الجمهورية الجزائرية الديمقراطية الشعبية
Peoples Democratic Republic of Algeria
وزارة التعليم العالي والبحث العلمي
Ministry of Higher Education and Scientific Research
جامعة غرداية
University of Ghardaïa
كلية العلوم والتكنولوجيا
Faculty of Science and Technology
قسم الآلية و الكهروميكانيك
Department of Automation and Electromechanics

Dissertation at the end of the study, with a view to obtaining the diploma

Master

Field: Science and Technology.

Sector: Automatic.

Specialty: Automatic and systems.

Theme:

Synthesis of a Speed Regulator for a DC Motor by Genetic Algorithms

Presented by:

Ben Dekken Abdelaziz

Ben Hamouda Hamouda

Before the jury composed of:

FIHAKHIR Amine Mehdi	**MCB**	**University**	**Supervisor**
BELGACEM Bekkar	**MCB**	**University**	**Examiner**
Djellouli taher	**MAA**	**University**	**Examiner**

Academic year :2021/2022

Acknowledgement

FIRST OF ALL WE WOULD LIKE TO THANK OUR GOD FOR GIVING US THE FAITH AND THE COURAGE TO FINISH THIS PROJECT. WE WOULD LIKE TO EXPRESS OUR WARMEST THANKS TO OUR SUPERVISOR MR FIHAKHIR AMINE MEHDI , FOR HIS HELP, SUPPORT AND ADVICE WHICH PROVIDED US WITH GREAT MOTIVATION TO ACCOMPLISH THIS WORK.

FINALLY, WE WOULD LIKE TO EXPRESS OUR GRATITUDE TO ALL THOSE WHO HAVE CONTRIBUTED, DIRECTLY OR INDIRECTLY, TO THE REALIZATION OF THIS WORK

ملخص

الهدف الرئيسي من هذا المشروع هو تحسين معلمات وحدة التحكم الكلاسيكية (PID) للتحكم في سرعة محرك DC باستخدام تقنية تحسين الخوارزمية الجينية (GA). مطلوب دراسة للتقنيات الكلاسيكية المختلفة لتوليف وحدة التحكم ، ثم ننتقل إلى النمذجة الرياضية لمحركات DC ، تليها نظرة عامة موجزة على طريقة التحسين المختارة. أخيرا ، في هذا القسم ، نقوم بتحسين معلمات وحدة تحكم PID المستخدمة للتحكم في سرعة MCC ؛ ستسمح لنا عمليات المحاكاة تحت Matlab / Simulink بتقييم كفاءة الطريقة المستخدمة مقارنة بالطريقة الكلاسيكية المستخدمة في وحدات التحكم الاصطناعية.

الكلمات المفتاحية: GA,PID (خوارزمية وراثية), محرك DC,طريقة Z-N

Abstract

The main goal of this work is to optimize the parameters of a classical controller (PID) to control the speed of a DC motor using a genetic algorithm optimization technique (GA). A study of different classical techniques for controller synthesis is required, and we then proceed to mathematical modeling of DC motors, followed by a brief overview of the chosen optimization method. Finally, in this section, we optimize the parameters of the PID controller used for MCC speed control; simulations under Matlab/Simulink will allow us to evaluate the efficiency of the used method compared to the classical method used for synthetic controllers.

Keywords: PID, GA (Algorithm Genetic), DC Motor,Z-N method

Résume

L'objectif principal de ce travail est d'optimiser les paramètres d'un contrôleur classique (PID) pour contrôler la vitesse d'un moteur à courant continu à l'aide d'une technique d'optimisation d'algorithme génétique (GA). Une étude des différentes techniques classiques de synthèse des contrôleurs est nécessaire, puis nous passons à la modélisation mathématique des moteurs à courant continu, suivie d'un bref aperçu de la méthode d'optimisation choisie. Enfin, dans cette section, nous optimisons les paramètres du contrôleur PID utilisé pour le contrôle de la vitesse MCC; les simulations sous Matlab/Simulink nous permettront d'évaluer l'efficacité de la méthode utilisée par rapport à la méthode classique utilisée pour les contrôleurs synthétiques.

Mots-clés : PID, AG (Algorithme Génétique), Moteur à courant continu,Méthode Z-N

TABLE OF CONTENTS

LIST OF FIGURE

LIST OF FIGURES

LIST OF TABLES

General Introduction

General Introduction

At present, the development of automation allows to occupy an important place in the modern world. The pursuit of high performance, the design and construction of increasingly complex and highly reliable equipment allows automated systems to be analyzed and their operation optimized. The functioning of systems continues to play an ever-increasing role in human activity. It also gave him access to the realm. Science is higher than industry or economics.

In most industrial plants, physical quantities (speed, position, temperature, etc.) must be maintained at specified values regardless of internal or external changes that may affect these quantities. Therefore, an auto-tuning system consists of many elements that are connected in a way that enables them to counteract the effects of disturbances on the system. In such adaptive systems, measurement is the basic measure to ensure adaptation, control, and order [9].

Among these classic controls, there is the PID (Proportional-Integral-Derivative) control which has proven itself in the field of automatic regulation and gives good results, thanks to the proportional action that improves speed, the integral for precision, and the derivative for stability.

The design and tuning of PID correctors has been a very interesting research topic since Ziegler and Nichols proposed their tuning method in 1942 [9] . Although there are several ways to tune the parameters of a PID controller, there is an ongoing research effort to come up with new tuning techniques. These are designed to make the system accurate and undisturbed. These are satisfactory methods for controlling linear systems. Over time, these techniques will become less effective, especially when the process to be ordered has a complex and non-linear structure. This can affect the operation of the system and lead to low robustness in the face of fluctuations in these parameters and significant overruns during transition arrangements. Therefore, there are several reasons for developing smart meta-heuristic optimization methods for tuning PID controllers.

A genetic algorithm (GA) is an optimization algorithm aimed at solving difficult optimization problems (often from the fields of operations research, engineering or artificial intelligence) for which no more effective conventional method is known.

In this context, the main objective in this thesis is to apply a control technique, it is the PID regulator optimized by GAs, to the DC Motor.

In order to achieve this objective, we wanted to organize this brief into **Three** separate chapters, namely:

- **The first chapter:** is devoted to the presentation of the different structures of the PID regulator, the different performance criteria as well as some of the classic methods of adjusting the control loops such as the Ziegler-Nichols method.
- **The second chapter:** overview to the mathematical modeling and the structure of the DC motor followed by a brief overview of the methods used to control the speed of a DC motor.
- **The third chapter:** presents the new optimization technique (GA), and its principle of operation, then controlling the speed of DC motor by PID regulator under the SIMULINK/MATLAB environment.

Chapter I : Classical techniques of regulator synthesis

I.1.Introduction

The PID regulator, also called PID corrector is a control system for improve the performance of a Closed Loop (CL) system servo. That last is the most used in the industry where its correction qualities apply to multiple physical quantities.

In addition, the classic PID-based controller offers the advantage of being easy to manipulate thanks to the simplicity of its design, it does not require a precise knowledge of the model and simple to implement in real time, for this several research works have based on this modest type of controller. This regulator allows with the help of its three parameters improve performance (damping, response time, ... etc.) of the process under control [1].

In this chapter, we introduce a generality about the PID regulator, and then we Let's then start with the presentation of the different control techniques, then we let's study the most familiar methods to improve servo performance a control system.

I.2.Principle of regulation

In most industrial and domestic appliances, it is important to maintain from physical quantities to imposed values, despite the presence of disturbances external/internal affecting these quantities. For example, the water level in a reservoir, the temperature of an oven, the speed and position of the motors, being by nature variable, must therefore be addressed through appropriate actions on the process under consideration [2].

If the disturbances affecting the quantity to be controlled are slow or negligible, a simple setting called Open Loop (OL) allows to maintain the requested value (by example: action on a water tap). However, in the majority of cases, this type of setting is not sufficient, which requires us to compare at every moment, the measured value of the magnitude set to that which one wishes to obtain and act accordingly on the magnitude of action, known as the regulating quantity [3].

The purpose of regulation is to maintain at a desired value (reference quantity), a physical quantity (set quantity) such as temperature, relative humidity, pressure... subject to disruption. After comparison between set quantity and quantity reference, this results in a deviation of adjustment, and depending on this deviation, the PID regulator forms a control signal (adjustment quantity) that will vary the power of adjustment via actuator (adjustment device) [4].

I.3.Open loop (OL) system

A system is said to be in OL when the order is developed without the help of theknowledge of output quantities, as shown in the diagram given in figure (1).

Figure I -1 : Block diagram of an open-loop system.

In the latter case, the behavior of the process is described by the following relationship:

$$S(s) = G(s) \times E(s) \quad (I - 1)$$

With:

- S: set quantity (output);
- E: adjusting size (input);
- G: transferfunction.

Among the disadvantages of the OL system [5]:

- There is no way to control, let alone compensate for errors,drifts, accidents that may occur inside the loop;
- There is no precision or especially fidelity that depends on qualityintrinsic components;
- The OL system does not compensate for disturbance signals.

I.4.Closed loop system (CL)

The closed loop (counter-reaction) is able to stabilize an unstable system by open loop. In a regulation in CL, a good part of the disruptive factors external are automatically compensated by the CL (counter-reaction) through the process.

The use of feedback is the fundamental principle in automation. The Control applied to the system is elaborated according to the setpoint and the output. The Figure (2) illustrates the principle of a unit return CL system [5].

Among the advantages of ordering in CL:

- Reduce errors by automatically adjusting the system input,
- Increase or decrease the sensitivity of systems,

- Strengthen robustness against external disruptions to the process,
- Produce reliable and reproducible performance.

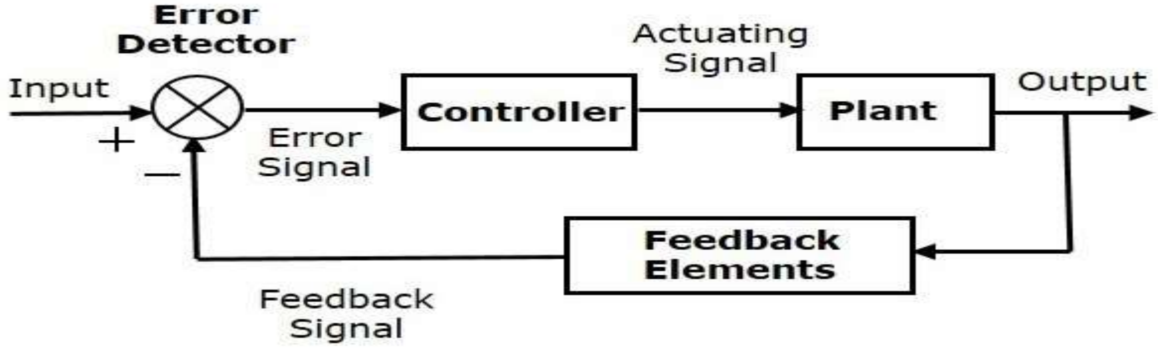


Figure I-2 : Block diagram of a system in CL with unit return

In the latter case, the behavior of the process is described by the relationship:

$$H_{BF}(s) = \frac{G(s)}{1+G(s)} \quad \text{(I - 2)}$$

With:

- ✍ G(s) represents the system transfer function in OL,
- ✍ HBF: Transfer function in CL;
- ✍ E: adjusting size (setpoint);
- ✍ S: set quantity;
- ✍ E: error (e = E(s) − S(s)).

I.4.1.Structure of PID regulators

The basic actions of a PID regulator can be combined with several structures. PID parameter values do not give the same process behavior depending on whether the structure is parallel or mixed [6]. That is why it is interesting for the automatician to know the different existing structures of the classic PID regulator.

For deferential structures we can take the notations

✓ X(s) : control set signal;

✓ Y(s) : adjusting size;

✓ e(s) : measurement/set deviation;

✓ W(s) : set quantity (measurement).

I.4.2.Parallel structure:

The parallel structure is illustrated in Figure (3). In this case the output Y(t) is given by [7]:

$$Y(t) = K_p e(t) + \frac{1}{T_i} \cdot \int_0^t e(t) \cdot d\tau + T_d \cdot \frac{de(t)}{dt} \quad \text{(I - 3)}$$

Considering the initial conditions are zero and applying the transform of the place to equation (3), we can get the transfer function of the PID regulator to parallel structure.

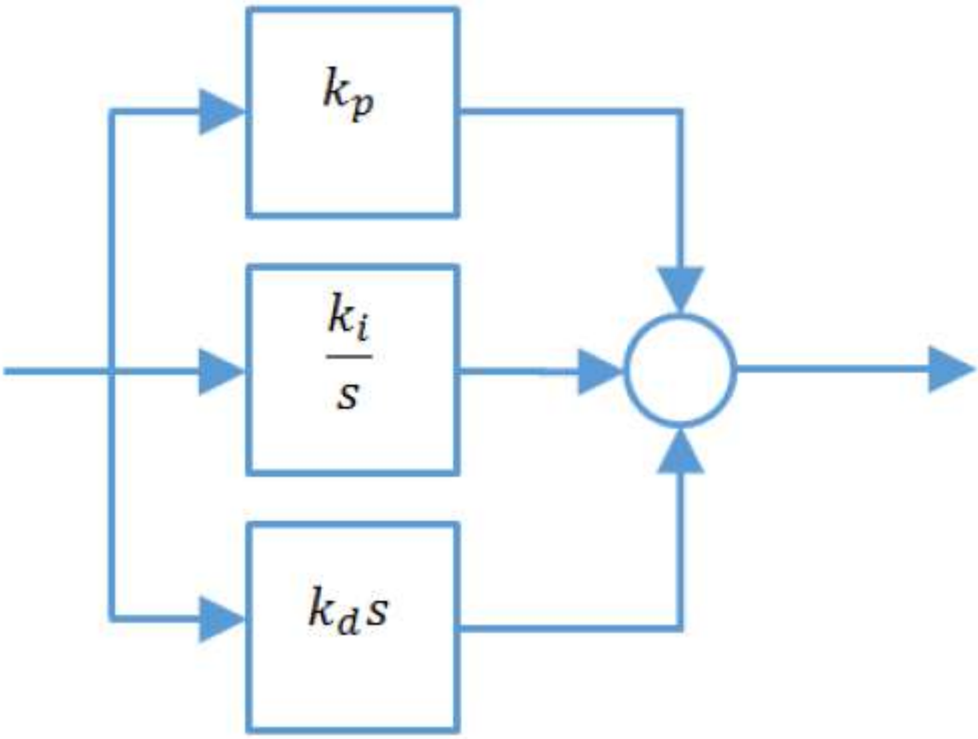


Figure 0-3 : PID regulator with parallel structure.

$$C(s) = \frac{Y(s)}{E(s)} = k_p + \frac{1}{T_i s} T_d \cdot s \quad \text{(I - 4)}$$

Or $T_i = \frac{1}{K_i}$ and$T_d = K_d$ (I - 5)

With, Ti and Td are respectively, the integration constant and the constant of derivation.

I.4.3.Serial structure

This structure is shown in Figure (4). In this case the output Y(t) is given by the following equation:

$$Y(t) = \alpha \cdot K_p \cdot e(t) + \frac{K_p}{T_i} \cdot \int_0^t e(t) \cdot d\tau + T_d \cdot \frac{de(t)}{dt} \qquad \text{(I - 6)}$$

With: $\alpha = \frac{T_i + T_d}{T_i}$ is the theoretical coefficient of interaction between integral action and derived action.

If we assume that the initial conditions are zero and applying the transformed from place to equation (6), we obtain the transfer function of the PID regulator with serial structure.

$$C(s) = \frac{Y(s)}{E(p)} = k_p \left(1 + \frac{1}{T_i s}\right)(1 + T_d \cdot s) \qquad \text{(I - 7)}$$

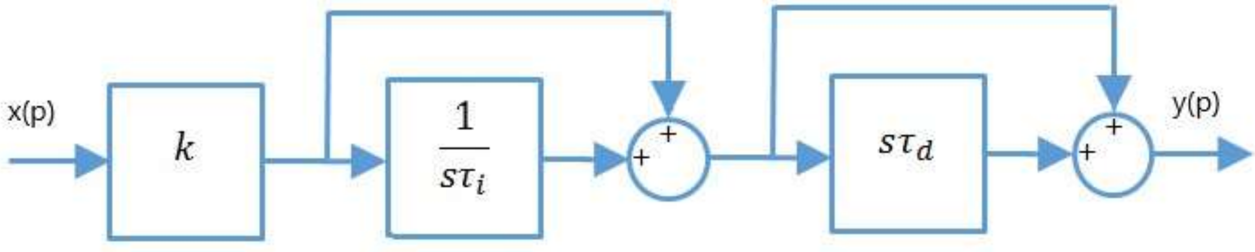


Figure 0-4 : PID regulator with serial structure.

I.4.4.Mixed structure

This structure is currently the most used by manufacturers [6], as shown by the block diagram in Figure (5).

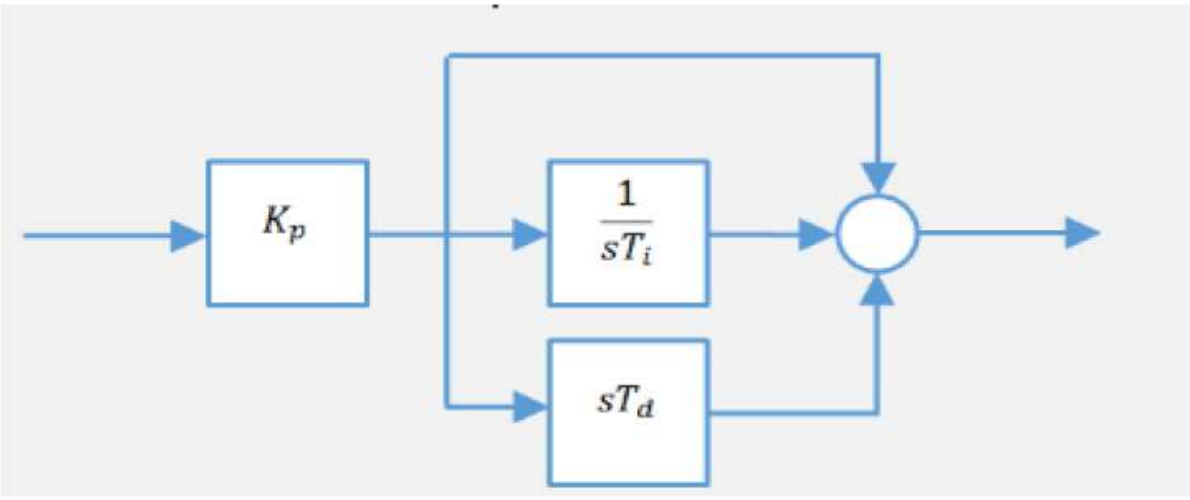


Figure I-5 : Mixed structure PID regulator.

The expression of the output of the PID regulator is given by:

$$Y(t) = K_p \cdot e(t) + \frac{K_p}{T_i} \cdot \int_0^t e(t) \cdot d\tau + T_d \cdot \frac{de(t)}{dt} \quad (I - 8)$$

By applying the Laplace transform, we obtain:

$$C(s) = \frac{Y(s)}{E(s)} = k_p \left(1 + \frac{1}{T_i s} + T_d \cdot s\right) \quad (I - 9)$$

I.5.REGULATION OF THE PARAMETERS OF A PID

Adjusting the parameters of the PID controller includes acting on the 3 parameters (proportional gain, integral gain, differential gain) of different actions to optimal values in order to obtain the correct response in terms of accuracy, speed, stability and robustness in the process. output. There are various techniques for setting optimal values in the scientific literature, such as:

- Ziegler-Nichols methods (open loop and closed loop) [8],
- the method of Broïda, Chien-Hrones-Reswick,
- the Cohen-Coon method,

I.5.1.Ziegler-Nichols method :

Ziegler and Nichols in 1942 proposed two experimental classical methods to determine and quickly adjust the parameters of PID controllers [8]. The latter is common. They often form the basis of manufacturers' PID controllers and tuning programs used by industrial processes. The Ziegler-Nichols method is based on the determination of certain characteristics of dynamic processes. The parameters of the PID controller are then functionally represented by a simple formula [8]. It is surprising that Ziegler-Nichols methods are so widely used because they only provide good tuning results in limited cases [7].

1. Reaction curve method

The method is based on modeling the exponential response of the process in CL using only simple processes. The principle is to record the response curve of the uncontrolled system in one step, and then take the system offline by analyzing the response (i.e. "graphical readout") to derive coefficient values. For this reason, the latter is not widely used in industry [7].

2. **Oscillation method:**

The oscillation method is an empirical method widely used by industry or automation specialists for setting idler pulleys for such chains. It has the advantage of not requiring precise modeling of enslaved systems, but includes experimental testing, making this approach very simple [7].

I.6.Advantages and disadvantages of PID regulators

Today, servo control using conventional PID regulators is one of the most used for the following reasons:

- it is very easy to implement it in real time;
- it is very effective for most real systems;
- the calculation of the coefficients is easy especially by the advanced research in thefield of optimization by modern algorithms;
- ... Etc.

However, it is important to note that PID regulator-based servo control is limited by a number of constraints such as:

- it may be ineffective for some systems that contain noise (derived coefficient) or which are not linear (PID servo being linear, the non-linearity of a system leads to instabilities);
- it is possible to optimize the response of a system by multiplying the servo (such as double PID servo).

I.7.Optimization by the Ziegler and Nichols method

Of all the proposed methods for computationally tuning PID controller parameters, the best documented is still the one proposed by Ziegler and Nichols in November 1942 [9]. It is based on reactive observations of the process and knowledge of the controller structure used. From this point, three variants have been proposed, the first for closed-loop adaptation, the other for open-loop adaptation, and the last for frequency adaptation [13].

I.7.1.Critical gain method

This approach requires the system to shut down to a simple proportional controller whose gain is increased until the system continues to oscillate (Figure I.6); we are at the limit of system stability.

After noting the critical gain Kcr and the response period Tcr, the selected controller parameters can be calculated using Table (1).

The values suggested by Ziegler and Nichols have been tested in many situations and it should be emphasized that they also result in relatively short climb times and high overtaking values.[13]

Table I-1 : Adjustment of regulator gains P, PI and PID according to the first method

Type	PID series	PID parallel	PID mixed
K_p	$0.3 * k_{cr}$	$0.6 * k_{cr}$	$0.6 * k_{cr}$
T_i	$\frac{T_{cr}}{4}$	$\frac{Tcr}{1.2 * k_{cr}}$	$\frac{T_{cr}}{2}$
T_d	$\frac{T_{cr}}{4}$	$\frac{Tcr * kcr}{13.3}$	$\frac{T_{cr}}{8}$

by Ziegler and Nichols. [13]

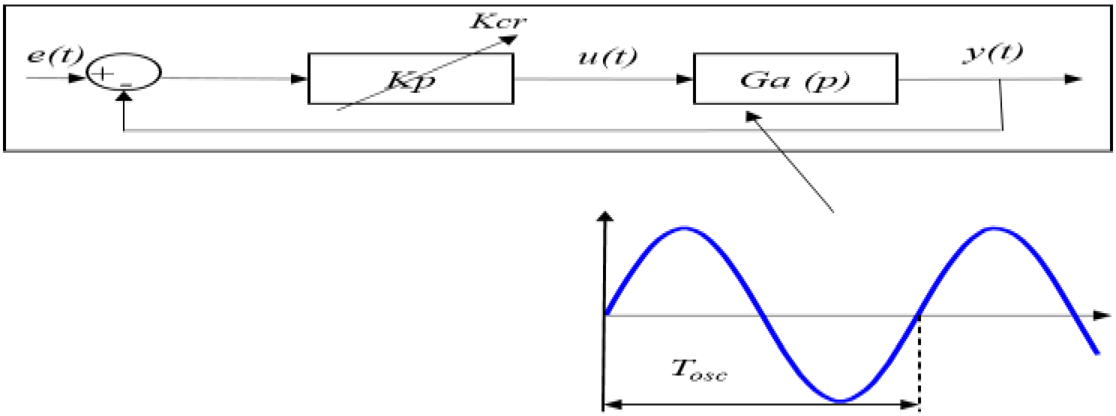


Figure I-6 : Critical gain method. [13]

Since this situation is not always satisfactory, the suggested coefficients have been slightly revised. Possible changes are suggested in Table 2. It should be noted that the Ti and Td parameters proposed in the Ziegler and Nichols method are both fixed at a constant ratio of 4.

Table I-2 : Ziegler and Nichols adjustment by critical gain method. [13]

Paramètres / Types de Régulateur	Kp	Ti	Td
P	0,4*Kcr	-	-
PI	0,4*Kcr	0,4*Tcr	-
PID	0,4*Kcr	0,4*Tcr	0,1*Tcr

I.7.2.Index response method

Most methods of calculating fit parameters involve calculating the response curve of the open-loop process after applying a step (Figure I.7). under any circumstances. We consider the linear region or response curve representing the maximum slope, we draw a line "grafting" this linear region, and we are interested in the intersection of this line with the x-axis (time axis): we thus define the time Di . Then, we Define the time Tg as the time it takes for the controlled variable to change at the same magnitude as the controller output, at the maximum rate of change (so it is necessary to answer the previously drawn line). These two parameters are then used to define the parameters for setting the regulator. [13]

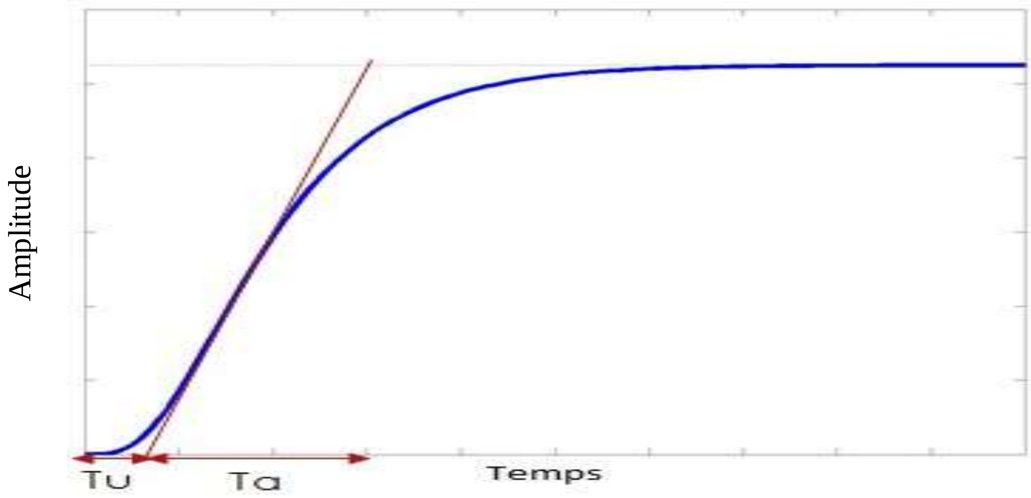


Figure I-7 : Index response of the system to be set alone: we measure the times Tu and Tg.

We can then calculate the coefficients of the chosen regulator using Table 3:

Table I-3 : Index response method. [13]

Type	PID série	PID parallèle	PID mixte
K_p	$0.6*T_a/T_u$	$1.2*T_a/T_u$	$1.2*T_a/T_u$
T_i	T_u	$1.67*T_u^2/T_a$	$2*T_u$
T_d	T_a	$0.6*T_a$	$T_u/2$

I.7.3.Advantages and disadvantages of the Ziegler and Nichols method

The advantages :

- ✓ Easy to implement (physically and from a computational point of view)
- ✓ Tested on the system in production, corresponds to reality, can be done on the fly if the characteristics of the system are modified (wear, change of the environment).

negatives :

- × The system may become unstable or pass into dangerous states (chemistry)
- × May take a long time if the system reacts very slowly (days, weeks in chemistry).
- × Fortunately for the systems that concern us (motors), it is not a problem if we limit the voltages/intensities of supply. [14]

I.11.Conclusion

In this chapter, we have given a general idea about the PID regulator and its different terms. Subsequently, then we presented some methods existing in the literature for adjusting PID regulator parameters, After we have presented examples of the application of optimization methods (Z-N), for the synthesis of an optimal PID regulator.

Chapter II : Modeling and control of a DC motor

II.1.Introduction :

DC motors are the important machine in the most control systems such as electrical systems in homes, vehicles, trains, and process control. It is well known that the mathematical model is very crucial for a control system design. For a DC motor, there are many models to represent the machine behavior with a good accuracy. However, the parameters of the model are also important because the mathematical model cannot provide a correct behavior without correct parameters in the model. In this paper study the characteristics of PID controller and its application to an industrial DC motor at steps included Structure, characteristic and the mathematical model .

II.2.Principle of operation:

DC motors work by conducting current in conductors within a magnetic field. There was a force on the ladder. When one set of these conductors is connected to a rotating anchor, all the forces on the individual conductors create a net torque.

A schematic diagram of this principle is shown in Figure II.1.

Most power sources are AC power (except for special applications of DC power such as railways). Therefore, in order to run a DC motor, the AC source must be rectified (and possibly conditioned) to provide the necessary DC voltage to power the motor.

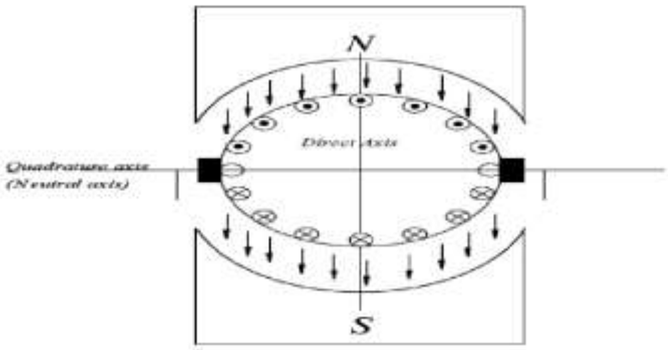


Figure I0-1 : Principle of operation of the DC motor

On Figure II.1 two axes are shown: the direct axis, which is the direction of the 24 magnetic field and the quadrature axis, which is perpendicular to it. The latter is sometimes referred to as the neutral axis. The neutral axis is the position at which the direction of current flow in the conductors reverses.

All the conductors are connected to the commutator segments on the end of the armature. Fixed sets of brushes make electrical contact with the commutator segments. The purpose of the commutator is to ensure that the direction of current flow in the conductors reverses when passing through the neutral axis. The position of the brushes has to coincide with the neutral axis.

The windings on the armature are designed as distributed windings. The armature has a certain number of slots, and each slot can usually hold two layers of conductors. It is common practice to insert each coil into a conductor with one conductor in the top slot and the other conductor in the opposite slot at the bottom. So for each coil, one side of it is below the north pole of the field winding and the other side is below the south pole. Since the current flows in opposite directions in each conductor, the force acting on each conductor

conductor lead to a torque, as shown in Figure II.2.

The left hand rule can be used to find the direction of the resultant force on the conductor. The left hand rule states that if the index of the left hand points in the direction of the magnetic flux and the middle finger points in the direction of the electric current, the thumb point in the direction of the resultant force ,this is shown Figure II.3.

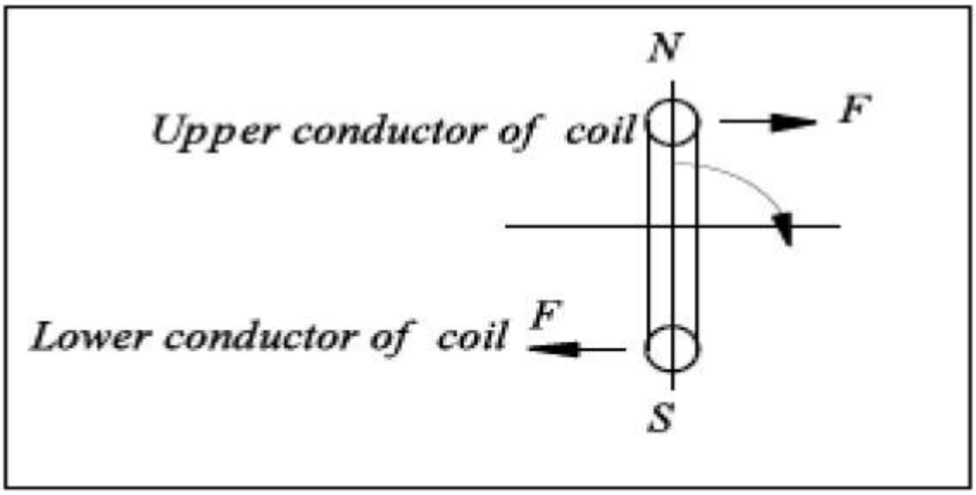


Figure II-2 : Forces on two opposite conductors of the same coil.

Figure II-3 : The left hand rule for finding the direction of force.

In order to be able to visualise the connections on the armature, it is customary to “open and spread” the armature flat. This type of connection is called lap winding, because coils actually overlap. The Figure shows a 12 slot armature, and 12 coils inserted in it. As each slot takes two conductors, each coil has one conductor in the top of one slot, and the second conductor in the bottom of the opposite slot. The magnetic fields are shown by two rectangles, labelled North and South. Notice, that for the machine to work properly, all conductor under the North pole should have their currents flowing in the same direction, while all conductors under the South pole should have their current flowing in the opposite direction to the those under the North pole. This leads to the addition of the torques from all conductors. The purpose of the commutator segment is to ensure that the current reverses in the relevant conductors at the correct point in time. The brushes complete the connection between the terminals and the segments.

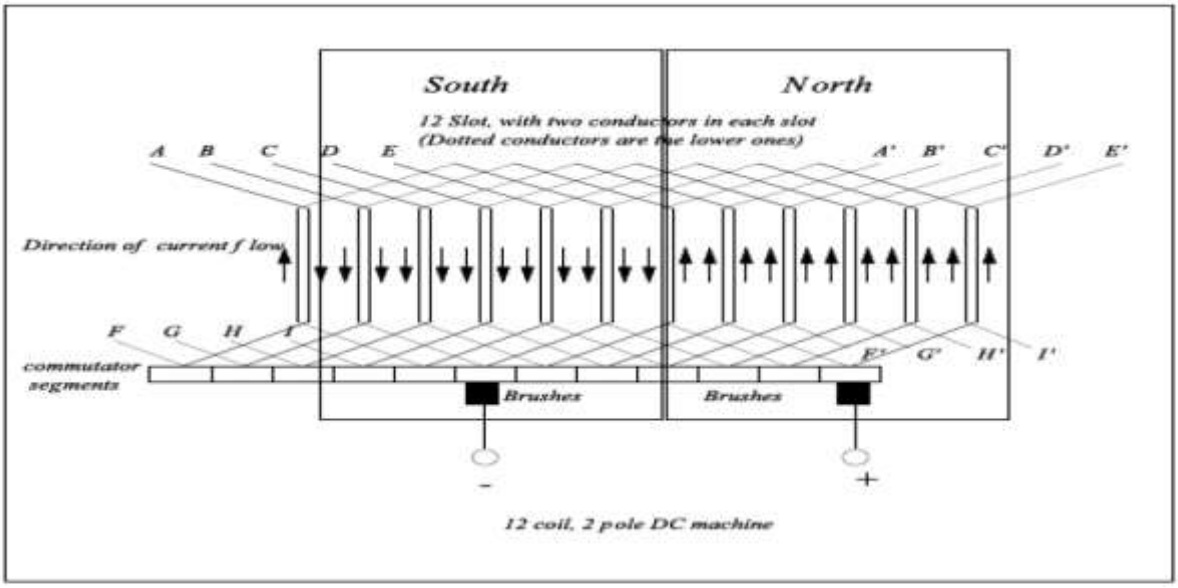


Figure 0-4 : Winding diagram for a 12 coil, 12 slot 2 pole machine.

II.3.Construction :

A DC motor consists of one set of coils called the armature winding, another set of coils or a set of permanent magnets called the stator. Applying voltage to the coil creates torque in the armature, which causes movement.

II.3.1.Stator :

- The stator is the stationary outside part of a motor.
- The stator of a permanent magnet dc motor is composed of two or more permanent magnet pole pieces.
- The magnetic field can alternatively be created by an *electromagnet*. In this case, a DC coil (field winding) is wound around a magnetic material that forms part of the stator.

II.3.2.Rotor :

- The *rotor* is the inner part which rotates.
- The rotor is composed of windings (called armature windings) which are connected to the external circuit through a mechanical commutator.
- Both stator and rotor are made of ferromagnetic materials. The two are separated by air-gap.

II.3.3.Winding :

A winding is made up of series or parallel connection of coils.

- Armature winding - The winding through which the voltage is applied or induced.

- Field winding - The winding through which a current is passed to produce flux (for the electromagnet).
- Windings are usually made of copper.

II.4.Mathematical model of dc motor:

A common actuator in a control system is a DC motor. It provides rotational motion directly and is coupled to wheels or drums, and cables can provide transitional motion. The circuit of the armature and the free body diagram of the rotor are shown in the following figure: 2.5 Mathematical model of DC motor The common actuator in the control system is the DC motor. It provides rotational motion directly and is coupled to wheels or drums, and cables can provide transitional motion. The armature circuit and rotor free body diagram are shown below:

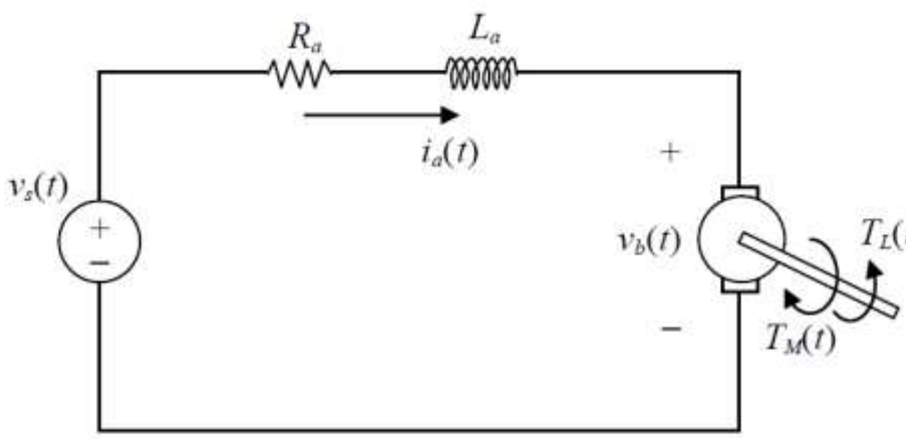


Figure II-5 : DC motor equivalent circuit

The torque is proportional to current as a following.

$$T = K_t i \quad (II - 1)$$

The back emf e, is related to the rotational velocity by the following equation:

$$e = K_e \dot{\theta}(t) \quad (II - 2)$$

In SI units (which we will use), Kt (armature constant) is equal to Ke (motor constant). From the figure above we can write the following equations based on Newton's law combined with Kirchhoff's law can be written as follows :

$$J.\ddot{\theta}(\mathrm{t}) + b.\dot{\theta}(\mathrm{t}) = K_e \cdot i(t) \quad (II - 3)$$

$$L \cdot \frac{di(t)}{dt} + R \cdot i(t) = V(t) - K.\dot{\theta}(t) \qquad (II - 4)$$

Using Laplace Transforms, the above modeling equations can be expressed in terms of operator(s)

$$(Js + b) \cdot \theta(s) = K \cdot I(s)$$
$$(Ls + R) \cdot I(s) = V(s) - K \cdot \theta(s) \qquad (II - 5)\ (II - 6)$$

From equation (5):

$$I(s) = \frac{(Js+b)\cdot\theta(s)}{K} \qquad (II - 7)$$

Substitute equation (7) into equation (6) to get:

$$\frac{(Ls+R)(Js+b)\cdot\theta(s)}{K} = V(s) - K.\theta(s) \qquad (II - 8)$$

From equation (8):

$$\frac{\theta(s)}{V(s)} = \frac{K}{(Ls+R)(Js+b)+K^2} \qquad (II - 9)$$

The DC motor parameters are given in table 1:

Table0-1 : DC Motor parameters.

Parameters	values
R	5.2 Ω
L	1.1mH
J	0.01 kg.m
B	0.02 N.m.sec/rad. s^{-1}
K	0.75 N.m.A^{-1}

II.5. Speed Control Methods Of DC Motor:

Speed Control Of Shunt Motor:

1. Flux Control Method:

The **speed of a dc motor** is inversely proportional to the flux per pole. Thus by decreasing the flux, speed can be increased and vice versa.

To control the flux, a rheostat is added in series with the field winding, as shown in the circuit diagram. Adding more resistance in series with the field winding will increase the speed as it decreases the flux. In shunt motors, as field current is relatively very small, I_{sh}^2R loss is small. Therefore, this method is quite efficient. Though speed can be increased above the rated value by reducing flux with this method, it puts a limit to maximum speed as weakening of field flux beyond a limit will adversely affect the commutation.

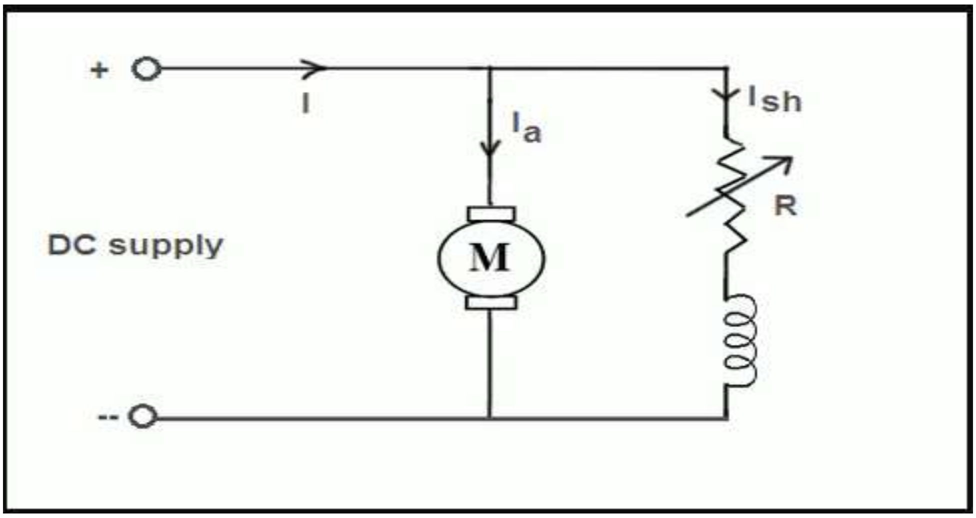


Figure II - 6: flux control

2. Armature Control Method

Speed of a dc motor is directly proportional to the back emf E_b and $E_b = V - I_aR_a$. That means, when supply voltage V and the armature resistance R_a are kept constant, then the speed is directly proportional to armature current I_a. Thus, if we add resistance in series with the armature, I_a decreases and, hence, the speed also decreases. Greater the resistance in series with the armature, greater the decrease in speed.

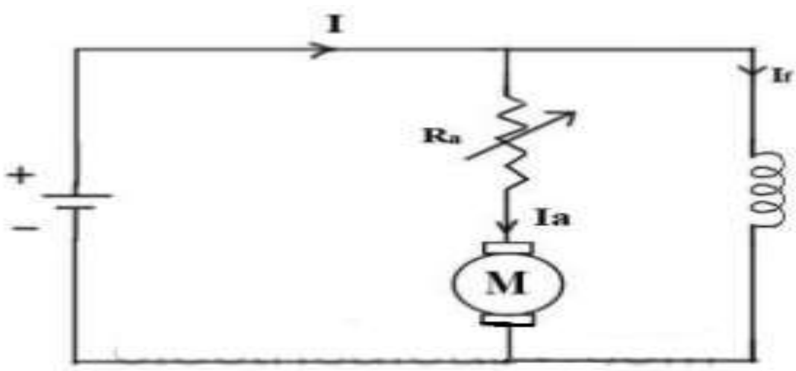


Figure II - 7: armature control

3. Voltage Control Method

a) Multiple voltage control:

In this method, the shunt field is connected to a fixed exciting voltage and armature is supplied with different voltages. Voltage across armature is changed with the help of suitable switchgear. The speed is approximately proportional to the voltage across the armature.

b) Ward-Leonard System:

This system is used where very sensitive **speed control of motor** is required (e.g electric excavators, elevators etc.). The arrangement of this system is as shown in the figure at right.

M_2 is the motor to which speed control is required.

M_1 may be any AC motor or DC motor with constant speed.

G is a generator directly coupled to M_1.

In this method, the output from generator G is fed to the armature of the motor M_2 whose speed is to be controlled. The output voltage of generator G can be varied from zero to its maximum value by means of its field regulator and, hence, the armature voltage of the motor M_2 is varied very smoothly. Hence, very smooth **speed control of the dc motor** can be obtained by this method.

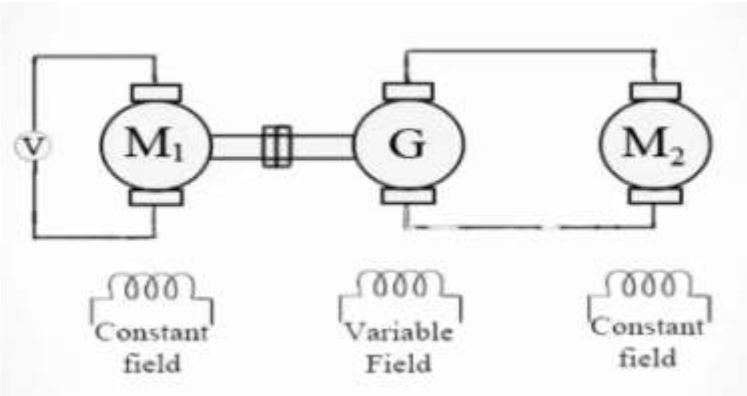


Figure II - 8: voltage control

Speed Control Of Series Motor

1. Flux Control Method

- Field diverter: A variable resistance is connected parallel to the series field as shown in fig (a). This variable resistor is called as a diverter, as the desired amount of current can be diverted through this resistor and, hence, current through field coil can be decreased. Thus, flux can be decreased to the desired amount and speed can be increased.
- Armature diverter: Diverter is connected across the armature as shown in fig (b).
  For a given constant load torque, if armature current is reduced then the flux must increase, as
  $T_a \propto \varnothing I_a$
  This will result in an increase in current taken from the supply and hence flux Ø will increase and subsequently **speed of the motor** will decrease.
- Tapped field control: As shown in fig (c) field coil is tapped dividing number of turns. Thus we can select different value of Ø by selecting different number of turns.
- Paralleling field coils: In this method, several speeds can be obtained by regrouping coils as shown in fig (d)

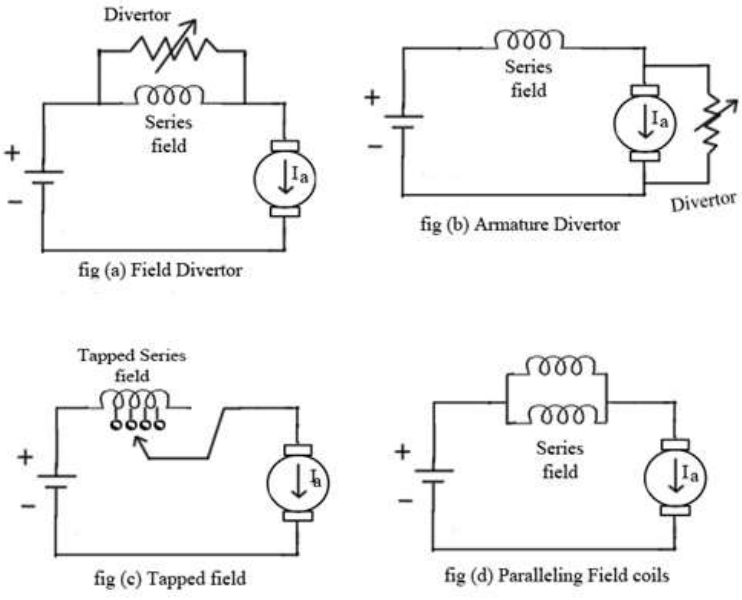


Figure II - 9: series motor

2. Variable Resistance In Series With Armature

By introducing resistance in series with the armature, voltage across the armature can be reduced. And, hence, speed reduces in proportion with it.

3. Series-Parallel Control

This system is widely used in electric traction, where two or more mechanically coupled series motors are employed. For low speeds, the motors are connected in series, and for higher speeds, the motors are connected in parallel.
When in series, the motors have the same current passing through them, although voltage across each motor is divided. When in parallel, the voltage across each motor is same although the current gets divided.

II.6.PID controller :

PID controllers have a long history in control engineering and they have been proven to be robust, simple and stable for many real world applications. Roughly, P action is related to the present error, I action is based on the past history of error, and D action is related to the future behavior of the error. From estimation point of view, P, D actions, and I correspond to filtering, smoothing and prediction problems respectively. This type of neuro-controller is meaningful from a viewpoint of application because PID controllers are so widely used today. The equation of a PID controller is given by :

$$u(t) = K_p e(t) + K_i \int_0^t e(t)dt + K_d \frac{de(t)}{dt}$$

Where kp is proportional, ki is integral, and kdk_d is derivative gain of the controller. for this study the derivative control canceled to use only PI Controller, so the control became:

$$u(t) = K_p e(t) + K_i \int_0^t e(t)dt$$

A proportional controller (k_p) will have the effect of reducing the rise time and will reduce but never eliminate the steadystate error. An integral control (k_i) will have the effect of eliminating the steady-state error.

II.7.Conclusion:

In this chapter, we will give an overview of a DC motor and how it works, then analyze its design aspirations (stator and rotor windings...), then we look at the mathematical model of a DC motor, and then discuss the speed control methods of a DC motor.

Chapter III : Optimization by genetic algorithm

III.1.Introduction :

Man has always aspired to perfection. Whatever he undertakes he desires achieve all that is better by improving performance (maximization) and decreasing its errors (minimization).

"The human desire for perfection finds expression in the theory of optimization".

She studies how to describe and achieve what is better, once we know how measure and modify what is good and bad... Optimization theory includes study quantitative optimums and methods for finding them. Beightler, Philips, and Wild (1979, p.1) [10].

Thus optimization seeks to improve performance by getting as close as possible possible from one or more optimum points. It makes it possible to significantly increase the the performance of the systems on which it is applied.

These systems, whatever their nature, are often represented by equations mathematics. The purpose of optimization will therefore be to find the values of or variables that maximize or minimize these functions.

Several optimization methods have been developed, but the most efficient of them have already reached their limit. The main limitation of such methods lies in the fact that there is no guarantee that the optimum found is a global optimum and not a local optimum (Figure 3.1). Ok that they have given good results in most situations, they do not allow to arrive at a practical solution when the problems addressed reach a size and/or a significant complexity.

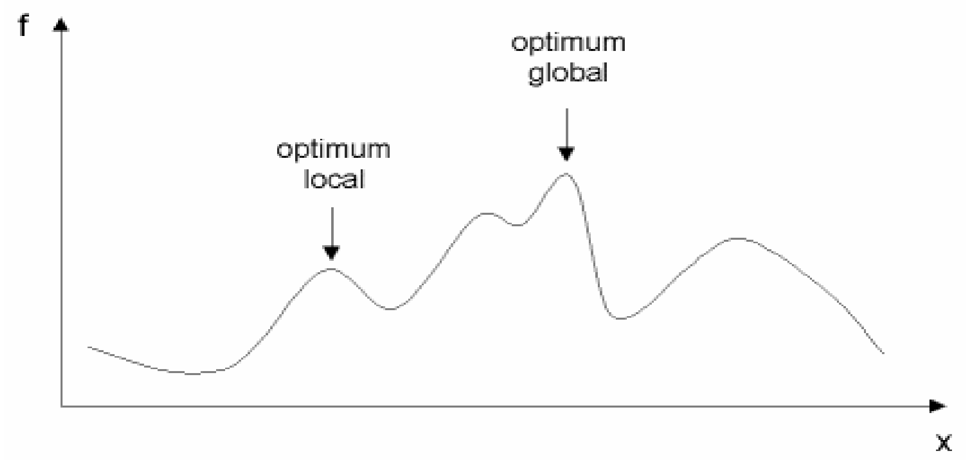


Figure III-1 : Global and local optimum.

For this reason, research has focused on methods that ensure the convergence towards the optimal solution despite the complexity and size of the problem at optimize.

The idea was to design robust research mechanisms that could adapt to any environment and constraints. When it comes to robustness, perfection and complexity nature does better than scientists do. This is why researchers have turned to adapting natural mechanisms for problem solving or improving techniques that already exist. Among these adaptations are neural networks, fuzzy logic and genetic algorithms. Genetic algorithms were inspired by natural selection, reproduction and genetics.

The mechanisms of natural selection and genetics were established by scientists Darwin and Mendel.

Darwin's observations and his theories on the evolution of living species, although reversed, however, remain well founded. Darwin explains that only the best adapted individuals, that is, able to perform tasks necessary for their survival, reproduce at higher rates while the least adapted individuals reproduce at lower rates. A population with a large variety will, from generation to generation, contain individuals whose genotype (set of characters) results in better adaptation and this because of the constraint of natural selection.

Mendel explained the mechanisms of reproduction and adaptation through crossing and mutation.

In each generation, the best individuals in a population will be selected to reproduce. Their descendants will form a new population which will in turn be subject to selection, crossing and mutation.

The scientific community could only marvel at the robustness, efficiency and flexibility of this system. Its iterative aspect, its ability to adapt and select the best elements did not leave insensitive researchers who quickly found a robust way optimization.

The genetic algorithms were developed by John Holland, his colleagues, and his student of the University of Michigan. In 1975, he published his book "Adaptation in Natural and Artificial Systems" which lays the foundations of genetic algorithms [10] [12].

Although not dependent on complex mathematical methods. Research on genetic algorithms leave the door open to all innovations and creativity Possible. Since their development they have been the subject of a large number of articles and theses. Doctorate [12] [11][10].

In this chapter, we will develop the notions of selection, crossing and mutation of genetic algorithms. We will try to design a genetic algorithm which will then be applied to the optimization of some functions. Once the algorithmgenetics validated, it will be applied to the control of our machine.

III.2.Definition of genetic algorithms:

Genetic algorithms are evolutionary optimization algorithms that look for the extreme(s) of a function defined over a given interval. Based on the Darwin's theory of evolution and the laws of genetics seen above. Those algorithms work (evolve) from how a population can evolve to the same steps: selection, crossing and mutation. Based on the information delivered by the function to be optimized, the algorithms genetics unlike other methods explore several points in space. At each generation only the points that correspond to the highest values of the function to be optimized will be selected. They will generate following the crossing and mutation new points (a new population). At each generation, an optimum is calculated.

To exploit such an algorithm, it is necessary to proceed according to the process following [10]:

III.2.1.Coding:

In this step, each point in space is associated with a data structure. This is usually done after modeling the problem to be treated.

III.2.2Choice of initial population:

The choice will be made on a heterogeneous set of individuals who will be the mother generation. That Choice is all the more important as it can make the algorithm more or less fast. The Initial population is spread over the entire field of research, in the case or problem to solve is unknown.

III.2.3.Define the function to be optimized:

This function is called fitness or individual evaluation function.

III.2.4.Defining the operators of population diversification over generations:

These operators are selection, crossing and mutation. During the selection the most suitable individuals are the only ones chosen to constitute the new generation. The operator of crossing recomposes genes and proposes new individuals. The mutation operator guarantees the exploration of the research space.

III.2.5.Define dimension parameters

The size of the population, the search interval, the probabilities of application of the crossing and mutation operators and the stopping criterion.

III.2.6.Stopping criteria:

The stop test may represent:

- The number of generations initially set.
- The value of the evaluation or fitness function has reached a value fixed a priori.
- The absence of the evolution of the value of the evaluation function of the individuals of a population to another.
- Individuals have achieved a certain degree of homogeneity.

In what follows we will go into detail about the different steps mentioned above.

III.3.Operators of genetic algorithms

III.3.1.Coding:

In the very beginning, the coding used by genetic algorithms was binary coding. That encoding is represented by a string of bits that contain the information needed for the description of a point in space. This makes it possible to create crossing and crossing operators. fairly simple mutation.

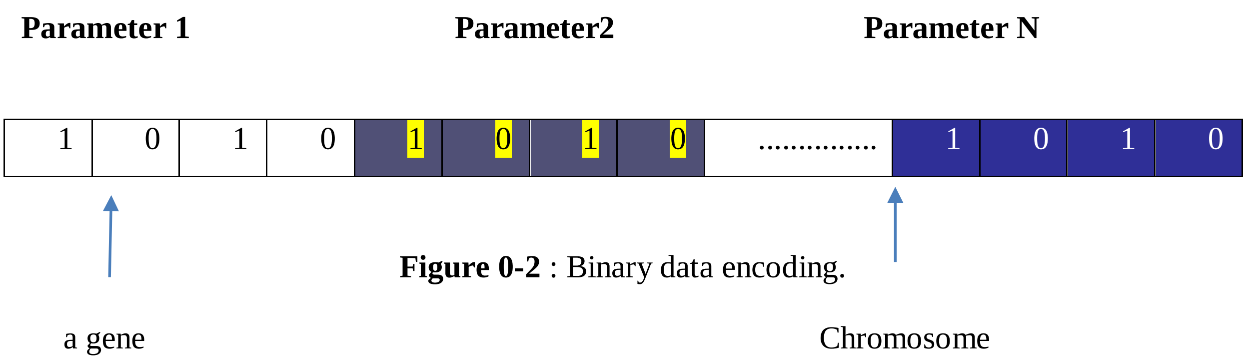


Figure 0-2 : Binary data encoding.

III.3.2.Generation of the initial population:

The speed of the algorithm depends a lot on the choice of the initial population. If the position of optimization in the state space is unknown it is normal to generate individuals randomly and exploit them so that they respect the constraints.

If the information is available, only generations are generated that respect the constraints, this accelerates convergence since there will be no step of elimination of points not respectful of constraints.

If the information is available, only generations are generated that respect the constraints, this accelerates convergence since there will be no step of elimination of points not respectful of constraints.

The diversity of a population is maintained over several generations by manipulations performed on the structure of chromosomes. This role is played by the operators of crossing and mutation.

The choice of population size is a delicate choice. For a relatively small size the algorithm evolves towards an uninteresting local optimum. For a size too high the algorithm takes longer to converge on a possible solution. The size of the population must be chosen in such a way as to achieve a good compromise between the calculation time and the quality of the result.

III.3.3.Objective function and fitness:

We call an objective function, the function we want to optimize. Fitness as for is the evaluation function of the individual. The fitness function is determined according to the problem posed (to be optimized). As part of the from a simple function optimization, the fitness function is the objective function. Fitness can be seen as a measure of profit, utility or quality.

It is used to assign to an individual a numerical value related to the interest he represents. As a solution, Individuals in a population will be selected or eliminated based on their fitness.

Only individuals with high fitness will be reproduced.

III.3.4.Selection:

Selection is used to identify the right elements of a population and to exclude them from the bad. For this, there are several selection methods.

III.3.4.1.Wheel selection:

Each individual is a chromosome. All of these chromosomes are placed on a roulette or each chromosome occupies a space proportional to its ability to adapt. Let Sf be the sum of the fitness of a population and fi the fitness of each chromosome. One defines Pi as the percentage occupied by this chromosome of the wheel pi=fi/sf Each chromosome i occupies Pi% of the roulette wheel.

Chromosomes are chosen following the launch of a ball, the chromosome designated by its stop will be selected and will participate in the training of the new generation. Assume that a chromosome occupies more than 90% of the roulette wheel, in this case it is very likely that it is the only selected which limits the evolution of the population. That's why we got to turned to another method of selection. Each chromosome i occupies Pi% of the roulette wheel.

Chromosomes are chosen following the launch of a ball, the chromosome designated by its stop will be selected and will participate in the training of the new generation. Assume that a chromosome occupies more than 90% of the roulette wheel, in this case it is very likely that it is the only selected which limits the evolution of the population. That's why we got to turned to another method of selection.

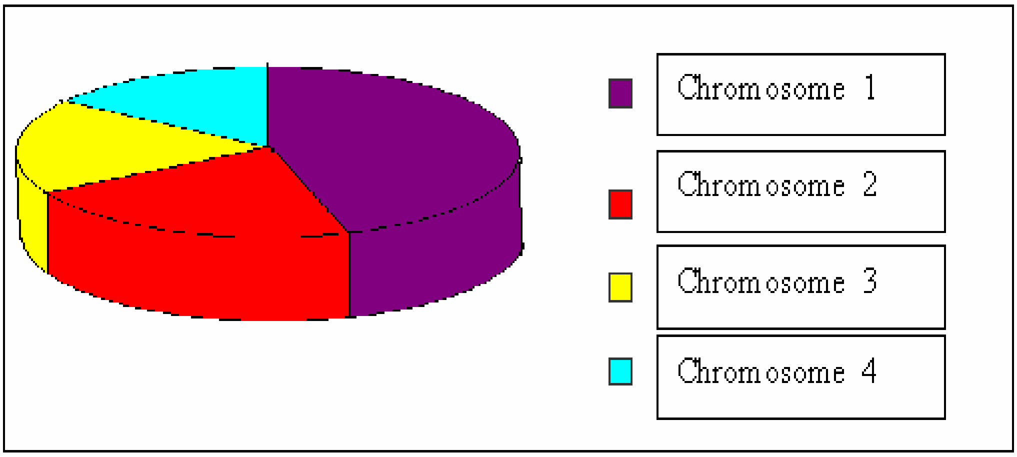


Figure 0-3 : Sélection par roulette.

III.3.4.2.Selection by rank:

This is done in the same way as for selection by roulette, except that, chromosomes are not placed on a wheel, but sorted by rank from worst to better. For an initial population of size N, the worst chromosome will be assigned to the rank 1 and best at rank N. Thus the selection will depend on the rank of the chromosome and not its ability to adapt. All chromosomes will then have a chance to be selected. That said, the selected chromosomes are not very different. Their evolution is slow. The genetic algorithm must therefore create several generations to arrive at the optimal solution.

The performance of a genetic algorithm is highly dependent on selection. The fact of create in each generation, a population made up of copies of the best individuals of the previous population, leads to faster convergence of the algorithm to the solution optimal. Given the importance of selection and the limitations of its techniques Researchers have been forced to develop more appropriate and capable of giving better results. The most interesting works are those of De Jong and Brindle [10].

III.3.4.3.De Jong selections :

De Jong combined the theories developed by Holland with his own particularly meticulous computer experiments. Knowing all the possible applications of the genetic algorithm De Jong

Particularly interested in its use in the optimization of functions. He set himself as objective, the improvement of the simple genetic algorithm, through the development of new selection methods. To this end he devised four selection methods. The three are cited as Following:

III.3.4.3.1.Elitism:

This method aims to preserve the elite of the population. When creating a new generation it is quite possible that good chromosomes are lost during the crossing and of the mutation. To remedy this, one or more of these chromosomes are copied into the new generation. Ind (t) is defined as the best individual generated by the algorithm up to the generation t. If, ind (t) is not contained in the generation (t+1) then it will be added to it.

III.3.4.3.2.Mathematical expectation method:

De Jong based his selection on the adaptability of each channel. This capacity is given by the relationship ci=fi/f Where f is the average fitness of the population.

It is assumed that all channels will be selected for the formation of the new generation. Each time a chain is selected for a crossover, a counter, which it is attached, is decremented by 0.5. If the channel is selected for reproduction without crossing its counter is decremented by1. Chains whose counter passes below zero will no longer be available for selection.

It is the combination of the previously mentioned methods, which leads to a method elitist of mathematical hope. Analysis of the results obtained by the three methods highlights the effectiveness and superiority of the second selection method.

III.3.4.4.Brindle selection :

Brindle was interested in improving the performance of the genetic algorithm by improved selection.

In his research, Brindle evokes several methods of selection, without for all that, in determine the best performer. Subsequent studies, however, demonstrated the superiority of two of them.

III.3.4.4.1.Stochastic selection for the remaining part without replacement:

The probability that an element i is selected is given by the relation:

pselect fi =f i/sf (3)

To constitute a population of size n, the number of expected copies for each element is *i i nc = n * pselect* (4)

Each element i will be granted with certainty a number of copies equal to the integer part of nci and a probability of a possible copy equal to its floating part [10].

III.3.4.4.2.Selection by tournament:

This method involves selecting one pair of individuals per lottery wheel. The individualwith the highest adaptation will be declared the winner of the tournament and will therefore be added to the new generation. This process will be repeated until the population is constituted. [10]

III.3.5.Crossing:

The crossing is the operator that ensures the exchange of information and the creation of new points. It is done with two parents and usually gives two children afterwards. Each parent is represented by a string of bits (binary encoding). Each channel is cut into one or more parts. The location of the clipping is designated by a Pc point chosen randomly. The crossing is done by exchanging these parts between the parents. We define by CP the number of crossing points. A one-point crossing is made for CP = 1. The first child consists of the first part of the first parent and the second of the second parent. The second child consists of the first part of the second parent and the second part of the first parent. The following figure shows a one-point crossing for a crossing point Pc = 3

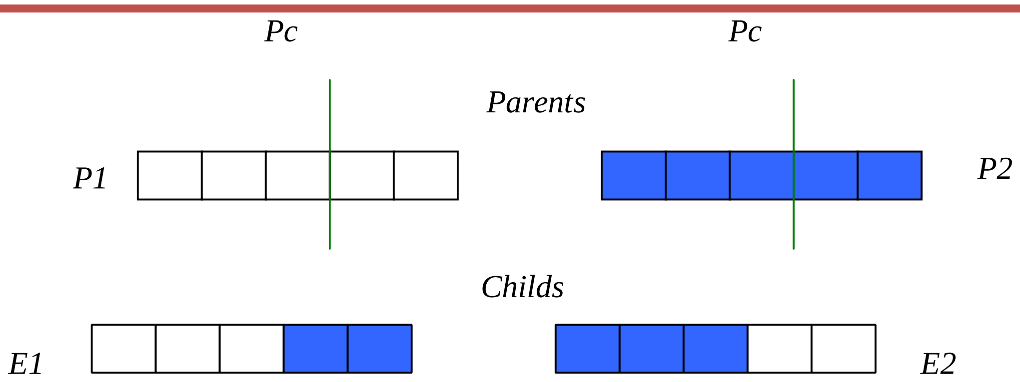


Figure III-4 : Crossing at a point.

The following figure shows a two-point crossing for the crossing points *Pc1=1 et pc2=3*

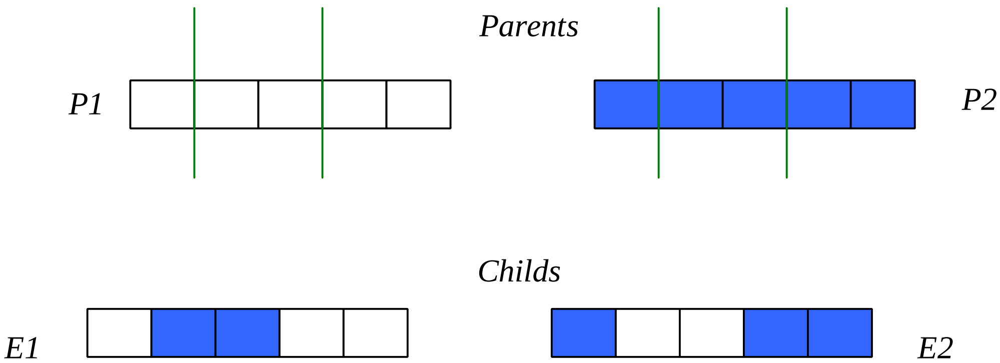


Figure III-5 : *Crossing into two points.*

During his research, De Jong deduced that the multi-point crossing decreases the performance of the algorithm especially as the number of crossing points is high [10].

III.3.5.1.Probability of crossing:

The probability of crossing Pcrois refers to the number of chromosomes (elements) at cross. For a population of size n and a probability of crossing Pcrois alone Pcrois*m elements will be paired and crossed. In most of the problems treated the probability of crossing chosen is 0.6 [10] [12].

III.3.6.Mutation:

Mutation means change or modification. In biology, it means the modification of DNA bases.

In genetic algorithms, it consists in exchanging the value of a selected bit at the chance (Figures III.6 and III.7). It ensures the diversity of the population through the exploration of new points in space.

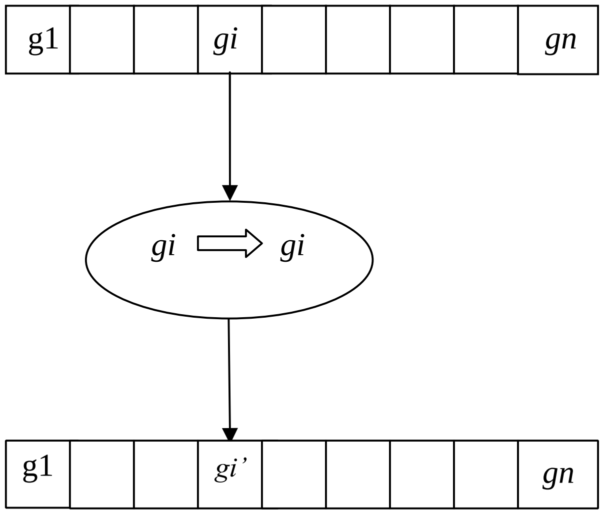


Figure III-6 : Principle of mutation.

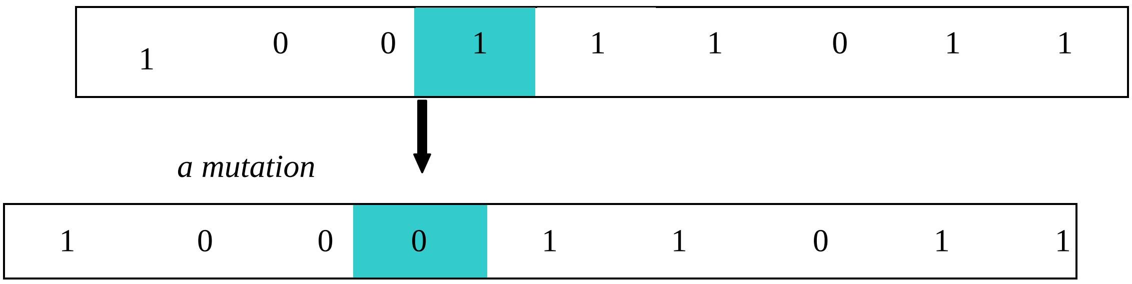


Figure III-7 : Schematic representation of a mutation in a chromosome

III.3.6.1.Probability of mutation:

The mutation probability Pmut denotes the number of bits to mutate k. Once the number is defined, k bits of the population will be chosen at random to be mutated.

Let's take the example of a population consisting of five chromosomes, where each chromosome is represented by a ten-bit chain. The total number of bits in the population will be 50 bits. For a mutation probability of 0.2 only ten bits will undergo a mutation. Their location will be chosen randomly.

110010101 0	100010110 0	001111101 1	101011101 1	101110010 0

mutation de la population

0100001010	1010100100	0110110111	0110111011	1011100100

Figure III-8 : Concept of probability of mutation.

Unlike crossing, the mutation slowly converges to the best solutions, so the probability of mutation must be small and lower than the probability of crossing. This low probability allows us to avoid premature convergence while allowing us to explore new points.

III.3.7.Stopping criterion:

Most often, a predefined number of generations is used as a stop criterion. The more complex the function to be optimized, the higher the number of generations.

III.4.Minimization by genetic algorithms :

The optimization of certain problems results not in the calculation of the maximum of a function f (x) but in that of its minimum. Genetic algorithms are optimization tools designed for the sole purpose of calculating the maximum of a function.

The minimization of a function f (x) by genetic algorithms results in the maximization of the function $g(x) = - f(x)$. If f_{max} is the maximum obtained from the function g(x) the minimum of the function f (x) will be $f_{min} = - f_{max}$

III.5.Real-coding genetic algorithms:

It is thanks to binary coding that the first convergence results were obtained. But this process knows its limits in problems of large dimensions because each point is represented by a part of the chain but the problem is not reflected. The variable order being important for a chromosome is not important for the problem [11] [10].

To avoid this disadvantage, genetic algorithms use real vector coding, this process makes it possible to keep the variables of the problem in the coding and this, without going through binary coding [11] [10].

Conventional, crossing and mutation operators, as defined above, will no longer be valid. Other crossing and mutation processes have been developed.

III.5.1.Barycentric crossing :

Bar centric crossing is used as part of an actual parameter coding. Children E1 and E2 will be trained from parents P1 and P2 as follows:

$$\begin{cases} E1=\alpha_1 P1+ (\alpha_1 -1)\, P2 \\ E2 =\alpha_2\, P2 + (\alpha_2 -1)\, P1 \end{cases} \quad (5)$$

The number α is chosen randomly. [12] [11] [10]

- If α1 = α2, the crossing is said to be symmetrical
- Siα1 = α2 = 0.5, the crossing is said to be arithmetic.
- If α1 ≠α2, the crossing is said to be asymmetrical.

III.5.2.Mutation:

For a binary encoding, we define two types of mutation

III.5.2.1.Uniform mutation :

This type of mutation is simple. The individuals to be mutated will be chosen randomly and will be replaced by randomly selected values in the search interval [11].

III.5.2.2.Non-uniform mutation:

This technique is applied according to the current generation t and the maximum number of generations gen – max. The mutated individual c′is defined as follows

$$c' = \begin{cases} c_1' = c + \Delta(t, b_i - c) \\ c_2' = c - \Delta(t, c - a_i) \end{cases}$$

We define τ as a random number such that τ= {0,1}

- Si τ= 0, $c' = c_1'$
- Si τ =1, $c' = c'_2$

With

$$\Delta(t, y) = y\left(1 - r^{\left(1 - \frac{t}{gen-max}\right)^b}\right)$$

[ai bi]: Search interval r is a random number chosen in the interval [0.1] b was estimated following experiments at the value 5 [12].

III.6.Implementation of the genetic algorithm:

Generally, the generation of the new population is done by applying the crossing and mutation operators according to their respective probability. As we have previously described a number of individuals are chosen to be subsequently matched and crossed. The mutation will be applied to the rest of the population that has not been crossed. For our algorithm we chose to proceed differently.

Among the population, the crossing will be applied only on the best individuals. Each couple will be represented by a descendant c from an arithmetic crossing. The number of offspring expected of each pair to obtain a new population (of the same size as the previous one) will be determined according to the fitness of the descendant c. once the crossing is completed we obtain a descendant population whose size is equal to that desired.

The mutation will be applied to the entire new population. If the best individual obtained after mutation has inferior fitness, the mutation is considered defective and will subsequently be eliminated. The new generation will therefore only come from the crossing of the mother population.

III.7.Validation of the method:

In the following we will test our algorithm for the optimization of two functions

1. $f(x) = x + 10\sin(5x) + 7\cos(4x)$ dans l'intervalle $[0 \quad 9]$
2. $f(x, y) = x\sin(4x) + 1.1 * y\sin(2y)$ dans l'intervalle $[-5 \quad 5]$ pour x et y

III.7.1.Optimization of the first function:

The appearance of the function f (x) is given by Figure (III.9). The maximum as well as the minimum of this function will be calculated by the genetic algorithm characterized by the following parameters:

- Size of the initial population	10
- Number of generations	40
- Stochastic selection	selection method without replacement
- Type of crossing	barycentric crossing
- Probability of crossing	0.6
- Mutation type	uniform mutation
- Probability of mutation	0.1

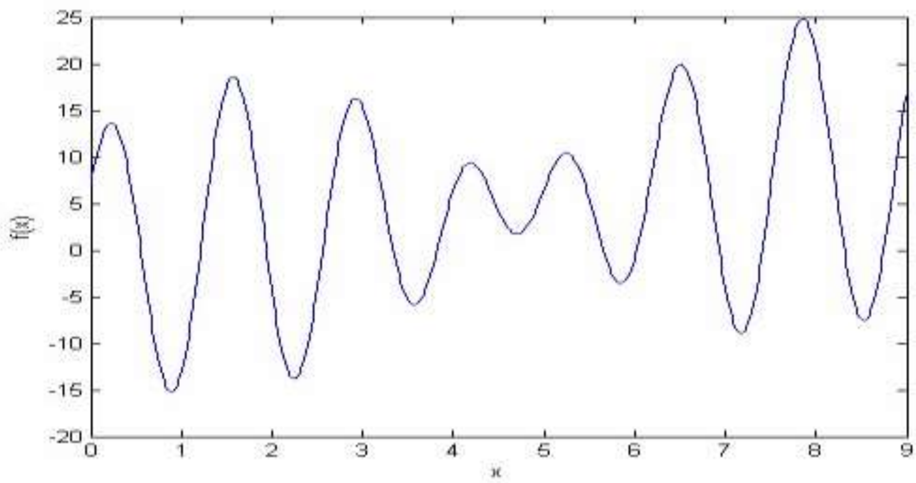


Figure III-9 : The appearance of the function to be optimized.

III.7.1.1.Maximization of f(x) :

The maximum obtained is f (x) = 24.8554 for a value of x = 7.8567.

The function f(x), the initial population generated and the result of the optimization are represented by the following figure (III.10). Figure (III.11) shows the fitness of the best individual obtained in each generation.

We notice that the genetic algorithm manages to determine the maximum of f(x) after only 14 generations

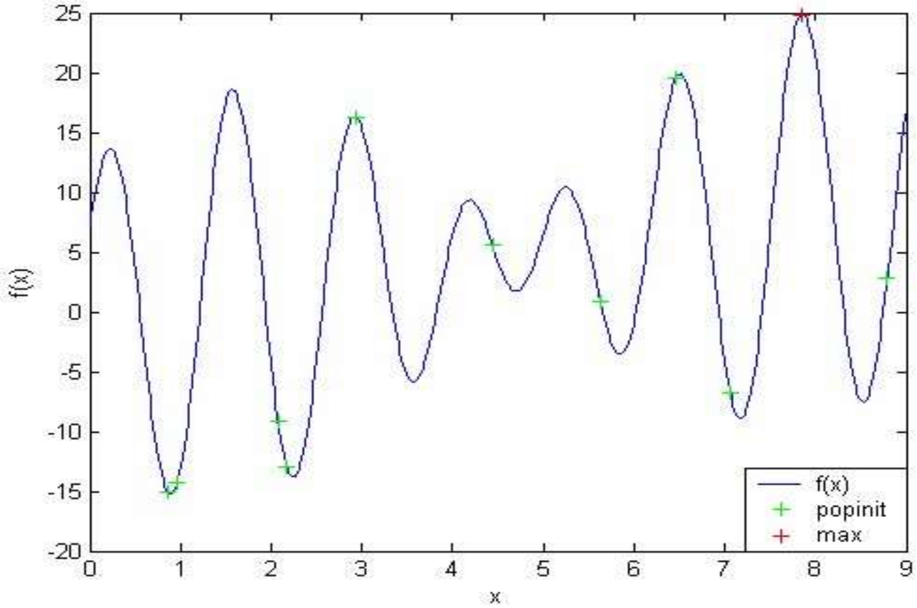


Figure III-10 : Result of optimization.

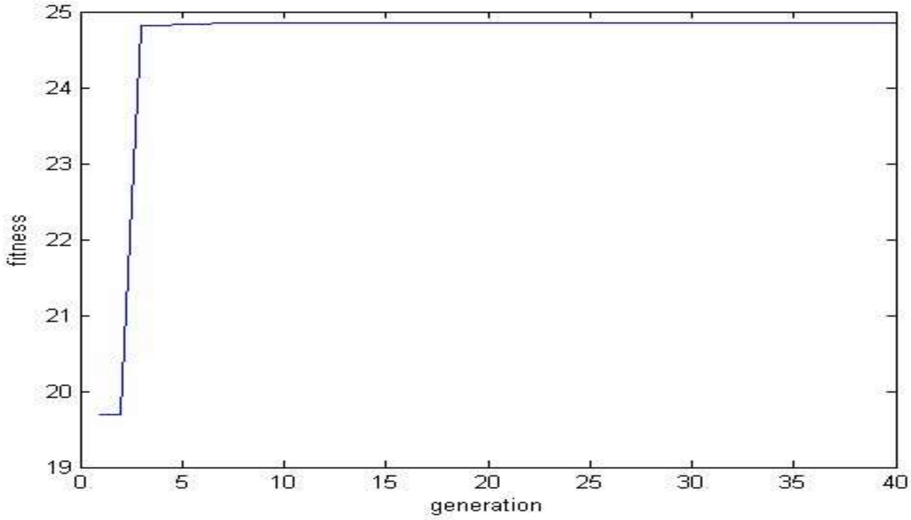


Figure III-11 : Fitness of the best solutions.

Minimization of f(x): Let be the function g(x) = − f (x). Maximizing the function g(x) gives the maximum g(x) = 15.1644 after 5 generations.

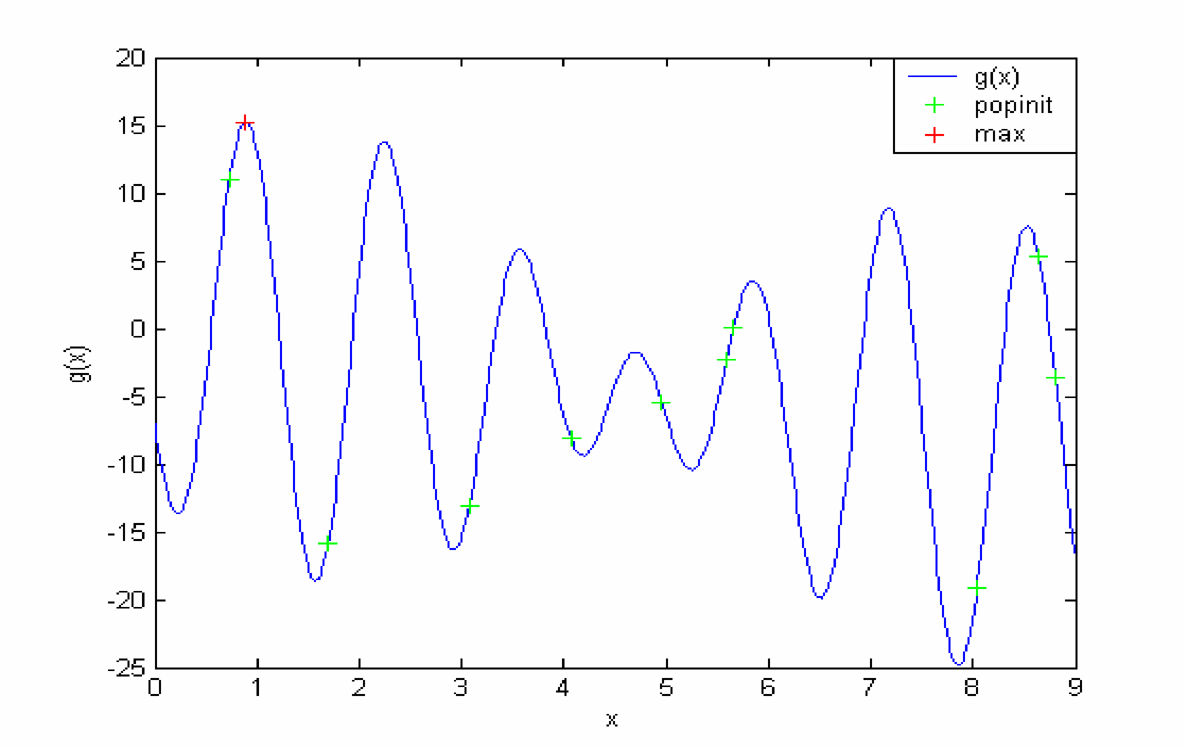


Figure III-12 : Optimization results.

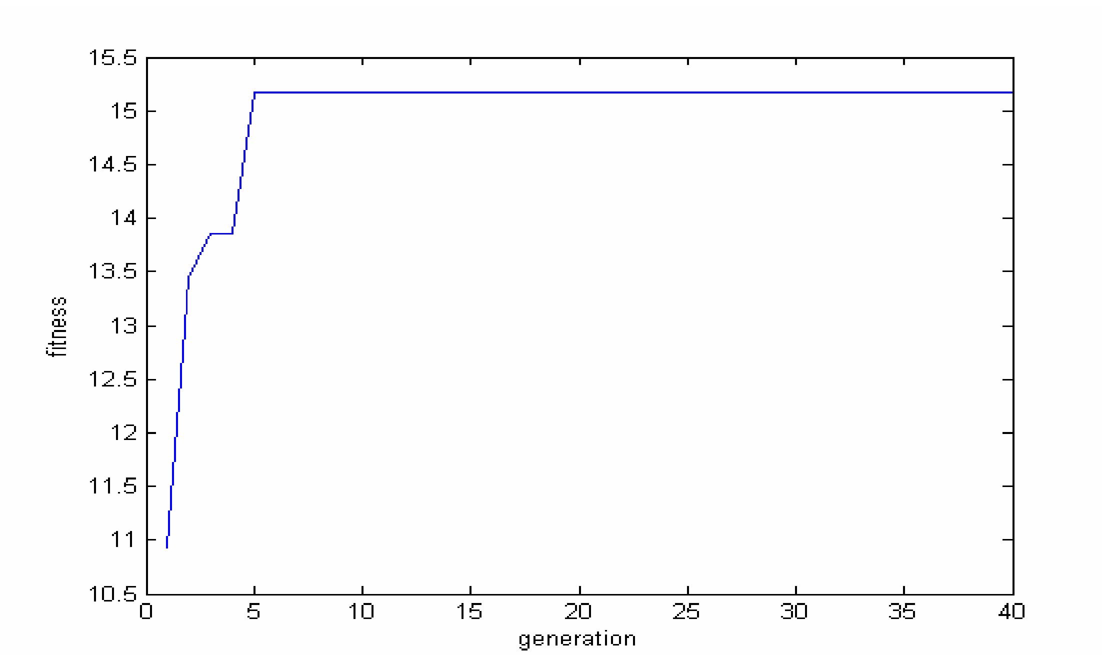


Figure III-13 : Fitness of the best solutions.

The minimum of the function $f(\mathrm{x})$

$f_{\min} = -15.1644.$

The results of the optimization are represented by the following figure

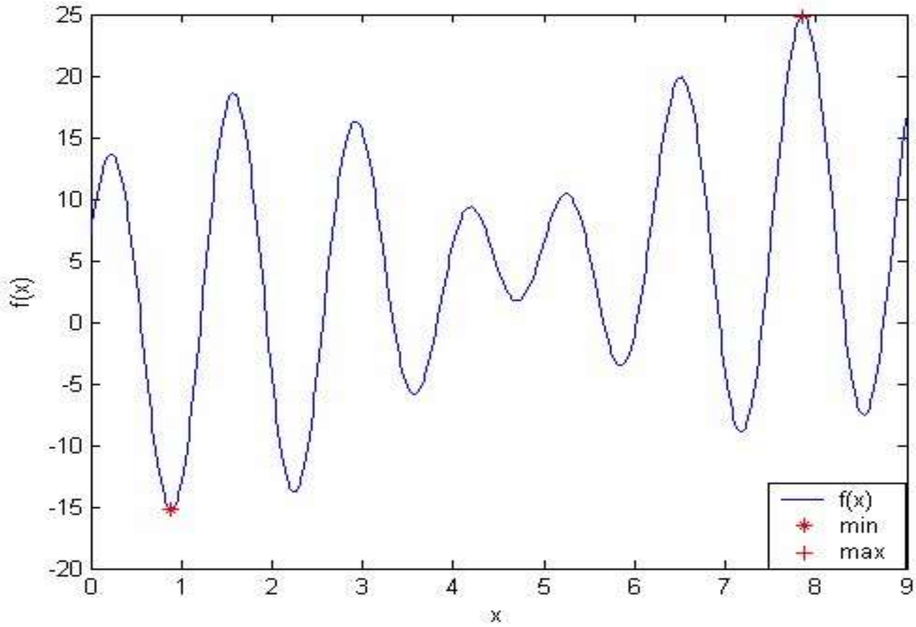


Figure III-14 : The optimums of the function f (x).

III.7.2.Optimization of the second function:

The appearance of this function is given in Figure III.15.

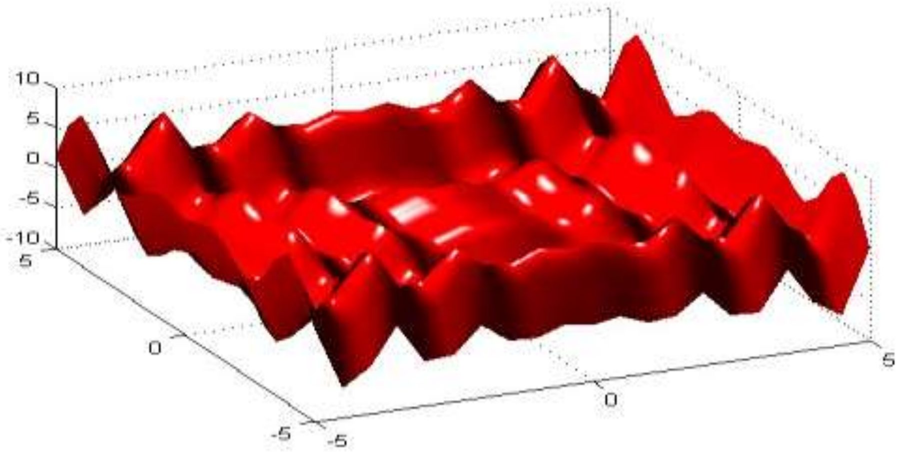


Figure III-15 : The appearance of the function f (x, y)

After 200 generation the minimum calculated by the genetic algorithm fmin = −18.5547 is given for the values x = 9.039 and y= 8.6682

III.8.Applications of genetic algorithms:

- Due to their high performance, simplicity and robustness, the applications of genetic algorithms are numerous and diverse. In addition to function optimization, they are applied to other areas of science such as :
- Biology: simulation of the biological cell (Rosenberg 1967 and Weinberg 1970)
- Pattern recognition (Cavicchio 1970).

III.9.Optimization of the parameters of PID regulator using Algorithem genetic / Matlab:

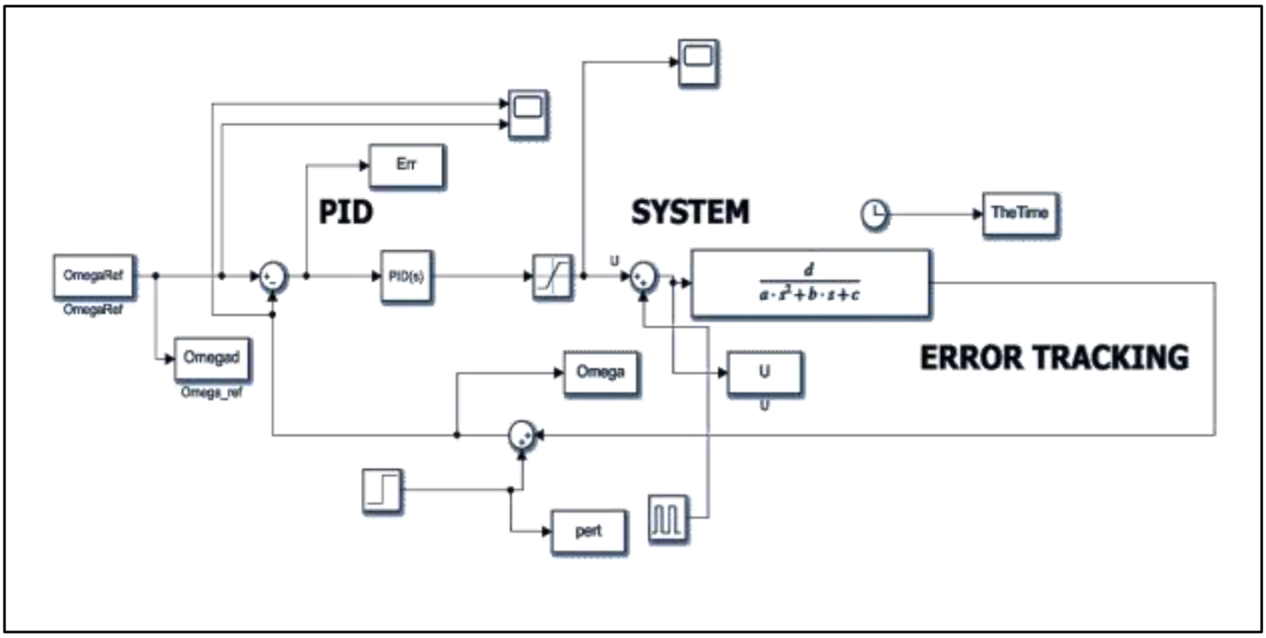


Figure III-16 : block diagram of the genetic algorithm (GA).

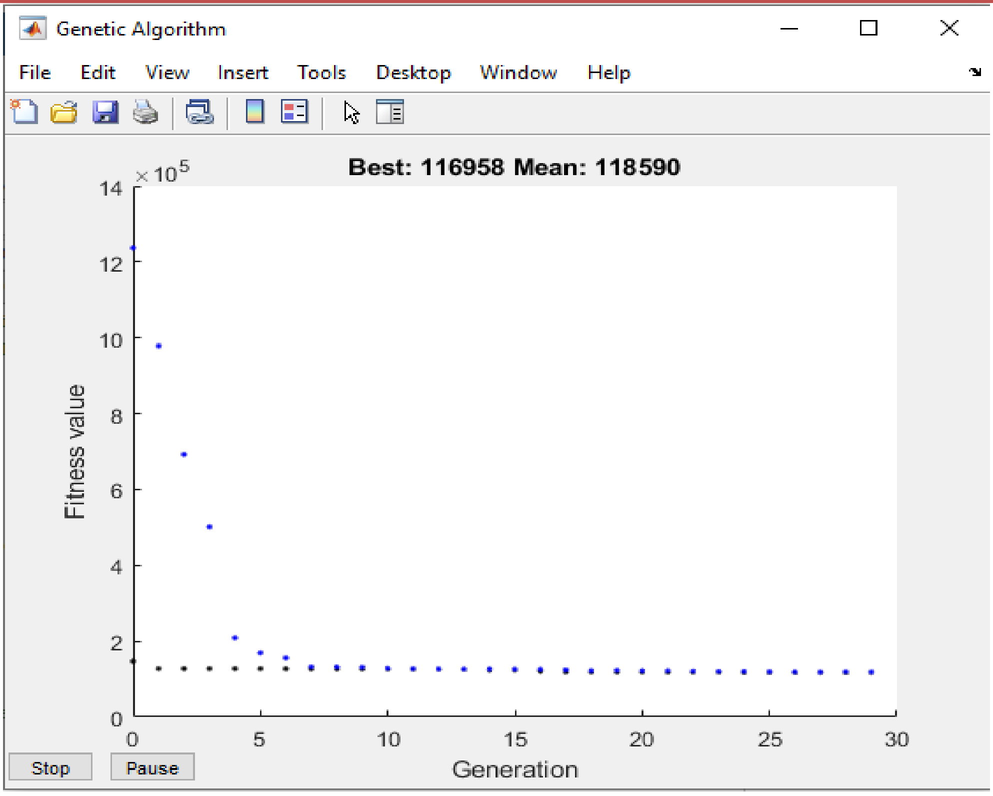


Figure III-17: Convergence of the Genetic Algorithm for the PID regulator.

The figure below shows the convergence of algorithm genetic used to optemize the pid controller:

We can see that the Fitnesse value converge to the best after 8 generation.

Table III-1: the parameters of PID regulator founed

Parameters identified	Kp	Ki	Kd
values	99.2210	2.0301	0.3429

Then we use them in our simulation :

```
sim('MCC_simulation.slx');
plot(TheTime,Err,'r','LineWidth',3)
title ('tracking Error');
grid;
figure
```

```
plot(TheTime,Omegad,'*',TheTime,Omega,'r','LineWidth',2)
xlabel('Time (second)')
ylabel('omega (turn/min)')
title ('rotational speed (rad/s)')
legend('omega ref','omega'),grid;
% axis([0 5,0 500]);
figure
plot(TheTime,U,'r','LineWidth',3)
title ('Control (V)')
grid;
```

III.10.simulation and result :

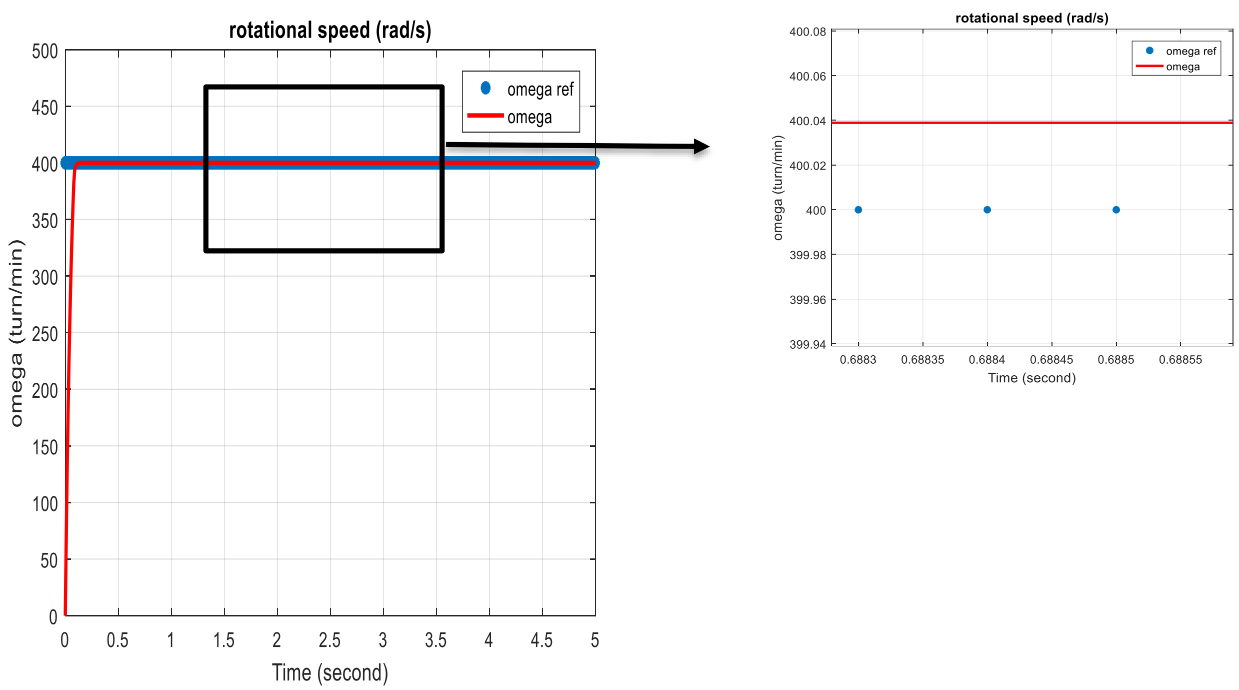


Figure III-18: rotational speed of the DC motor

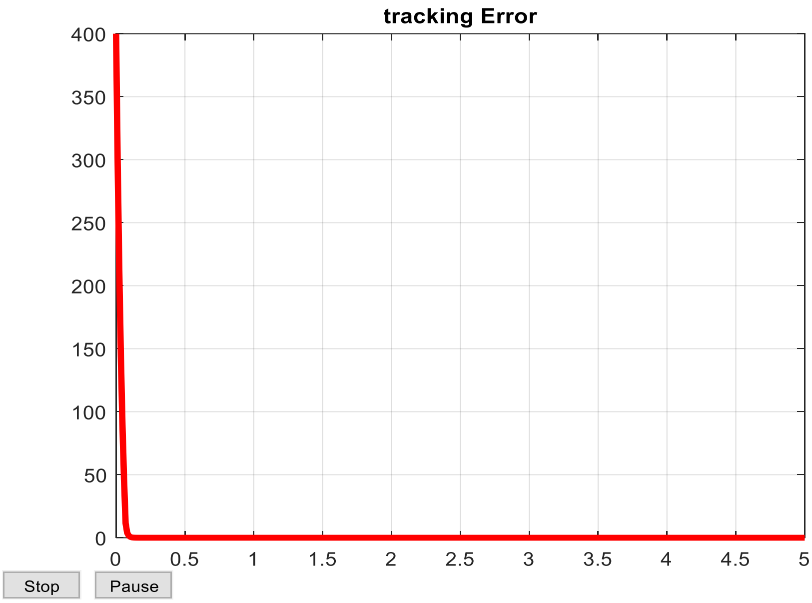


Figure III-19 Tracking error of the rotational speed

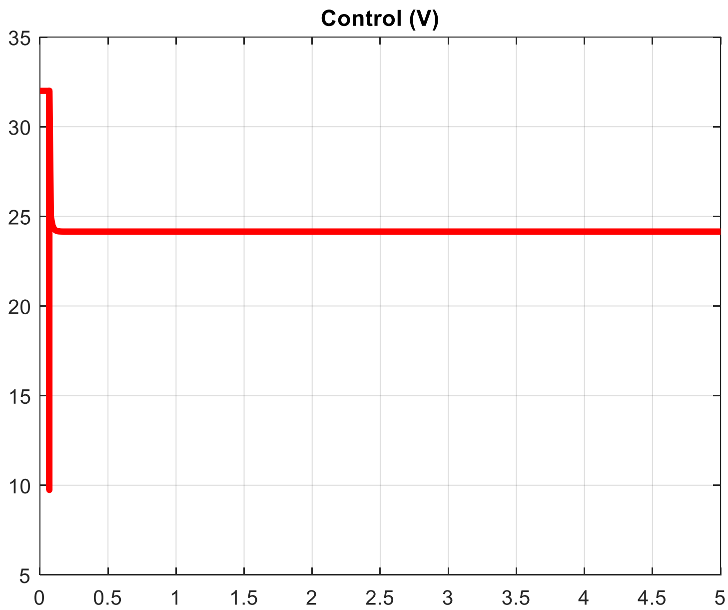


Figure III-20 the signal of control

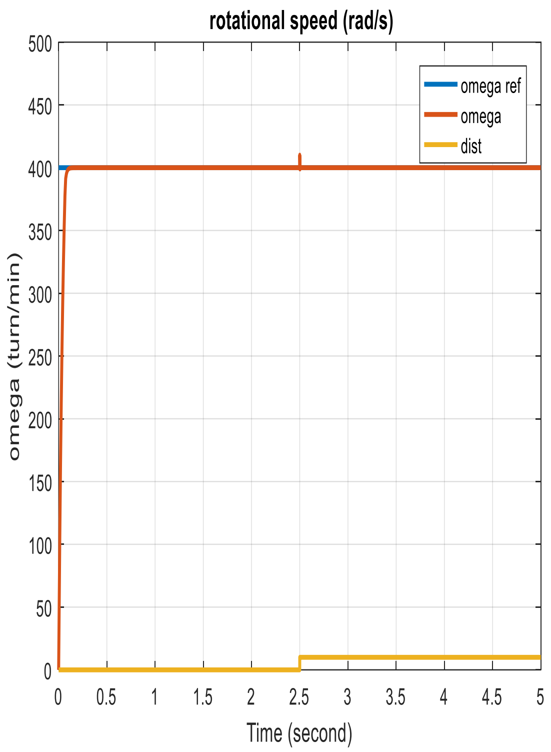
III.11.Interpretation of results

In those past figures we can see that the DC Motor reached the maximum speed (400 round per min) in a short time and with a little amount of error .

III.12.Robustness Test

Versus external disturbance:

The following graphs present the system response with external disturbance:



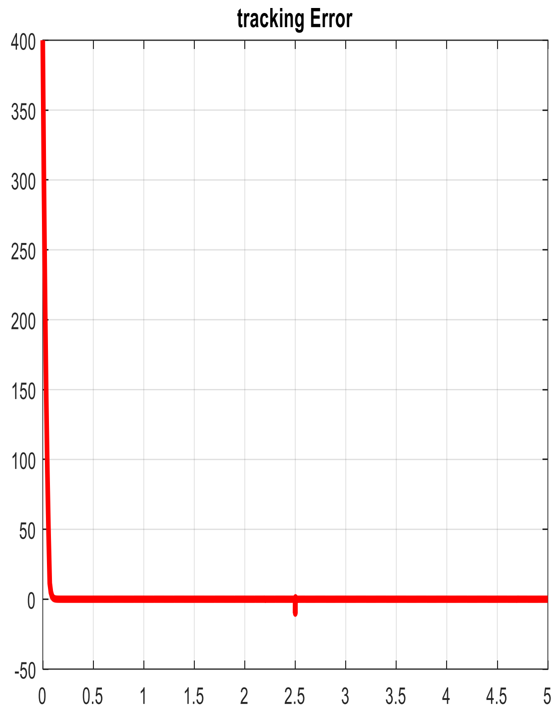


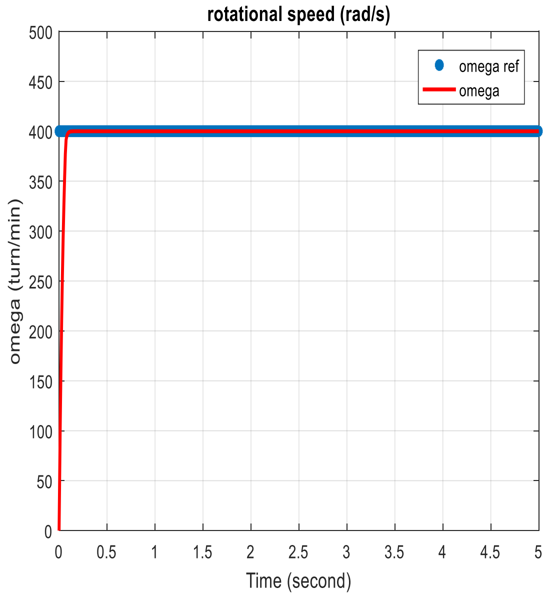
Figure III-21: rotational speed of the DC motor **Figure III-22:** the control

With external disturbance

From the figures above, it can be said that the control is robust versus the external disturbance

Versus disruption of the order:

We make a disturbance of the control then we simulate the system with the disturbed control:



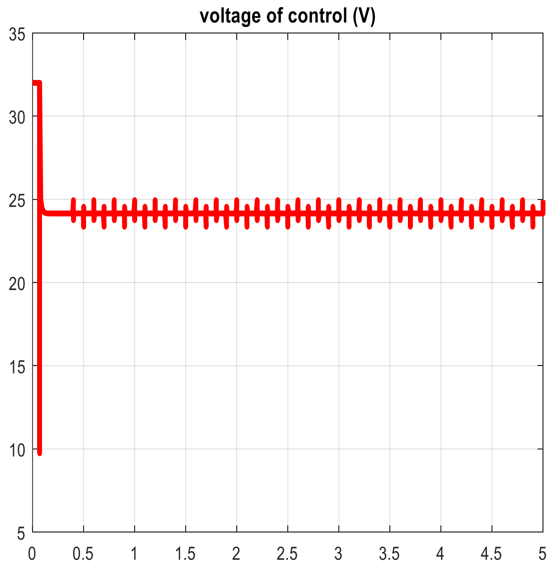


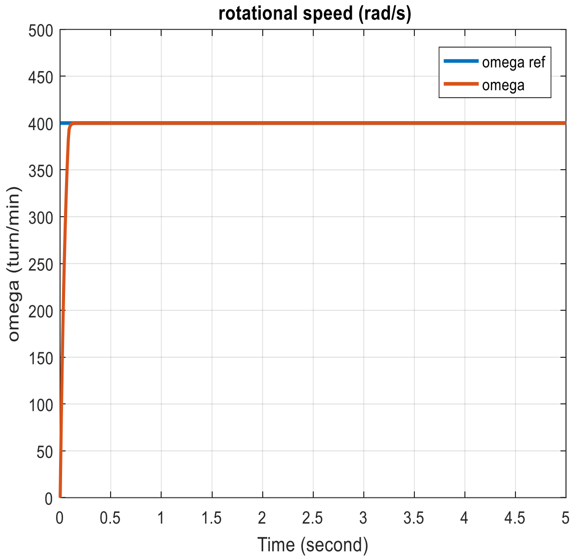
Figure III-23: the rotational speed of DC motor with Control disturbed

Figure III-24: the Control disturbed

From the figures above we can say that the control is robust versus the disturbance of the control.

Versus parametric change :

- We make a change of the parameters of the motor (the resistance and the coil) of 20% then we will simulate the system



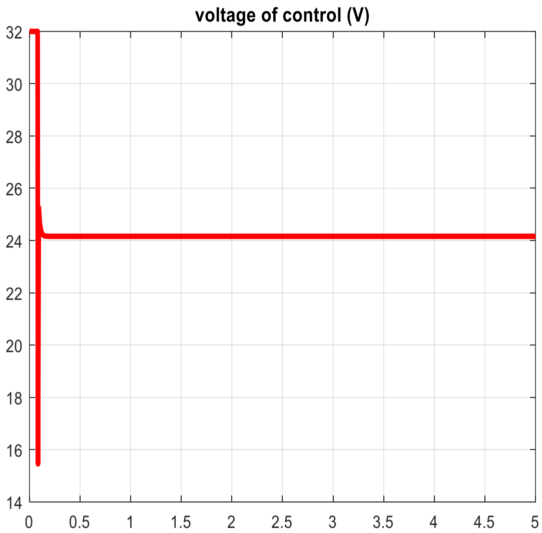


Figure III-25: the rotational speed of DC motor with parametric change

Figure III-26: the Control

From the figures above, we can say that the control is robust versus the parametric disturbance.

III.13.Conclusion:

Genetic algorithms are exploration algorithms developed for optimization purposes. Based on the mechanisms of natural selection and genetics, they evolve with each generation, a population that reproduces, following crossing operations and mutations.

In this chapter, we have introduced the basic principles of genetic algorithms. Their applications are numerous and diverse. In the field of electrical engineering, genetic algorithms are used for the parametric identification of machines and for the optimization of regulators used within vector control.

General conclusion

General conclusion

The work we have done in this paper aims to propose a technique for controlling the speed of a DC motor. This technique Genetic Algorithm (GA) includes setting the optimal parameters PID (Kp, Ki, Kd) and (Kp, Ki, Kd) to improve the performance of the system to be tuned in closed loop.

whereas the first chapter Describes different structures of PID controllers, different performance criteria, and some classic methods of setting up control loops, such as the Ziegler-Nichols method, and the next chapter provides an overview of the mathematical modeling and design of DC motors, followed by a brief overview of the methods used to control the speed of DC motors, in the last chapter the new optimization technology (GA) and its working principle are introduced, and then the speed of DC motor is controlled by PID controller in SIMULINK/MATLAB environment.

To achieve this goal, we use a performance criterion, the integral of the absolute value of the velocity tracking error. The criterion is defined as the objective function to be minimized in order to find the optimal solution of the PID controller results during optimization.

BIBLIOGRAPHIC REFERENCES

[1] K. Aström-Hägglund, « PID controllers: theory design and tuning », Second edition-instrument society of America, 1995.

[2] P. Rousseau, « Asservissement linéaires continus », Ellipses, Paris, 2004.

[3] S.ARFA & E.M. BARHOUMI, « Introduction aux systèmes asservis », ISET Kairouan-Tunisie.

[4] D. Ross-Etienne & D.Mickaël Camus, « Asservissement par PID », Mars 2010.

[5] H. BOUICHE & M. BRAHAM, « Commande PID d'un moteur à courant continu », Mémoire de Master en Automatique, Université de Bejaia, Juin 2010.

[6] P. Prouvost, « Automatique-contrôle et régulation », France, 2004.

[7]. S. BENZENATI & M. RABIAI Commande floue d'un moteur à courant continu à excitation séparée, Mémoire de Master en Automatique, Université de Bejaia, Juin 2013.

[8]. I. CHALANE & T. OUARI, « Optimisation des paramètres d'un PID par essaims particulaires (PSO) », Mémoire de Master en Automatique, Université A. MIRA de Bejaia, Juin 2017.

[9]. Ziegler J .G., Nichols N .B., Optimum setting for automatic controllers, ASME, vol. 64,

pp. 759-768, novembre 1942.

[10] Goldberg, D. E., Genetic Algorithms in Search, Optimization & Machine Learning, Addison-Wesley, 1989.

[11] Bentley, P. J. and Wakefield, J. P., An Analysis of Multiobjective Optimisation in Genetic Algorithms, Submitted to Evolutionary Computation, 1996.

[12] Holland, J. H., Genetic Algorithms, Scientific American, pp. 66-72, 1992.

[13] Zoghmar M, Habchi A, Étude comparative entre deux régulateurs PID et FLC

appliqués à la Machine à Courant Continu, master Génie électrique, Université LARBI

BEN M'HIDI.

[14] A. Zaatri, Asservissements et Régulation Industrielle, Cours destiné aux étudiants de

4éme Année Ingénieurs, Université Mentouri, Constantine.

Annex

Program of Genetic Algorithm

```
% exemple GA
% here, we want to find the the parameters of a PID controller (Kp, Ki, Kd)
% the system is second order 'd/(ax^2+bx+c)'
% the aim is to track a desired signal
clear all;
close all;
clc;
% system parameters
% system parameters
R  = 1.91;
Kc = 60.3*1e-3;
Ke = 60.3*1e-3;
L = 0.63*1e-3;
J = 1e-4;
f = 2.5*(1e-6);
%**************************
Umax = 32;
Umin = 0;
a = L*J;
b = R*J + L*f;
c = R*f + Kc*Ke;
d = Kc;
% a = 1;
% b = 2;
% c = 3;
% d = 4;
% sim parameters
time_sim = 5;  % Sim Time
Ts = 1e-4;     % Step Time
Kp_min = 0.1;
Kp_max = 100;
Ki_min = 0.1;
```

```
Ki_max = 100;
Kd_min = 0.1;
```

Program of simulation

```
Kd_max = 20;

OmegaRef = 400; % 400 (rad/s)
% perturbation Omega
PertO_Amp = 0;
PertO_temps = 0;
% perturbation Commande
PertU_Amp = 0;
PertU_periode = 50;
PertU_Temps = 0;
global Kp Ki Kd
Kp = 1;
Ki = 1;
Kd = 1;
options=gaoptimset('Generations',30,'PopulationSize',20,'StallGenLimit',inf,...'StallTimeLimit',inf,'TolFun',1e-006,'TolCon',1e-006,'plotfcns',@gaplotbestf);

[Controller,TheError]=ga(@FonctionCout,3,[],[],[],[],[Kp_min,Ki_min Kd_min],[Kp_max Ki_max Kd_max],[],options);
Kp = Controller(1)
Ki = Controller(2)
Kd = Controller(3)
sim('MCC_simulation.slx');
plot(TheTime,Err,'r','LineWidth',3)
title ('tracking Error');
grid;
figure
plot(TheTime,Omegad,'*',TheTime,Omega,'r','LineWidth',2)

xlabel('Time (second)')
ylabel('omega (turn/min)')
title ('rotational speed (rad/s)')
legend('omega ref','omega'),grid;
% axis([0 5,0 500]);
```

```
figure
plot(TheTime,U,'r','LineWidth',3)
title ('Control (V)')
grid;
```