 2014.

[48] WORDEN, K.; WONG, C.; PARLITZ, U.; HORNSTEIN, A.; ENGSTER, D.; TJAHJOWIDODO, T.; AL-BENDER, F.; RIZOS, D. ; FASSOIS, S.. **Identification of pre-sliding and sliding friction dynamics: Grey box and black-box models**. Mechanical Systems and Signal Processing, 21(1):514 – 534, 2007.

[49] YANG, X.-S.. **Introduction to Mathematical Optimization: From Linear Programming to Metaheuristics**. Cambridge International Science Publishing, UK, 2008.

[50] BOYD, S.; VANDENBERGHE, L.. **Convex Optimization**. Cambridge University Press, USA, 2004.

[51] AYALA, H. V. H.; DOS SANTOS COELHO, L.. **Cascaded evolutionary algorithm for nonlinear system identification based on correlation functions and radial basis functions neural networks**. Mechanical Systems and Signal Processing, 68-69:378 – 393, 2016.

[52] MERUANE, V.; HEYLEN, W.. **An hybrid real genetic algorithm to detect structural damage using modal properties**. Mechanical Systems and Signal Processing, 25(5):1559 – 1573, 2011.

[53] WORDEN, K.; MANSON, G.. **On the identification of hysteretic systems. Part I: Fitness landscapes and evolutionary identification**. Mechanical Systems and Signal Processing, 29:201 – 212, 2012.

[54] PERERA, R.; RUIZ, A.. **A multistage fe updating procedure for damage identification in large-scale structures based on multiobjective evolutionary optimization**. Mechanical Systems and Signal Processing, 22(4):970 – 991, 2008.

[55] AGUIRRE, L. A.; BARBOSA, B. H. ; BRAGA, A. P.. **Prediction and simulation errors in parameter estimation for nonlinear systems**. Mechanical Systems and Signal Processing, 24(8):2855 – 2867, 2010.

[56] CHEN, Q.; WORDEN, K.; PENG, P. ; LEUNG, A.. **Genetic algorithm with an improved fitness function for (N)ARX modelling**. Mechanical Systems and Signal Processing, 21(2):994 – 1007, 2007.

[57] LEI, Y.; HAN, D.; LIN, J. ; HE, Z.. **Planetary gearbox fault diagnosis using an adaptive stochastic resonance method**. Mechanical Systems and Signal Processing, 38(1):113 – 124, 2013.

[58] SITARZ, P.; POWAŁKA, B.. **Dual ant colony operational modal analysis parameter estimation method**. Mechanical Systems and Signal Processing, 98:231 – 267, 2018.

[59] BARALDI, P.; BONFANTI, G. ; ZIO, E.. **Differential evolution-based multi-objective optimization for the definition of a health indicator for fault diagnostics and prognostics**. Mechanical Systems and Signal Processing, 102:382 – 400, 2018.

[60] TANG, H.; XUE, S. ; FAN, C.. **Differential evolution strategy for structural system identification**. Computers and Structures, 86(21):2004–2012, 2008.

[61] FENG, H.; YIN, C.-B.; WEN WENG, W.; MA, W.; JING ZHOU, J.; HUA JIA, W. ; LI ZHANG, Z.. **Robotic excavator trajectory control using an improved GA based PID controller**. Mechanical Systems and Signal Processing, 105:153 – 168, 2018.

[62] LIANG, H.; ZOU, J.; ZUO, K. ; KHAN, M. J.. **An improved genetic algorithm optimization fuzzy controller applied to the wellhead back pressure control system**. Mechanical Systems and Signal Processing, 142:106708, 2020.

[63] KHEHRA, B. S.; PHARWAHA, A. P. S.. **Comparison of genetic algorithm, particle swarm optimization and biogeography-based optimization for feature selection to classify clusters of microcalcifications**. Journal of The Institution of Engineers Ser. B, 98:189–202, 2017.

[64] YANG, G.; ZHANG, Y.; YANG, J.; JI, G.; DONG, Z.; SHUIHUA WANG, C. F. ; WANG, Q.. **Automated classification of brain images using wavelet-energy and biogeography-based optimization**. Multimedia Tools and Applications, 75:15601–15617, 2016.

[65] GUO, W.; CHEN, M.; WANG, L. ; WU, Q.. **Backtracking biogeography-based optimization for numerical optimization and mechanical design problems**. Applied Intelligence volume, 44:894–903, 2016.

[66] HADIDI, A.; NAZARI, A.. **Design and economic optimization of shell-and-tube heat exchangers using biogeography-based (BBO) algorithm**. Applied Thermal Engineering, 51(1):1263–1272, 2013.

[67] MO, H.; XU, L.. **Research of biogeography particle swarm optimization for robot path planning**. Neurocomputing, 148:91–99, 2015.

[68] ZHENG, Y.; LU, X.; ZHANG, M. ; CHEN, S.. **Biogeography-Based Optimization: Algorithms and Applications**. Springer Singapore, Beijing, 1st edition, 2019.

[69] ZHAO, F.; QIN, S.; ZHANG, Y.; MA, W.; ZHANG, C. ; SONG, H.. **A hybrid biogeography-based optimization with variable neighborhood search mechanism for no-wait flow shop scheduling problem**. Expert Systems with Applications, 126:321–339, 2019.

[70] RIFAI, A. P.; NGUYEN, H.-T.; AOYAMA, H.; DAWAL, S. Z. M. ; MASRUROH, N. A.. **Non-dominated sorting biogeography-based optimization for bi-objective reentrant flexible manufacturing system scheduling**. Applied Soft Computing, 62:187–202, 2018.

[71] MA, H.; SU, S.; SIMON, D. ; FEI, M.. **Ensemble multi-objective biogeography-based optimization with application to automated warehouse scheduling**. Engineering Applications of Artificial Intelligence, 44:79–90, 2015.

[72] KIM, S.-S.; BYEON, J.-H.; YU, H. ; LIU, H.. **Biogeography-based optimization for optimal job scheduling in cloud computing**. Applied Mathematics and Computation, 247:266–280, 2014.

[73] LIN, J.; ZHANG, S.. **An effective hybrid biogeography-based optimization algorithm for the distributed assembly permutation flow-shop scheduling problem**. Computers and Industrial Engineering, 97:128–136, 2016.

[74] NIU, Q.; ZHANG, L. ; LI, K.. **A biogeography-based optimization algorithm with mutation strategies for model parameter estimation of solar and fuel cells**. Energy Conversion and Management, 86:1173–1185, 2014.

[75] GOLLEE, C.; MAJSCHAK, J.-P.. **A parameter identification case-study for a dynamical mechanical system using frequency response analysis and a particle swarm algorithm for trajectory optimization**. Engineering Science and Technology, an International Journal, 23(4):769–780, 2020.

[76] ALFI, A.; MODARES, H.. **System identification and control using adaptive particle swarm optimization**. Applied Mathematical Modelling, 35(3):1210–1221, 2011.

[77] SALEEM, A.; TAHA, B.; TUTUNJI, T. ; AL-QAISIA, A.. **Identification and cascade control of servo-pneumatic system using particle swarm optimization**. Simulation Modelling Practice and Theory, 52:164–179, 2015.

[78] CHARALAMPAKIS, A.; DIMOU, C.. **Identification of Bouc–Wen hysteretic systems using particle swarm optimization**. Computers and Structures, 88(21):1197–1205, 2010.

[79] JIANG, P.; ZHOU, Q. ; SHAO, X.. **Surrogate Model-Based Engineering Design and Optimization**. Springer Singapore, Singapore, 1st edition, 2020.

[80] JOY, T. T.; RANA, S.; GUPTA, S. ; VENKATESH, S.. **Fast hyperparameter tuning using Bayesian optimization with directional derivatives**. Knowledge-Based Systems, 205:106247, 2020.

[81] ROWOLD, M.; WISCHNEWSKI, A. ; LOHMANN, B.. **Constrained Bayesian optimization of a linear feed-forward controller**. IFAC-PapersOnLine, 52(29):1 – 6, 2019. 13th IFAC Workshop on Adaptive and Learning Control Systems ALCOS 2019, Winchester, UK.

[82] ZHANG, Q.; HU, W.; LIU, Z. ; TAN, J.. **TBM performance prediction with Bayesian optimization and automated machine learning**. Tunnelling and Underground Space Technology, 103:103493, 2020.

[83] CHENG, H.; DING, X.; ZHOU, W. ; DING, R.. **A hybrid electricity price forecasting model with Bayesian optimization for German energy exchange**. International Journal of Electrical Power and Energy Systems, 110:653 – 666, 2019.

[84] SHIN, S.; LEE, Y.; KIM, M.; PARK, J.; LEE, S. ; MIN, K.. **Deep neural network model with Bayesian hyperparameter optimization for prediction of NOx at transient conditions in a diesel engine**. Engineering Applications of Artificial Intelligence, 94:103761, 2020.

[85] MAIER, M.; ZWICKER, R.; AKBARI, M.; RUPENYAN, A. ; WEGENER, K.. **Bayesian optimization for autonomous process set-up in turning**. CIRP Journal of Manufacturing Science and Technology, 26:81 – 87, 2019.

[86] MAIER, M.; RUPENYAN, A.; AKBARI, M.; ZWICKER, R. ; WEGENER, K.. **Turning: Autonomous process set-up through Bayesian optimization and Gaussian process models**. Procedia CIRP, 88:306 – 311, 2020. 13th CIRP Conference on Intelligent Computation in Manufacturing Engineering, 17-19 July 2019, Gulf of Naples, Italy.

[87] J., Y.; K., D.. **Bayesian system identification of molecular cascades**. Procedia CIRP, 4984, 2008.

[88] BAHERI, A.; BIN-KARIM, S.; BAFANDEH, A. ; VERMILLION, C.. **Real-time control using Bayesian optimization: A case study in**

airborne wind energy systems. Control Engineering Practice, 69:131 – 140, 2017.

[89] KANG, Y.; KRARTI, M.. **Bayesian-emulator based parameter identification for calibrating energy models for existing buildings**. Building Simulation, 9:411–428, 2016.

[90] XUE, Y.; LIU, Y.; JI, C. ; XUE, G.. **Hydrodynamic parameter identification for ship manoeuvring mathematical models using a bayesian approach**. Ocean Engineering, 195:106612, 2020.

[91] VILLEMONTEIX, J.; VAZQUEZ, E. ; WALTER, E.. **Bayesian optimization for parameter identification on a small simulation budget**. IFAC Proceedings Volumes, 42(10):1603–1608, 2009. 15th IFAC Symposium on System Identification, Saint-Malo, France.

[92] FALCO, A. D.; GIRARDI, M.; PELLEGRINI, D.; ROBOL, L. ; SEVIERI, G.. **Model parameter estimation using Bayesian and deterministic approaches: the case study of the Maddalena Bridge**. Procedia Structural Integrity, 11:210 – 217, 2018.

[93] SMARRA, F.; DI GIROLAMO, G. D.; DE IULIIS, V.; JAIN, A.; MANGHARAM, R. ; D'INNOCENZO, A.. **Data-driven switching modeling for MPC using regression trees and random forests**. Nonlinear Analysis: Hybrid Systems, 36:100882, 2020.

[94] QIU, Y.. **Forecasting the consumer confidence index with tree-based MIDAS regressions**. Economic Modelling, 91:247–256, 2020.

[95] GHIASI, M. M.; ZENDEHBOUDI, S.. **Application of decision tree-based ensemble learning in the classification of breast cancer**. Computers in Biology and Medicine, 128:104089, 2021.

[96] DIEZ, M.. **Porosity optimization in nanoporous materials via machine learning**. 2020. Master Thesis, Politecnico de Torino.

[97] AKBARI, E.; MORADI, R.; AFROOZEH, A.; ALIZADEH, A. ; NILASHI, M.. **A new approach for prediction of graphene based ISFET using regression tree and neural network**. Superlattices and Microstructures, 130:241 – 248, 2019.

[98] POLIKAR, R.. **Ensemble Machine Learning**. Springer, NY, USA, 2012.

[99] PAPADOPOULOS, S.; KARAKATSANIS, I.. **Short-term electricity load forecasting using time series and ensemble learning methods**. Proceedings of IEEE Power and Energy Conference, Champaign, IL, USA, p. 1–6, 2015.

[100] QIUA, X.; REN, Y.; SUGANTHAN, P. N. ; AMARATUNGA, G. A. J.. **Empirical mode decomposition based ensemble deep learning for load demand time series forecasting**. Journal of Applied Soft Computing, 54:246–255, 2017.

[101] M.SAVIOZZI; S.MASSUCCO ; F.SILVESTRO. **Implementation of advanced functionalities for distribution management systems: Load forecasting and modeling through artificial neural networks ensembles**. Journal of Electric Power Systems Research, 167:230–239, 2017.

[102] SPERATI, S.; ALESSANDRINI, S. ; MONACHE, L. D.. **An application of the ECMWF ensemble prediction system for short-term solar power forecasting**. Journal of Solar Energy, 133:437–450, 2016.

[103] RAZA, M. Q.; N.MITHULANANTHAN ; SUMMERFIELD, A.. **Solar output power forecast using an ensemble framework with neural predictors and Bayesian adaptive combination**. Journal of Solar Energy, 166:226–241, 2018.

[104] NOGOSEKE, L. F.; ANDRADE, G. H. B. ; BOARETTO, M. A. R.. **Investigating the use of extremely randomized trees, gradient boosting machine, K-nearest neighbors and their ensemble applied to fault detection**. Proceedings of the 24th ABCM International Congress of Mechanical Engineering, Curitiba, Brazil, 2017.

[105] WONG, C.; WORDEN, K.. **Generalised NARX shunting neural network modelling of friction**. Mechanical Systems and Signal Processing, 21:553–572, 2007.

[106] ZHANG, G. P.. **Time series forecasting using a hybrid ARIMA and neural network model**. Neurocomputing, 50:159–175, 2003.

[107] ENGELBRECHT, A. P.. **Computational Intelligence: An Introduction**. Wiley, England, 2nd edition, 2007.

[108] GOLDBERG, D. E.. **Genetic Algorithms in Search, Optimization, and Machine Learning**. Addison-Wesley Professional, Boston, MA, United States, 1st edition, 1989.

[109] LJUNG, L.. **Perspectives on system identification**. IFAC Proceedings Volumes, 41(2):7172 – 7184, 2010. 17th IFAC World Congress, Seoul, South Korea.

[110] BOHLIN, T.. **Interactive System Identification: Prospects and Pitfalls**. Springer-Verlag Berlin Heidelberg, Heidelberg, Germany, 1991.

[111] KALABA, R.; SPINGARN, K.. **Control, Identification, and Input Optimization**. Springer US, New York, USA, 1982.

[112] NARENDRA, K. S.; PARTHASARATHY, K.. **Identification and control of dynamical systems using neural networks**. IEEE Transactions on Neural Networks, 1(1):4–27, 1990.

[113] HAYKIN, S.. **Neural Networks and Learning Machines**. Prentice Hall, Upper Saddle River, NJ,USA, 3rd edition, 2009.

[114] BILLINGS, S. A.. **Nonlinear System Identification: NARMAX Methods in the Time, Frequency, and Spatio-Temporal Domains**. John Wiley and Sons, New York, 1st edition, 2013.

[115] X., Z.; R., W.; T., Z.; L., W.; Y., L. ; Y., Z.. **Short-term load forecasting based on RBM and NARX neural network**. In: DS., H.; M., G.; K., H. ; A., H., editors, INTELLIGENT COMPUTING METHODOLOGIES:ICIC 2018. LECTURE NOTES IN COMPUTER SCIENCE, volumen 10956, p. 193–203. 2018.

[116] BI, W.; XU, Y. ; WANG, H.. **Comparison of searching behaviour of three evolutionary algorithms applied to water distribution system design optimization**. Water, 12:695, 2020.

[117] ROKACH, L.. **Ensemble-based classifiers**. Artificial Intelligence Review, 33:1–39, 2010.

[118] RIBEIRO, M. H. D. M.; DOS SANTOS COELHO, L.. **Ensemble approach based on bagging, boosting and stacking for short-term prediction in agribusiness time series**. Applied Soft Computing, 86:105837, 2020.

[119] RHINEHART, R. R.. **Engineering Optimization Applications, Methods and Analysis**. ASME Press and John Wiley and Sons Ltd, New York, 1st edition, 2018.

[120] KUMAR, K.; ROY, S. ; DAVIM, J. P.. **Soft Computing Techniques for Engineering Optimization**. CRC Press, Boca Raton, 1st edition, 2019.

[121] BELEGUNDU, A. D.; CHANDRUPATLA, T. R.. **Optimization Concepts and Applications in Engineering**. Cambridge University Press, 3rd edition, 2019.

[122] SIOSHANSI, R.; CONEJO, A. J.. **Optimization in Engineering Models and Algorithms**. Springer International Publishing, 1st edition, 2017.

[123] RAO, S. S.. **Engineering Optimization Theory and Practice**. John Wiley and Sons Ltd, New York, 5th edition, 2020.

[124] YANG, X.-S.. **Engineering Optimization: An Introduction with Metaheuristic Applications**. John Wiley and Sons, New York, 1st edition, 2010.

[125] WAHDE, M.. **Biologically Inspired Optimization Methods**. WIT-PRESS, 1st edition, 2008.

[126] SIMON, D.. **Evolutionary Optimization Algorithms**. John Wiley and Sons, 1st edition, 2013.

[127] AHN, C. W.. **Advances in Evolutionary Algorithms: Theory, Design and Practice**. Springer-Verlag Berlin Heidelberg, Berlin, 1st edition, 2006.

[128] CÂMARA, D.. **Bio-inspired Networking**. ISTE Press Ltd, London, 1st edition, 2015.

[129] BAECK, T.; FOGEL, D. ; MICHALEWICZ, Z.. **Evolutionary Computation 2: Advanced Algorithms And Operators**. Taylor and Francis US, NY, 1st edition, 2000.

[130] HOLLAND, J. H.. **Adaptation in Natural and Artificial Systems**. The University of Michigan Press, Ann Arbor, USA, 1st edition, 1975.

[131] MITCHELL, M.. **An Introduction to Genetic Algorithms**. Bradford Book, Cambridge, MA , USA, 1st edition, 1998.

[132] MARTÍ, R.; PARDALOS, P. M. ; RESENDE, M. G.. **Handbook of Heuristics**. Springer International Publishing, Switzerland, 1st edition, 2018.

[133] DORIGO, M.; MANIEZZO, V. ; COLORNI, A.. **Ant system: optimization by a colony of cooperating agents**. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 26(1):29–41, 1996.

[134] DORIGO, M.; MANIEZZO, V. ; COLORNI, A.. **Positive feedback as a search strategy**. Technical Report No. 91-016, Politecnico di Milano, Italy, 1991.

[135] ERIC BONABEAU, MARCO DORIGO, G. T.. **Swarm Intelligence: From Natural to Artificial Systems**. Oxford University Press, USA, 1st edition, 1999.

[136] SIMON, D.. **Biogeography-based optimization**. Trans. Evol. Comp, 12(6):702–713, dec 2008.

[137] MACARTHUR, R. H.; WILSON, E. O.. **The theory of biogeography**. Princeton University Press, Princeton, 1st edition, 1967.

[138] SIMON, D.. **A probabilistic analysis of a simplified biogeography-based optimization algorithm**. Evolutionary Computation, 19(2):167–188, 2011.

[139] STORN, R.; PRICE, K.. **Differential evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces**. International Computer Science Institute - Tech. Rep. TR-95-012 [Online], 1995.

[140] MALLIPEDDI, R.; SUGANTHAN, P.; PAN, Q. ; TASGETIREN, M.. **Differential evolution algorithm with ensemble of parameters and mutation strategies**. Applied Soft Computing, 11(2):1679 – 1696, 2011.

[141] DAS, S.; SUGANTHAN, P. N.. **Differential evolution: A survey of the state-of-the-art**. IEEE Transactions on Evolutionary Computation, 15(1):4–31, 2011.

[142] YANG, X.-S.; CHIEN, S. F. ; TING, T.. **Bio-Inspired Computation in Telecommunications**. Morgan Kaufmann, Waltham, MA, USA, 1st edition, 2015.

[143] KENNEDY, J.; EBERHART, R.. **Particle swarm optimization**. Proceedings of ICNN'95 - International Conference on Neural Networks, Perth, Australia, 4:1942–1948, 1995.

[144] BLUM, C.; MERKLE, D.. **Swarm intelligence: Introduction and applications**. In: SWARM INTELLIGENCE, Springer-Verlag Berlin Heidelberg, 2008.

[145] EBERHART, R.; SHI, Y. ; KENNEDY, J.. **Swarm Intelligence**. Morgan Kaufmann, USA, 1st edition, 2001.

[146] DAVIS, S. E.; CREMASCHI, S. ; EDEN, M. R.. **Efficient surrogate model development: Impact of sample size and underlying model dimensions**. In: Eden, M. R.; Ierapetritou, M. G. ; Towler, G. P., editors, 13TH INTERNATIONAL SYMPOSIUM ON PROCESS SYSTEMS ENGINEERING (PSE 2018), volumen 44 de **Computer Aided Chemical Engineering**, p. 979 – 984. Elsevier, 2018.

[147] KIM, S. H.; BOUKOUVALA, F.. **Machine learning-based surrogate modeling for data-driven optimization: a comparison of subset selection for regression techniques**. Optimization Letters, 14:989 – 1010, 2020.

[148] TULLEKEN, H. J.. **Grey-box modelling and identification using physical knowledge and Bayesian techniques**. Automatica, 29(2):285 – 308, 1993.

[149] BOX, G. E.; TIAO, G. C.. **Bayesian Inference in Statistical Analysis**. John Wiley and Sons Ltd, 1992.

[150] J., K. H.. **A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise**. J. Basic Eng, 86:97–106, 1964.

[151] ZHILINSKAS, A. G.. **Single-step Bayesian search method for an extremum of functions of a single variable**. Cybernetics, 11:160–166, 1975.

[152] MOCKUS, J.; TIESIS, V. ; ZILINSKAS, A.. **The application of Bayesian methods for seeking the extremum**. In: Dixon, L.; Szego, G., editors, TOWARDS GLOBAL OPTIMISATION, volumen 2, p. 117–129, 1978.

[153] QIAN, C.; XIONG, H. ; XUE, K.. **Bayesian optimization using Pseudo-Points**. In: Bessiere, C., editor, PROCEEDINGS OF THE TWENTY-NINTH INTERNATIONAL JOINT CONFERENCE ON ARTIFICIAL INTELLIGENCE, p. 3044–3050. International Joint Conferences on Artificial Intelligence Organization, 2020.

[154] FRAZIER, P. I.. **A tutorial on Bayesian optimization**. 2018.

[155] JONES, D. R.; SCHONLAU, M. ; WELCH, W. J.. **Efficient global optimization of expensive black-box functions**. Journal of Global Optimization, 13:455–492, 1998.

[156] SRINIVAS, N.; KRAUSE, A.; KAKADE, S. M. ; SEEGER, M. W.. **Information-theoretic regret bounds for Gaussian process optimization in the bandit setting**. IEEE Transactions on Information Theory, 58(5):3250–3265, 2012.

[157] LIN, N.; NOE, D. ; HE, X.. **Springer handbook of engineering statistics**. In: Pham, H., editor, TREE-BASED METHODS AND THEIR APPLICATIONS, London, 2006. Springer-Verlag London.

[158] LANDAU, S.; BARTHEL, S.. **Recursive partitioning**. In: Peterson, P.; Baker, E. ; McGaw, B., editors, INTERNATIONAL ENCYCLOPEDIA OF EDUCATION, p. 383 – 389. Elsevier, Oxford, 3rd edition, 2010.

[159] BREIMAN, L.; FRIEDMAN, J. H.; OLSHEN, R. A. ; STONE, C. J.. **Classification And Regression Trees**. CRC Press, Boca Raton, 1st edition, 1984.

[160] A.J., I.. **Modern Multivariate Statistical Techniques**, chapter Recursive Partitioning and Tree-Based Methods, p. 281–314. Springer, New York, 1st edition, 2013.

[161] KUHN, M.; JOHNSON, K.. **Applied Predictive Modeling**. Springer, New York, 1st edition, 2013.

[162] AMONTONS, G.. **De la resistance causee dans les machines**. Memoires de lAcademie Royale des Sciences, p. 206–226, 1699.

[163] COULOMB, C. A.. **Theorie des machines simples, en ayant egard au frottement de leurs parties, et a la roideur des cordages**. Paris, France, 1785.

[164] POPOVA, E.; POPOV, V. L.. **The research works of Coulomb and Amontons and generalized laws of friction**. Friction, 3:183–190, 2015.

[165] MARQUES, F.; FLORES, P.; CLARO, J. C. P. ; LANKARANI, H. M.. **A survey and comparison of several friction force models for dynamic analysis of multibody mechanical systems**. Nonlinear Dynamics, 86:1407–1443, 2016.

[166] KHALIL, W.; DOMBRE, E.. **Modeling, Identification and Control of Robots**. Butterworth-Heinemann, London, UK, 1st edition, 2004.

[167] JOHANASTROM, K.; DE WIT, C. C.. **Revisiting the LuGre friction model**. IEEE Control Systems Magazine, 28(6):101–114, 2008.

[168] THRELFALL, D.. **The inclusion of coulomb friction in mechanisms programs with particular reference to dram au programme dram**. Mechanism and Machine Theory, 13(4):475 – 483, 1978.

[169] TUSTIN, A.. **The effects of backlash and of speed-dependent friction on the stability of closed-cycle control systems**. Journal of the Institution of Electrical Engineers - Part IIA: Automatic Regulators and Servo Mechanisms, 94(1):143–151, 1947.

[170] TAO, G.; LEWIS, F. L.. **Adaptive Control of Nonsmooth Dynamic Systems**. Springer-Verlag, London, UK, 1st edition, 2001.

[171] NIRANJAN, P.; SHETTY, S. C.; BYNDOOR, C. D.; SAIRAM, K. V. S. S. S. ; KARINKA, S.. **Friction identification and control of ball screw driven system using PLC**. In: 2016 IEEE INTERNATIONAL CONFERENCE ON RECENT TRENDS IN ELECTRONICS, INFORMATION COMMUNICATION TECHNOLOGY - RTEICT, p. 803–808. 2016.

[172] WU, L.. **Identification of friction parameters based on genetic algorithm in servo control system**. In: Deng H., Miao D., Lei J., Wang F.L. (eds) Artificial Intelligence and Computational Intelligence. AICI 2011. Lecture Notes in Computer Science, 7002, 2011.

[173] MÁRTON, L.; LANTOS, B.. **Friction modelling and robust adaptive compensation**. IFAC Proceedings Volumes, 38(1):257 – 262, 2005.

[174] GOMONWATTANAPANICH, O.; PATTANAPUKDEE, A. ; MONGKOLWONGROJN, M.. **Compensation and estimation of friction by using extended Kalman filter**. 2006 SICE-ICASE International Joint Conference Busan South Korea, 2006.

[175] BO, L.; PAVELESCU, D.. **The friction-speed relation and its influence on the critical velocity of stick-slip motion.** Wear, 82:277–289, 1982.

[176] JANOT, A.; GAUTIER, M. ; BRUNOT, M.. **Data set and reference models of EMPS**. Workshop on Nonlinear System Identification Benchmarks Eindhoven The Netherlands, 2019.

[177] FORGIONE, M.; PIGA, D.. **dynoNet: A neural network architecture for learning dynamical systems**. International Journal of Adaptive Control and Signal Processing, 35(4):612–626, 2021.

[178] FORGIONE, M.; PIGA, D.. **Continuous-time system identification with neural networks: model structures and fitting criteria**. European Journal of Control, 2021.

[179] JANOT, A.; YOUNG, P. C. ; GAUTIER, M.. **Identification and control of electro-mechanical systems using state-dependent parameter estimation**. International Journal of Control, 90:643–660, 2017.

[180] LEVENBERG, K.. **A method for the solution of certain non-linear problems in least squares**. Quarterly of Applied Mathematics, 2:164–168, 1944.

[181] MARQUARDT, D. W.. **An algorithm for least-squares estimation of nonlinear parameters**. SIAM Journal on Applied Mathematics, 11:431–441, 1963.

[182] GAUTIER, M.. **Dynamic identification of robots with power model**. volumen 3, p. 1922–1927. Proceedings of International Conference on Robotics and Automation, Albuquerque, NM, USA, 1997.

[183] SWEVERS, J.; VERDONCK, W. ; SCHUTTER, J. D.. **Dynamic model identification for industrial robots**. IEEE Control Systems Magazine, 27:5:58–71, 2007.

[184] GAUTIER, M.; BRIOT, S.. **Global identification of joint drive gains and dynamic parameters of robots**. Journal of Dynamic Systems, Measurement, and Control, 136:5:051025–051025–9, 2014.

[185] DE WIT, C. C.; LISCHINSKY, P.. **Adaptive friction compensation with dynamic friction model**. IFAC Proceedings, 29:2078–2083, 1996.

[186] LIU, L.; WU, Z.. **A new identification method of the Stribeck friction model based on limit cycles**. Journal of Mechanical Engineering Science, 228:2678–2683, 2014.

[187] RAISSI, M.; PERDIKARIS, P. ; KARNIADAKIS, G.. **Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations**. Journal of Computational Physics, 378:686 – 707, 2019.

[188] YANG, L.; ZHANG, D. ; KARNIADAKIS, G. E.. **Physics-informed generative adversarial networks for stochastic differential equations**. SIAM Journal on Scientific Computing, 42(1):A292–A317, 2020.

[189] HE, Q.; BARAJAS-SOLANO, D.; TARTAKOVSKY, G. ; TARTAKOVSKY, A. M.. **Physics-informed neural networks for multiphysics data assimilation with application to subsurface transport**. Advances in Water Resources, 141:103610, 2020.

A
Tables

A.1
Dynamic Parameters

Table A-1: Dynamic parameters of the asymmetric friction models.

Model	**EA**	**M**	$\mathbf{F_C^+}$	$\mathbf{F_C^-}$	$\mathbf{F_V^+}$	$\mathbf{F_V^-}$	$\mathbf{F_S^+}$	$\mathbf{F_S^-}$	δ_σ	$\mathbf{v_S}$	$\mathbf{v_o}$
	ACO	95.31± 0.18	19.68± 0.88	20.5± 1.24	169.11± 9.56	239.36± 11.89	-	-	-	-	-
	BBO	95.33± 0.07	19.7± 0.22	20.09± 0.14	168.24± 2.68	243.13± 1.53	-	-	-	-	-
F_C^a	DE	95.32± 0.08	19.48± 0.5	20.07± 0.55	170.75± 5.55	243.61± 5.6	-	-	-	-	-
	GA	95.22± 0.27	19.59± 0.59	19.55± 0.41	169.78± 5.48	249.89± 4.75	-	-	-	-	-
	PSO	95.31± 0.42	20± 0.29	20.31± 0.37	164.01± 3.6	240.03± 4.68	-	-	-	-	-
	ACO	95.26± 0.42	20.73± 1.41	20.5± 1.48	156.58± 14.93	239.06± 14.43	-	-	-	-	0.11± 0.01
	BBO	95.31± 0.06	19.97± 0.28	20.03± 0.18	165.08± 3.05	243.74± 1.95	-	-	-	-	0.05± 0.01
F_F^a	DE	95.24± 0.21	19.92± 0.88	19.82± 0.45	165.77± 10.35	246.08± 5.01	-	-	-	-	0.06± 0.03
	GA	95.29± 0.20	19.57± 0.54	19.63± 0.45	170.13± 5.42	248.13± 4.94	-	-	-	-	0.04± 0.03
	PSO	95.34± 1.65	17.86± 1.2	18.07± 1.02	166.21± 10.22	244.57± 9.48	-	-	-	-	0.05± 0.02
	ACO	95.70± 0.68	15.29± 5.86	25.93± 7.11	198.13± 33.66	206.98± 35.69	23.35± 7.21	22.67± 8.51	-	0.16± 0.03	-
	BBO	95.40± 0.24	17.69± 1.12	23.95± 1.48	187.13± 6.92	205.90± 10.46	21.28± 0.59	23.12± 1.58	-	0.07± 0.01	-
F_T^a	DE	95.40± 0.36	18.20± 7.87	20.71± 9.64	180.31± 49.54	236.48± 61.60	19.53± 3.91	20.07± 9.09	-	0.104± 0.04	-
	GA	95.30± 0.52	15.94± 5.60	25.88± 7.10	195.81± 31.71	210.33± 32.60	21.09± 3.23	25.13± 7.36	-	0.09± 0.01	-
	PSO	95.61± 1.22	18.52± 1.65	24.34± 2.94	186.22± 9.91	202.18± 14.06	22.90± 2.38	23.42± 3.26	-	0.07± 0.01	-
	ACO	95.17± 1.08	21.89± 3.67	21.24± 3.12	159.49± 21.25	238.03± 20.4	16.81± 3.23	18.23± 3.22	1.75± 0.66	0.1± 0.07	-
	BBO	95.27± 0.12	22.06± 0.66	21.4± 0.62	147.08± 5.72	231.64± 5.79	14.2± 1.26	16.02± 1.05	1.55± 0.22	0.04± 0.03	-
F_V^a	DE	95.3± 0.24	22.64± 5.93	20.09± 4.15	151.55± 30.45	244.57± 24.92	16.4± 2.96	17.79± 2.15	1.74± 0.83	0.07± 0.06	-
	GA	95.22± 0.32	21.94± 2.57	21.53± 1.67	150.50± 19.70	230.91± 12.63	11.68± 3.28	12.79± 4.29	1.33± 0.46	0.02± 0.02	-
	PSO	95.35± 0.77	21.65± 1.03	21.6± 1.02	153.77± 8.98	231.93± 8.94	17.79± 1.35	18.48± 1.8	1.92± 0.21	0.05± 0.01	-

Table A-2: Dynamic parameters of the symmetric friction models.

Model	**EA**	**M**	$\mathbf{F_C}$	$\mathbf{F_V}$	$\mathbf{F_S}$	**Off**	δ_σ	$\mathbf{v_S}$	$\mathbf{v_o}$
	ACO	95.37± 0.19	19.46± 1.25	209.85± 15.14	-	-3.14± 0.07	-	-	-
	BBO	95.27± 0.03	19.65± 0.19	208.27± 2.27	-	-3.12± 0.03	-	-	-
F_C^s	DE	95.28± 0.001	19.4± 0.005	211.36± 0.06	-	-3.13± 0.001	-	-	-
	GA	95.31± 0.20	19.32± 0.73	212.48± 8.33	-	-3.11± 0.12	-	-	-
	PSO	95.04± 0.53	19.88± 0.57	206.4± 5.96	-	-2.82± 0.24	-	-	-
	ACO	95.24± 0.44	19.59± 0.89	208.69± 8.02	-	-3.19± 0.29	-	-	0.11± 0.01
	BBO	95.27± 0.03	19.66± 0.11	208.4± 1.22	-	-3.12± 0.01	-	-	0.05± 0.01
F_F^s	DE	95.28± 0.01	19.44± 0.12	210.89± 1.39	-	-3.13± 0.004	-	-	0.07± 0.02
	GA	95.30± 0.08	19.50± 0.77	210.06± 7.5	-	-3.09± 0.09	-	-	0.04± 0.03
	PSO	95.45± 0.88	19.87± 0.65	203.98± 6.57	-	-2.51± 0.62	-	-	0.05± 0.02
	ACO	95.05± 0.41	19.17± 2.38	200.34± 28.39	17.64± 4.12	-1.12± 2.40	-	0.11± 0.06	
	BBO	95.35± 0.26	20.51± 0.55	177.76± 4.30	19.57± 0.77	-1.12 ± 0.50	-	0.06± 0.01	
F_T^s	DE	94.33± 0.15	19.66± 0.51	179.63± 13.77	17.99± 1.41	-0.23± 1.61	-	0.07± 0.02	
	GA	95.30± 1.48	19.46± 1.27	176.65± 18.16	17.15± 2.18	0.53± 2.06	-	0.08± 0.02	
	PSO	95.34± 1.23	20.12± 1.15	187.54± 12.75	19.28± 1.53	-0.97± 1.14	-	0.06± 0.01	
	ACO	95.29± 0.8	20.21± 5.25	196.56± 35.68	17.51± 2.45	-3.29± 0.33	2.07± 0.77	0.08± 0.05	
	BBO	95.23± 0.12	20.94± 1.14	196.72± 9.92	14.24± 1.72	-3.05± 0.08	1.66± 0.4	0.02± 0.02	
F_V^s	DE	95.3± 0.18	18.2± 10.53	222.11± 58.98	15.67± 4.82	-3.14± 0.05	2.18± 1.28	0.07± 0.03	
	GA	95.28± 0.32	22.55± 3.92	182.10± 31.52	15.02± 2.86	-3.13± 0.23	1.51± 0.69	0.03± 0.03	
	PSO	95.1± 0.66	20.54± 2.98	200.13± 25.14	18.28± 1.68	-2.78± 0.34	1.95± 0.37	0.05± 0.02	

A.2
Test Results

Table A-3: Evaluation Metrics of F_C^a friction model.

Test	**ACO**	$\mathbf{\Delta_{Val}}(\%)$	**BBO**	$\mathbf{\Delta_{Val}}(\%)$	**DE**	$\mathbf{\Delta_{Val}}(\%)$	**GA**	$\mathbf{\Delta_{Val}}(\%)$	**PSO**	$\mathbf{\Delta_{Val}}(\%)$
1	8.030E-03	-0.91	8.086E-03	-1.61	8.033E-03	-0.95	8.023E-03	-0.82	8.144E-03	-2.35
2	8.067E-03	-1.38	8.052E-03	-1.19	8.061E-03	-1.30	**7.876E-03**	**1.02**	8.071E-03	-1.43
3	8.030E-03	-0.92	8.056E-03	-1.23	8.033E-03	-0.95	8.019E-03	-0.77	8.119E-03	-2.03
4	8.052E-03	-1.19	8.046E-03	-1.11	8.074E-03	-1.46	8.087E-03	-1.62	8.079E-03	-1.53
5	7.967E-03	-0.13	8.051E-03	-1.18	8.033E-03	-0.95	7.991E-03	-0.42	8.065E-03	-1.36
6	8.047E-03	-1.13	8.052E-03	-1.18	8.036E-03	-0.99	8.078E-03	-1.51	8.136E-03	-2.24
7	8.000E-03	-0.53	8.065E-03	-1.35	8.035E-03	-0.98	8.086E-03	-1.61	8.062E-03	-1.31
8	8.109E-03	-1.91	8.070E-03	-1.42	8.051E-03	-1.17	8.068E-03	-1.39	8.113E-03	-1.95
9	8.056E-03	-1.24	8.063E-03	-1.32	8.046E-03	-1.12	7.975E-03	-0.23	8.085E-03	-1.60
10	8.024E-03	-0.83	8.038E-03	-1.01	8.067E-03	-1.38	7.968E-03	-0.13	8.104E-03	-1.84
11	**7.915E-03**	**0.54**	8.057E-03	-1.25	8.033E-03	-0.95	7.986E-03	-0.35	8.087E-03	-1.62
12	8.091E-03	-1.68	8.063E-03	-1.33	8.033E-03	-0.95	8.022E-03	-0.81	8.093E-03	-1.70
13	8.043E-03	-1.07	8.046E-03	-1.11	8.039E-03	-1.03	7.983E-03	-0.32	8.098E-03	-1.77
14	8.075E-03	-1.48	8.067E-03	-1.37	8.036E-03	-0.98	8.126E-03	-2.12	8.086E-03	-1.62
15	8.082E-03	-1.56	8.050E-03	-1.17	8.061E-03	-1.30	8.017E-03	-0.74	8.093E-03	-1.71
16	8.029E-03	-0.90	8.042E-03	-1.06	8.058E-03	-1.27	7.993E-03	-0.44	8.131E-03	-2.19
17	8.168E-03	-2.65	8.059E-03	-1.27	8.036E-03	-0.99	8.002E-03	-0.56	8.100E-03	-1.79
18	7.973E-03	-0.20	8.062E-03	-1.31	8.051E-03	-1.17	8.090E-03	-1.67	8.111E-03	-1.93
19	8.065E-03	-1.36	8.060E-03	-1.28	8.033E-03	-0.95	8.073E-03	-1.45	8.082E-03	-1.56
20	8.014E-03	-0.71	8.074E-03	-1.47	8.069E-03	-1.40	8.030E-03	-0.91	8.053E-03	-1.20
μ	8.042E-03	-1.06	8.058E-03	-1.26	8.046E-03	-1.11	8.025E-03	-0.84	8.096E-03	-1.74
σ	5.33E-05	6.70E-01	1.12E-05	1.41E-01	1.39E-05	1.74E-01	5.61E-05	7.05E-01	2.42E-05	3.04E-01

Table A-4: Evaluation Metrics of F_C^s friction model.

Test	**ACO**	$\mathbf{\Delta_{Val}}(\%)$	**BBO**	$\mathbf{\Delta_{Val}}(\%)$	**DE**	$\mathbf{\Delta_{Val}}(\%)$	**GA**	$\mathbf{\Delta_{Val}}(\%)$	**PSO**	$\mathbf{\Delta_{Val}}(\%)$
1	8.150E-03	-1.56	8.130E-03	-1.31	8.102E-03	-0.96	8.072E-03	-0.59	8.126E-03	-1.27
2	8.241E-03	-2.69	8.119E-03	-1.17	8.102E-03	-0.96	8.149E-03	-1.55	8.121E-03	-1.19
3	8.437E-03	-5.14	8.104E-03	-0.98	8.102E-03	-0.96	8.195E-03	-2.12	8.201E-03	-2.19
4	**7.978E-03**	**0.58**	8.155E-03	-1.63	8.102E-03	-0.96	**8.001E-03**	**0.30**	8.119E-03	-1.17
5	**7.991E-03**	**0.42**	8.101E-03	-0.96	8.102E-03	-0.96	8.182E-03	-1.96	8.103E-03	-0.98
6	**8.024E-03**	**0.01**	8.110E-03	-1.06	8.102E-03	-0.96	8.203E-03	-2.22	8.151E-03	-1.57
7	8.111E-03	-1.08	8.105E-03	-1.00	8.102E-03	-0.96	8.051E-03	-0.33	8.189E-03	-2.05
8	8.148E-03	-1.54	8.129E-03	-1.30	8.102E-03	-0.96	**8.019E-03**	**0.08**	8.114E-03	-1.11
9	8.275E-03	-3.11	8.098E-03	-0.91	8.103E-03	-0.97	8.139E-03	-1.42	8.116E-03	-1.14
10	8.149E-03	-1.55	8.123E-03	-1.22	8.102E-03	-0.96	8.106E-03	-1.01	8.194E-03	-2.11
11	8.129E-03	-1.30	8.154E-03	-1.61	8.102E-03	-0.96	8.114E-03	-1.11	8.121E-03	-1.20
12	**7.973E-03**	**0.64**	8.120E-03	-1.18	8.102E-03	-0.96	**7.974E-03**	**0.63**	8.160E-03	-1.69
13	8.179E-03	-1.93	8.104E-03	-0.99	8.102E-03	-0.96	8.092E-03	-0.83	8.138E-03	-1.41
14	8.075E-03	-0.62	8.117E-03	-1.15	8.102E-03	-0.96	8.121E-03	-1.20	8.147E-03	-1.52
15	8.099E-03	-0.92	8.150E-03	-1.57	8.102E-03	-0.96	8.112E-03	-1.08	8.156E-03	-1.64
16	8.250E-03	-2.81	8.127E-03	-1.28	8.102E-03	-0.96	8.027E-03	-0.02	8.115E-03	-1.13
17	8.205E-03	-2.25	8.188E-03	-2.03	8.102E-03	-0.96	8.063E-03	-0.48	8.103E-03	-0.97
18	8.025E-03	-0.01	8.103E-03	-0.97	8.102E-03	-0.96	**7.963E-03**	**0.77**	8.128E-03	-1.29
19	8.171E-03	-1.82	8.117E-03	-1.15	8.102E-03	-0.96	8.129E-03	-1.30	8.113E-03	-1.10
20	8.142E-03	-1.46	8.114E-03	-1.11	8.102E-03	-0.96	8.169E-03	-1.79	8.117E-03	-1.15
μ	8.138E-03	-1.41	8.123E-03	-1.23	8.102E-03	-0.96	8.094E-03	-0.86	8.137E-03	-1.39
σ	1.107E-04	1.38E+00	2.231E-05	2.78E-01	2.257E-07	2.81E-03	6.966E-05	8.68E-01	2.915E-05	3.63E-01

Table A-5: Evaluation Metrics of F_F^a friction model.

Test	**ACO**		**BBO**		**DE**		**GA**		**PSO**	
number	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$
1	1.342E-02	8.618E-03	1.373E-02	8.027E-03	1.385E-02	**7.800E-03**	1.362E-02	8.161E-03	1.316E-02	8.697E-03
2	1.313E-02	9.074E-03	1.369E-02	8.064E-03	1.363E-02	8.280E-03	1.375E-02	8.033E-03	1.316E-02	8.694E-03
3	1.328E-02	8.928E-03	1.378E-02	7.912E-03	1.376E-02	**7.924E-03**	1.382E-02	**7.799E-03**	1.303E-02	8.774E-03
4	1.330E-02	8.760E-03	1.377E-02	7.930E-03	1.344E-02	8.673E-03	1.391E-02	**7.717E-03**	1.330E-02	8.519E-03
5	1.329E-02	8.783E-03	1.372E-02	7.966E-03	1.385E-02	**7.770E-03**	1.387E-02	**7.752E-03**	1.288E-02	8.864E-03
6	1.332E-02	8.820E-03	1.364E-02	8.196E-03	1.379E-02	**7.877E-03**	1.379E-02	**7.931E-03**	1.307E-02	8.584E-03
7	1.341E-02	8.721E-03	1.372E-02	8.000E-03	1.373E-02	8.075E-03	1.383E-02	**7.782E-03**	1.314E-02	8.529E-03
8	1.331E-02	8.728E-03	1.373E-02	7.992E-03	1.373E-02	**8.007E-03**	1.362E-02	8.195E-03	1.298E-02	8.701E-03
9	1.326E-02	8.924E-03	1.377E-02	7.907E-03	1.373E-02	8.077E-03	1.386E-02	**7.779E-03**	1.328E-02	8.582E-03
10	1.360E-02	8.401E-03	1.380E-02	7.876E-03	1.383E-02	**7.826E-03**	1.387E-02	**7.751E-03**	1.327E-02	8.473E-03
11	1.341E-02	8.680E-03	1.374E-02	7.956E-03	1.368E-02	8.185E-03	1.370E-02	8.175E-03	1.249E-02	9.286E-03
12	1.314E-02	9.063E-03	1.377E-02	7.911E-03	1.318E-02	8.923E-03	1.393E-02	**7.723E-03**	1.317E-02	8.671E-03
13	1.336E-02	8.748E-03	1.379E-02	7.895E-03	1.375E-02	8.025E-03	1.382E-02	**7.891E-03**	1.257E-02	9.439E-03
14	1.321E-02	8.994E-03	1.379E-02	7.847E-03	1.339E-02	8.621E-03	1.386E-02	**7.772E-03**	1.337E-02	8.378E-03
15	1.332E-02	8.833E-03	1.371E-02	8.036E-03	1.373E-02	**8.004E-03**	1.371E-02	8.103E-03	1.257E-02	9.226E-03
16	1.350E-02	8.523E-03	1.374E-02	8.002E-03	1.363E-02	8.234E-03	1.377E-02	**7.875E-03**	1.287E-02	8.795E-03
17	1.347E-02	8.503E-03	1.376E-02	7.974E-03	1.372E-02	8.080E-03	1.382E-02	**7.797E-03**	1.278E-02	9.126E-03
18	1.341E-02	8.522E-03	1.377E-02	7.933E-03	1.376E-02	**7.881E-03**	1.352E-02	8.486E-03	1.348E-02	8.230E-03
19	1.370E-02	8.084E-03	1.372E-02	8.022E-03	1.371E-02	**8.007E-03**	1.380E-02	**7.904E-03**	1.353E-02	8.284E-03
20	1.311E-02	8.896E-03	1.375E-02	7.971E-03	1.378E-02	**7.933E-03**	1.378E-02	**7.976E-03**	1.297E-02	8.998E-03
μ	**1.3348E-02**	8.7302E-03	**1.3745E-02**	**7.9708E-03**	**1.3684E-02**	8.1102E-03	1.3783E-02	**7.9301E-03**	**1.3055E-02**	8.7426E-03
σ	**1.4652E-04**	2.3502E-04	**3.8403E-05**	**7.6349E-05**	**1.6206E-04**	2.9978E-04	1.0143E-04	1.9955E-04	**2.8648E-04**	3.2258E-04

Table A-6: Evaluation Metrics of F_F^s friction model.

Test	**ACO**		**BBO**		**DE**		**GA**		**PSO**	
number	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$
1	1.319E-02	9.011E-03	1.374E-02	8.025E-03	1.372E-02	8.087E-03	1.385E-02	**7.818E-03**	1.367E-02	8.064E-03
2	1.352E-02	8.581E-03	1.363E-02	8.282E-03	1.347E-02	8.582E-03	1.374E-02	8.132E-03	1.359E-02	8.192E-03
3	1.339E-02	8.822E-03	1.377E-02	**7.961E-03**	1.371E-02	8.111E-03	1.379E-02	**7.898E-03**	1.373E-02	**7.946E-03**
4	1.334E-02	8.915E-03	1.371E-02	8.040E-03	1.378E-02	**7.921E-03**	1.367E-02	8.222E-03	1.366E-02	**8.022E-03**
5	1.334E-02	8.774E-03	1.375E-02	**7.998E-03**	1.364E-02	8.272E-03	1.368E-02	8.103E-03	1.354E-02	8.373E-03
6	1.357E-02	8.445E-03	1.376E-02	**7.977E-03**	1.382E-02	**7.858E-03**	1.366E-02	8.209E-03	1.347E-02	8.425E-03
7	1.345E-02	8.660E-03	1.374E-02	**8.001E-03**	1.367E-02	8.206E-03	1.358E-02	8.438E-03	1.375E-02	**7.925E-03**
8	1.362E-02	8.391E-03	1.371E-02	8.089E-03	1.375E-02	**8.019E-03**	1.387E-02	**7.834E-03**	1.356E-02	8.450E-03
9	1.323E-02	9.022E-03	1.361E-02	8.290E-03	1.357E-02	8.425E-03	1.383E-02	**7.844E-03**	1.357E-02	8.412E-03
10	1.335E-02	8.855E-03	1.379E-02	**7.922E-03**	1.370E-02	8.145E-03	1.381E-02	**7.900E-03**	1.359E-02	8.302E-03
11	1.332E-02	8.895E-03	1.377E-02	**7.976E-03**	1.371E-02	8.119E-03	1.364E-02	8.043E-03	1.375E-02	**7.994E-03**
12	1.321E-02	9.083E-03	1.375E-02	8.016E-03	1.362E-02	8.322E-03	1.392E-02	**7.833E-03**	1.373E-02	7.976E-03
13	1.339E-02	8.769E-03	1.378E-02	**7.927E-03**	1.370E-02	8.139E-03	1.374E-02	**7.915E-03**	1.354E-02	8.304E-03
14	1.347E-02	8.684E-03	1.375E-02	**7.992E-03**	1.345E-02	8.671E-03	1.369E-02	**7.997E-03**	1.346E-02	8.296E-03
15	1.353E-02	8.625E-03	1.374E-02	**8.011E-03**	1.368E-02	8.175E-03	1.392E-02	**7.835E-03**	1.366E-02	8.195E-03
16	1.342E-02	8.829E-03	1.376E-02	**7.986E-03**	1.374E-02	8.039E-03	1.366E-02	8.283E-03	1.361E-02	8.166E-03
17	1.332E-02	8.968E-03	1.370E-02	8.116E-03	1.334E-02	8.906E-03	1.359E-02	8.422E-03	1.368E-02	8.078E-03
18	1.342E-02	8.652E-03	1.375E-02	**7.999E-03**	1.367E-02	8.207E-03	1.386E-02	**7.804E-03**	1.368E-02	**8.023E-03**
19	1.296E-02	9.436E-03	1.374E-02	**7.991E-03**	1.383E-02	**7.843E-03**	1.362E-02	8.193E-03	1.360E-02	8.160E-03
20	1.324E-02	8.851E-03	1.376E-02	**7.972E-03**	1.365E-02	8.251E-03	1.377E-02	8.090E-03	1.343E-02	8.522E-03
μ	**1.3364E-02**	8.8135E-03	**1.3736E-02**	8.0286E-03	**1.3662E-02**	8.2149E-03	**1.3744E-02**	8.0406E-03	**1.3614E-02**	8.1913E-03
σ	**1.4802E-04**	2.3032E-04	**4.3594E-05**	9.6708E-05	**1.2024E-04**	2.5980E-04	**1.0374E-04**	1.9899E-04	**9.3639E-05**	1.8050E-04

Table A-7: Evaluation Metrics of F_T^a friction model.

Test	**ACO**		**BBO**		**DE**		**GA**		**PSO**	
number	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$
1	1.351E-02	8.286E-03	1.369E-02	9.100E-03	1.363E-02	8.250E-03	1.388E-02	1.029E-02	**1.360E-02**	8.367E-03
2	1.381E-02	8.812E-03	1.368E-02	8.713E-03	1.367E-02	9.655E-03	1.364E-02	9.994E-03	**1.375E-02**	9.053E-03
3	1.377E-02	9.490E-03	1.369E-02	8.614E-03	1.370E-02	1.033E-02	1.369E-02	8.421E-03	1.407E-02	9.520E-03
4	1.397E-02	1.022E-02	1.372E-02	8.417E-03	1.370E-02	9.051E-03	1.348E-02	**7.787E-03**	**1.373E-02**	8.437E-03
5	1.380E-02	8.389E-03	1.358E-02	9.183E-03	1.369E-02	**7.419E-03**	1.361E-02	8.760E-03	1.395E-02	9.547E-03
6	1.356E-02	**7.720E-03**	1.366E-02	8.581E-03	1.369E-02	9.300E-03	1.382E-02	**7.467E-03**	1.424E-02	9.145E-03
7	1.363E-02	8.794E-03	1.374E-02	8.863E-03	1.369E-02	**7.040E-03**	1.381E-02	8.720E-03	**1.353E-02**	8.371E-03
8	1.364E-02	8.953E-03	1.366E-02	8.877E-03	1.370E-02	1.005E-02	1.368E-02	9.160E-03	1.379E-02	9.413E-03
9	1.414E-02	9.657E-03	1.366E-02	8.763E-03	1.372E-02	8.604E-03	1.370E-02	9.463E-03	**1.365E-02**	8.438E-03
10	1.352E-02	**7.271E-03**	1.369E-02	8.865E-03	1.363E-02	**5.971E-03**	1.357E-02	**7.566E-03**	**1.375E-02**	9.894E-03
11	1.367E-02	9.272E-03	1.369E-02	9.169E-03	1.365E-02	**5.817E-03**	1.358E-02	8.166E-03	1.395E-02	8.294E-03
12	1.364E-02	1.023E-02	1.367E-02	8.820E-03	1.367E-02	**8.023E-03**	1.360E-02	**7.201E-03**	1.410E-02	8.976E-03
13	1.358E-02	**7.614E-03**	1.372E-02	8.372E-03	1.369E-02	8.802E-03	1.382E-02	9.659E-03	**1.370E-02**	8.283E-03
14	1.362E-02	9.303E-03	1.374E-02	8.418E-03	1.373E-02	**7.767E-03**	1.388E-02	8.655E-03	1.408E-02	9.016E-03
15	1.376E-02	1.125E-02	1.370E-02	8.562E-03	1.372E-02	8.792E-03	1.389E-02	8.656E-03	1.384E-02	9.205E-03
16	1.369E-02	9.172E-03	1.367E-02	8.788E-03	1.370E-02	9.990E-03	1.392E-02	8.471E-03	1.394E-02	8.215E-03
17	1.371E-02	**7.681E-03**	1.366E-02	9.096E-03	1.371E-02	8.463E-03	1.346E-02	8.824E-03	**1.366E-02**	9.361E-03
18	1.393E-02	1.005E-02	1.368E-02	8.461E-03	1.369E-02	8.852E-03	1.383E-02	8.508E-03	**1.377E-02**	8.464E-03
19	1.367E-02	**7.688E-03**	1.368E-02	9.110E-03	1.370E-02	**7.231E-03**	1.380E-02	9.185E-03	**1.368E-02**	8.554E-03
20	1.382E-02	1.123E-02	1.370E-02	8.684E-03	1.372E-02	8.846E-03	1.332E-02	**7.811E-03**	1.422E-02	9.686E-03
μ	**1.372E-02**	9.054E-03	**1.368E-02**	8.773E-03	**1.369E-02**	8.413E-03	**1.370E-02**	8.638E-03	1.385E-02	8.912E-03
σ	1.543E-04	1.138E-03	3.410E-05	2.557E-04	2.606E-05	1.222E-03	1.624E-04	8.132E-04	2.015E-04	5.290E-04

Table A-8: Evaluation Metrics of F_T^s friction model.

Test	**ACO**		**BBO**		**DE**		**GA**		**PSO**	
number	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$
1	1.380E-02	8.437E-03	1.372E-02	9.130E-03	1.370E-02	9.674E-03	1.356E-02	1.001E-02	**1.373E-02**	9.231E-03
2	1.380E-02	**7.953E-03**	1.374E-02	9.069E-03	1.370E-02	9.306E-03	1.352E-02	8.949E-03	1.423E-02	**7.123E-03**
3	1.382E-02	8.862E-03	1.370E-02	9.239E-03	1.370E-02	9.289E-03	1.389E-02	8.675E-03	1.402E-02	9.158E-03
4	**1.376E-02**	9.587E-03	1.374E-02	8.961E-03	1.371E-02	9.239E-03	1.380E-02	8.678E-03	**1.372E-02**	9.019E-03
5	**1.376E-02**	**7.789E-03**	1.381E-02	8.613E-03	1.372E-02	9.305E-03	1.377E-02	9.226E-03	1.387E-02	9.263E-03
6	**1.377E-02**	8.130E-03	1.371E-02	8.748E-03	1.371E-02	9.236E-03	1.374E-02	9.464E-03	**1.377E-02**	9.362E-03
7	1.381E-02	**7.542E-03**	1.369E-02	9.108E-03	1.374E-02	8.508E-03	1.367E-02	1.052E-02	1.379E-02	9.682E-03
8	**1.366E-02**	9.197E-03	1.373E-02	9.282E-03	1.370E-02	9.784E-03	1.394E-02	1.030E-02	1.388E-02	8.556E-03
9	1.382E-02	9.740E-03	1.371E-02	9.134E-03	1.371E-02	8.868E-03	1.375E-02	9.253E-03	**1.369E-02**	8.481E-03
10	1.397E-02	8.171E-03	1.371E-02	9.372E-03	1.371E-02	9.517E-03	1.354E-02	1.101E-02	**1.371E-02**	8.922E-03
11	1.381E-02	8.486E-03	1.371E-02	9.348E-03	1.370E-02	9.688E-03	1.382E-02	8.571E-03	**1.374E-02**	9.053E-03
12	1.378E-02	9.361E-03	1.375E-02	8.862E-03	1.369E-02	1.097E-02	1.370E-02	8.933E-03	1.381E-02	9.096E-03
13	**1.368E-02**	1.011E-02	1.375E-02	9.012E-03	1.374E-02	8.632E-03	1.359E-02	1.006E-02	**1.371E-02**	8.461E-03
14	**1.366E-02**	8.820E-03	1.369E-02	9.460E-03	1.371E-02	9.385E-03	1.432E-02	1.069E-02	1.410E-02	**7.803E-03**
15	1.409E-02	8.712E-03	1.369E-02	9.203E-03	1.371E-02	9.385E-03	1.369E-02	1.022E-02	1.394E-02	9.214E-03
16	1.388E-02	9.061E-03	1.387E-02	9.357E-03	1.375E-02	9.114E-03	1.380E-02	9.826E-03	**1.374E-02**	8.887E-03
17	1.392E-02	8.061E-03	1.371E-02	9.163E-03	1.370E-02	9.818E-03	1.371E-02	9.461E-03	**1.369E-02**	9.160E-03
18	**1.372E-02**	9.928E-03	1.379E-02	8.879E-03	1.370E-02	9.290E-03	1.363E-02	1.095E-02	1.382E-02	9.919E-03
19	1.384E-02	9.067E-03	1.375E-02	9.148E-03	1.369E-02	9.517E-03	1.374E-02	8.945E-03	1.389E-02	8.987E-03
20	**1.375E-02**	**7.994E-03**	1.373E-02	8.807E-03	1.368E-02	9.638E-03	1.369E-02	9.771E-03	1.387E-02	8.965E-03
μ	1.381E-02	8.751E-03	**1.373E-02**	9.095E-03	**1.371E-02**	9.408E-03	**1.374E-02**	9.675E-03	1.384E-02	8.917E-03
σ	1.000E-04	7.302E-04	4.384E-05	2.217E-04	1.824E-05	4.937E-04	1.708E-04	7.556E-04	1.430E-04	6.028E-04

Table A-9: Evaluation Metrics of F_V^a friction model.

Test	**ACO**		**BBO**		**DE**		**GA**		**PSO**	
number	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$
1	1.3937E-02	7.770E-03	1.3653E-02	7.999E-03	1.3692E-02	7.995E-03	1.3743E-02	7.919E-03	1.3693E-02	8.027E-03
2	1.3735E-02	7.987E-03	1.3687E-02	7.969E-03	1.3679E-02	8.031E-03	1.3568E-02	8.080E-03	1.3701E-02	8.016E-03
3	1.3864E-02	7.836E-03	1.3687E-02	7.975E-03	1.3619E-02	8.035E-03	1.3705E-02	7.952E-03	1.3904E-02	7.803E-03
4	1.3849E-02	7.868E-03	1.3641E-02	8.008E-03	1.3693E-02	8.017E-03	1.3556E-02	8.091E-03	1.3698E-02	8.015E-03
5	1.3865E-02	7.856E-03	1.3668E-02	8.005E-03	1.3608E-02	8.045E-03	1.3677E-02	8.018E-03	1.3703E-02	8.012E-03
6	1.3679E-02	8.059E-03	1.3662E-02	7.991E-03	1.3681E-02	8.039E-03	1.3708E-02	7.925E-03	1.3714E-02	7.970E-03
7	1.3626E-02	8.090E-03	1.3659E-02	7.975E-03	1.3680E-02	8.032E-03	1.3616E-02	8.042E-03	1.3704E-02	7.973E-03
8	1.3768E-02	7.922E-03	1.3677E-02	7.975E-03	1.3623E-02	8.035E-03	1.3628E-02	8.061E-03	1.3742E-02	7.980E-03
9	1.3740E-02	7.998E-03	1.3688E-02	7.976E-03	1.3696E-02	7.946E-03	1.3468E-02	8.177E-03	1.3744E-02	7.948E-03
10	1.3742E-02	7.991E-03	1.3637E-02	8.032E-03	1.3683E-02	8.029E-03	1.3637E-02	8.016E-03	1.3750E-02	7.967E-03
11	1.3675E-02	8.042E-03	1.3640E-02	8.004E-03	1.3715E-02	8.003E-03	1.3664E-02	7.995E-03	1.3678E-02	7.993E-03
12	1.3770E-02	7.910E-03	1.3651E-02	7.997E-03	1.3709E-02	8.001E-03	1.3664E-02	8.011E-03	1.3665E-02	7.995E-03
13	1.3611E-02	8.103E-03	1.3684E-02	7.997E-03	1.3682E-02	8.030E-03	1.3612E-02	8.011E-03	1.3727E-02	7.955E-03
14	1.3690E-02	8.028E-03	1.3654E-02	8.008E-03	1.3690E-02	8.012E-03	1.3626E-02	8.068E-03	1.3696E-02	8.015E-03
15	1.3507E-02	8.210E-03	1.3650E-02	8.009E-03	1.3657E-02	8.006E-03	1.3623E-02	8.067E-03	1.3718E-02	7.991E-03
16	1.3680E-02	8.051E-03	1.3662E-02	7.986E-03	1.3645E-02	8.022E-03	1.3673E-02	7.979E-03	1.3743E-02	7.985E-03
17	1.3820E-02	7.841E-03	1.3683E-02	7.957E-03	1.3645E-02	8.010E-03	1.3733E-02	7.915E-03	1.3663E-02	8.049E-03
18	1.3827E-02	7.855E-03	1.3662E-02	7.989E-03	1.3644E-02	8.047E-03	1.3683E-02	7.959E-03	1.3768E-02	7.954E-03
19	1.3496E-02	8.242E-03	1.3676E-02	7.983E-03	1.3668E-02	8.022E-03	1.3696E-02	7.973E-03	1.3745E-02	7.968E-03
20	1.3745E-02	7.975E-03	1.3646E-02	7.999E-03	1.3717E-02	7.996E-03	1.3728E-02	7.976E-03	1.3721E-02	7.974E-03
μ	**1.3731E-02**	**7.982E-03**	**1.3663E-02**	**7.992E-03**	**1.3671E-02**	**8.018E-03**	**1.3650E-02**	**8.012E-03**	**1.3724E-02**	**7.980E-03**
σ	**1.1252E-04**	**1.230E-04**	**1.6604E-05**	**1.699E-05**	**3.0972E-05**	**2.261E-05**	**6.5562E-05**	**6.467E-05**	**4.9874E-05**	**4.822E-05**

Table A-10: Evaluation Metrics of F_V^s friction model.

Test	**ACO**		**BBO**		**DE**		**GA**		**PSO**	
number	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$	$\varepsilon_{\mathbf{Est}}$	$\varepsilon_{\mathbf{Val}}$
1	1.3629E-02	8.089E-03	1.3632E-02	8.064E-03	1.362E-02	8.093E-03	1.3601E-02	8.138E-03	1.3650E-02	8.193E-03
2	1.3582E-02	8.118E-03	1.3669E-02	8.025E-03	1.371E-02	8.033E-03	1.3633E-02	8.083E-03	1.3630E-02	8.162E-03
3	1.3640E-02	8.131E-03	1.3620E-02	8.079E-03	1.363E-02	8.107E-03	1.3637E-02	8.053E-03	1.3679E-02	8.097E-03
4	1.3538E-02	8.215E-03	1.3682E-02	8.035E-03	1.362E-02	8.076E-03	1.3441E-02	8.309E-03	1.3575E-02	8.169E-03
5	1.3355E-02	8.395E-03	1.3647E-02	8.061E-03	1.363E-02	8.104E-03	1.3652E-02	8.063E-03	1.3659E-02	8.105E-03
6	1.3590E-02	8.165E-03	1.3672E-02	8.059E-03	1.364E-02	8.087E-03	1.3510E-02	8.214E-03	1.3631E-02	8.072E-03
7	1.3755E-02	**8.005E-03**	1.3651E-02	8.112E-03	1.371E-02	8.048E-03	1.3660E-02	8.084E-03	1.3605E-02	8.197E-03
8	1.3637E-02	8.111E-03	1.3648E-02	8.048E-03	1.365E-02	8.113E-03	1.3436E-02	8.277E-03	1.3651E-02	8.129E-03
9	1.3649E-02	8.106E-03	1.3645E-02	8.057E-03	1.362E-02	8.118E-03	1.3756E-02	7.999E-03	1.3645E-02	8.117E-03
10	1.3929E-02	**7.756E-03**	1.3632E-02	8.069E-03	1.366E-02	8.090E-03	1.3623E-02	8.135E-03	1.3616E-02	8.104E-03
11	1.3555E-02	8.207E-03	1.3626E-02	8.083E-03	1.359E-02	8.086E-03	1.3502E-02	8.205E-03	1.3657E-02	8.098E-03
12	1.3628E-02	8.136E-03	1.3657E-02	8.064E-03	1.363E-02	8.088E-03	1.3659E-02	8.086E-03	1.3598E-02	8.144E-03
13	1.3644E-02	8.121E-03	1.3664E-02	8.040E-03	1.366E-02	8.098E-03	1.3583E-02	8.139E-03	1.3674E-02	8.109E-03
14	1.3621E-02	8.127E-03	1.3664E-02	8.032E-03	1.365E-02	8.051E-03	1.3584E-02	8.130E-03	1.3661E-02	8.126E-03
15	1.3692E-02	8.070E-03	1.3650E-02	8.055E-03	1.364E-02	8.094E-03	1.3438E-02	8.268E-03	1.3613E-02	8.236E-03
16	1.3629E-02	8.128E-03	1.3598E-02	8.123E-03	1.363E-02	8.092E-03	1.3830E-02	7.876E-03	1.3625E-02	8.129E-03
17	1.3692E-02	8.076E-03	1.3688E-02	**8.004E-03**	1.361E-02	8.100E-03	1.3657E-02	8.028E-03	1.3625E-02	8.073E-03
18	1.3699E-02	7.995E-03	1.3636E-02	8.051E-03	1.368E-02	8.050E-03	1.3532E-02	8.204E-03	1.3601E-02	8.118E-03
19	1.3865E-02	7.877E-03	1.3638E-02	8.059E-03	1.357E-02	8.136E-03	1.3624E-02	8.125E-03	1.3657E-02	8.125E-03
20	1.3647E-02	8.068E-03	1.3694E-02	**8.021E-03**	1.363E-02	8.062E-03	1.3668E-02	8.045E-03	1.3670E-02	8.113E-03
μ	**1.3649E-02**	8.095E-03	**1.3651E-02**	8.057E-03	**1.364E-02**	8.086E-03	**1.3601E-02**	8.123E-03	**1.3636E-02**	8.131E-03
σ	**1.1405E-04**	1.245E-04	**2.3342E-05**	2.783E-05	**3.378E-05**	2.536E-05	**1.0005E-04**	1.016E-04	**2.7813E-05**	4.102E-05