



Republic of Iraq

Ministry of Higher Education & Scientific Research

University of Kerbala

College of Engineering

Department of Electrical and Electronic Engineering

**Intelligent Control for Underactuated Robot Based on
Optimization Technic**

A Thesis Submitted to the Council of the College of Engineering/University
of Kerbala in a Partial Fulfillment of the Requirements for the Degree of
Master of Science (M.Sc.) in Electrical Engineering

Written By:

Aya Flyah Hassan

Supervised By:

Prof.Dr. Haider Galil Kamil

Asst.Prof. Dr. Ahmed Abdulhadi Ahmed

March 2024

Ramadan 1445

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

﴿ وَلَسَوْفَ يُعْطِيكَ رَبُّكَ فَتَرْضَىٰ ﴾

صدق الله العلي العظيم

(الضحى: الآية 5)

Examination committee certification

We certify that we have read the thesis entitled “**Intelligent Control for Underactuated Robot Based on Optimization Techniques**” and as an examining committee, we examined the student “**Aya Flyah Hassan**” in its content and in what is connected with it and that, in our opinion, it is adequate as a thesis for the degree of Master of Science (M.Sc.) in Electrical Engineering.

Supervisor

Signature:

Name: Prof.Dr. Haider Galil

Kamil

Date: 28/3/2024

Supervisor

Signature:

Name: Asst.Prof. Dr. Ahmed

Abdulhadi Ahmed

Date: 28/3/2024

Member

Signature:

Name: Asist. Prof. Dr. Dhirgaam

A Kadhim

Date: 26/3/2024

Member

Signature:

Name: Asist. Prof. Dr. Jabbar

Salman Hussein

Date: 26/3/2024

Chairman

Signature:

Name: Asist. Prof. Dr. Abidaoun H. Shallal

Date: 28/3/2024

Signature:

Name: Prof. Dr. Muayad Kod

Head of the Department of Electrical
and Electronic Engineering

Date: 17/4/2024

Signature:

Name: Assist. Prof. Dr. Haider
Nadhom Azziz

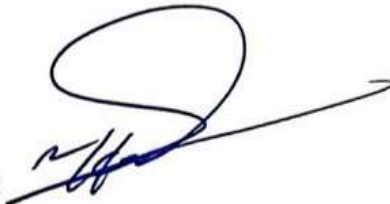
Dean of Engineering College

Date: 17/4/2024

Supervisor certificate

I certify that the thesis entitled "**Intelligent Control for Underactuated Robot Based on Optimization Techniques**" was prepared by **Aya Flyah Hassan** under our supervision at the Department of Electrical Engineering, College of Engineering, University of Kerbala as a partial fulfilment of the requirements for the Degree of Master of Science (M.Sc.) in Electrical Engineering.

Signature:



Prof. Dr. Haider Galil Kamil

Date: 28/3 / 2024

Signature:




Asst. Prof. Dr. Ahmed Abdulhadi Ahmed

Date: 28/3 / 2024

Linguistic certificate

I certify that the thesis entitled " **Intelligent Control for Underactuated Robot Based on Optimization Techniques**" which has been submitted by **Aya Flyah Hassan** has been proofread and its language is amended to meet the English style.

Signature: 

Lect. Dr. Hussein Ali Hadi

Date 28/3 / 2024

Undertaking

I certify that the research work “**Intelligent Control for Underactuated Robot Based on Optimization Techniques**” is mine. The work has not been presented elsewhere for assessment, and the materials used from other sources have been properly acknowledged/referred.

Signature: 

Aya Flyah Hassan

Date: 28/3 / 2024

Dedication

To

My father, thanks to whom I completed my studies and his constant faith in me.

My mother who supported me from my childhood until I was able to achieve my dream.

My brothers and sisters, thank you for your faith in me and your support.

My supervisor, Prof. Dr. Haider Galil Kamil, for his belief in me, his help, and his support.

Acknowledgements

I would like to thank my family for their support, love, and everything. We would like to express our sincere appreciation to my supervisors, “Prof. Dr. Haider Galil Kamil and Asst. Prof. Dr. Ahmed Abdulhadi Ahmed” for their directions and accurate notes in completing this work.

Also, I would like to extend my thanks to all lecturers and the staff of the Electrical and Electronic Engineering Department of the College of Engineering at the University of Kerbala for providing the facilities to complete this work.

More thanks and appreciation to my country, Iraq, and to every good Iraqi person.

Abstract

This study aims to understand the complexity and control balancing in the upright position of a three-link underactuated robot system. The Robogymnast is one of the important types of three-link systems mimicking human acrobatics; it composes three joints and three links (arm, torso, and leg, respectively) powered by two geared DC motors.

A mathematical model for the robot derived using Euler-Lagrange equations. Since the system is a nonlinear multi-link mechanism requiring a complex mathematical model considering information accuracy. It presents more challenges modeling the Robogymnast and dealing with control motion problems. The Euler-Lagrange formula and Artificial Neural Network (ANN) model are used to model the nonlinear Robogymnast system. The comparison results show that the dynamic model obtained by the ANN is significantly better than the model derived from the Euler-Lagrange formula because the ANN model accepted the initial deviation of absolute angle for each link up to 3 degrees. In contrast, the dynamic model derived from the Euler-Lagrange formula accepted 1 degree and becomes unstable at 3 degrees.

Firstly, a Discrete Linear Quadratic Regulator (DLQR) controller is used to balance the gymnastic robot in the upright position. The construction of DLQR depends on the selection of the weight matrices.

Secondly, to find optimum values of the weighting matrices; a swarm optimization technique called Whale Optimization Algorithm (WOA) is applied to adjust the weighting matrices. As well as, another optimization technique is used to find the optimum values of weighing matrices, this technique is called Aquila Optimization (AO). The evaluation of the best

technique has been implemented. The WOA-based DLQR controller achieves the best result according to the transient response of the relative angles but consumes higher voltage from the two motors compared to the AO-based DLQR controller. The first, second, and third links reached a steady state after 1.825 seconds with a minimum deviation (-0.5° and -1.5° for the first and second links, respectively, and no deviation for the third link). Moreover, the control voltage of the first motor consumed 7.12V, and the second motor consumed 2V to achieve the desired response, which is less than the limited voltage (12V).

Thirdly, a Fuzzy Logic Controller (FLC) was designed to achieve online tuning to stabilize and balance the system. The result of the FLC showed that the system consumed more settling time than WOA-based DLQR to be stable in an inverted position. Therefore, a hybrid controller combining FLC with WOA-based DLQR was proposed to achieve online tuning with less settling time for the relative angular position (1.5 seconds) and acceptable deviation of the links from the upright balancing point (-1.15° and -3.4° for the first and second links respectively, and no deviation for the third link). The first motor consumed 6.7 volts of control effort, but the second motor consumed only -1.5 volts; this was considered satisfactory voltage to bring the Robogymnast to an inverted position and stabilize it in the upright balancing point within a suitable duration.

Finally, the comparison among the previous methods demonstrated that the hybrid system achieves online tuning with a satisfactory response to stabilize the gymnastic robot vertically. The comparison with previous research demonstrates that the FLC with the WOA-based DLQR method achieves the best transient response regarding overshoot, settling time, and less control effort than the other methods.

Table of Contents

Abstract.....	iv
Table of Contents	vi
List of Tables.....	ix
List of Figures	x
List of Abbreviations.....	xii
List of Symbols	xiii
Chapter One: Introduction.....	1
1.1 Background.....	1
1.2 Problem Statement.....	2
1.3 Aims and Objectives of the Thesis.....	3
1.4 Research Methodology	3
1.5 Thesis Organization.....	4
Chapter Two: Backgrounds and Literature Review.....	6
2.1 Introduction	6
2.2 Balancing Control Problem	6
2.3 Discrete Linear Quadratic Regulator.....	19
2.4 Whale Optimization Algorithm (WOA).....	20
2.4.1 Pseudo-code of the WOA algorithm.....	21
2.5 Aquila Optimization (AO) Algorithm.....	25
2.5.1 Pseudo-code of the AO algorithm.....	27
2.6 Fuzzy Logic Control (FLC).....	32
2.7 Summary.....	35
Chapter Three: The System Description and Mathematical Model	36

3.1	Introduction	36
3.2	System Description.....	36
3.3	Robogymnast Mathematical Model in the Upward Position Derived by Euler-Lagrange.....	37
3.4	Mathematical Model in the Upward Position Derived by Elman Neural Network (ENN)	48
3.5	Summary.....	50
Chapter Four: Controller Design with Different Control Strategy ...		52
4.1	Introduction	52
4.2	Results of DLQR Applying a Mathematical Model Derived from the Euler-Lagrange	52
4.3	Results of DLQR Applying a Mathematical Model Derived from the Elman Neural Network (ENN).....	55
4.4	Results of Whale Optimization Algorithm.....	58
4.5	Results of Aquila Optimizer.....	60
4.6	Results of Fuzzy Logic Control.....	62
4.7	Hybrid System Fuzzy Logic and Whale Optimization	65
4.7.1	Results of Hybrid Control	67
4.8	Comparative results	69
4.8.1	Comparison Between Before and After Optimization Techniques	69
4.8.1	Comparison Between WOA-based DLQR, FLC, and Hybrid Controller.....	71
4.9	Comparison with other work.....	74
4.10	Summary.....	79

Chapter Five: Conclusion, Contributions, and Future Work	80
5.1 Conclusion	80
5.2 Contributions	82
5.3 Future Work.....	82
References	83
Appendices	91
Appendix A.....	91

List of Tables

Table 2-1 General overview based on Type of System and Controller	13
Table 2-2: Parameters of Optimization	22
Table 2-3 Fuzzy Logic rules base.....	34
Table 3-1: Values of parameters for Robogymnast.....	45
Table 3-2: Values of parameters for Motor.....	45
Table 4-1: Compared performance analysis for angles response DLQR+WOA with DLQR	76
Table 4-2 : Compared performance analysis for relative angles response DLQR+WOA with DLQR+IWO	76
Table 4-3: Compared performance analysis for relative angles response FLC+WOA with FLIWOH.	77
Table 4-4: Compared performance analysis for angles response DLQR+WOA with DLQR	79

List of Figures

Fig. 2.1. Flow chart of the whale optimization.....	26
Fig. 2.2 Flow chart of Aquila optimization algorithm	33
Fig. 2.3 Membership function with two inputs and one output.	34
Fig. 3.1 Hardware components of Robogymnast	38
Fig. 3.2 Robogymnast schematic in the upright posture	38
Fig. 3.3 Elman Neural Network model of the Robogymnast	49
Fig. 4.1 Time response with initial $q_1=1^\circ$, $q_2=-1^\circ$, $q_3=1^\circ$ of the Euler-Lagrange formula. (a) Relative angle 1, 2, 3 (b) Control effort 1, 2.	54
Fig. 4.2 Time response with initial $q_1=3^\circ$, $q_2=3^\circ$, $q_3=3^\circ$ of the Euler-Lagrange formula. (a) Relative angle 1, 2, 3 (b) Control effort 1, 2.	54
Fig. 4.3 Time response with initial $q_1=3^\circ$, $q_2=3^\circ$, $q_3=3^\circ$ of the ENN model. (a) Relative angle 1, 2, 3 (b) Control effort 1, 2.	56
Fig. 4.4 Block diagram for optimization-based DLQR controller.	57
Fig. 4.5. Time response after WOA with initial $q_1=3^\circ$, $q_2=3^\circ$, $q_3=3^\circ$	59
Fig. 4.6 Convergence curve of WOA.....	60
Fig. 4.7 Time response after Aquila Optimization with initial $q_1=3^\circ$, $q_2=3^\circ$, $q_3=3^\circ$. (a) Relative angle 1, 2, 3 (b) Control effort 1, 2.	61
Fig. 4.8 Simulation System of Fuzzy Logic	62
Fig. 4.9 FLC time response with initial deflection $q_1=3^\circ$, $q_2=3^\circ$, $q_3=3^\circ$. (a) Relative angle 1 (b) Relative angle 2 (c) Relative angle 3 (d) Control effort 1 (e) Control effort 2.....	65
Fig. 4.10. Simulation System of Hybrid Controller.	66
Fig. 4.11 Hybrid control time response with initial deflection $q_1=3^\circ$, $q_2=3^\circ$, $q_3=3^\circ$. (a) Relative angle 1 (b) Relative angle 2 (c) Relative angle 3 (d) Control effort 1 (e) Control effort 2.....	68

Fig. 4.12 Comparison time response between DLQR, WOA-based DLQR, and AO-based DLQR. (a) Relative angle 1 (b) Relative angle 2 (c) Relative angle 3 (d) Control effort 1 (e) Control effort 2.	71
Fig. 4.13. Comparison time response between WOA-based DLQR, FLC and Hybrid controllers. (a) Relative angle 1 (b) Relative angle 2 (c) Relative angle 3 (d) Control effort 1 (e) Control effort 2.....	74
Fig. 4.14. Time response after WOA-based DLQR controller with initial $q_1=1.3^\circ$, $q_2=0.2^\circ$, $q_3=-6.5^\circ$. (a) Relative angle 1, 2, 3 (b) Control effort 1, 2..	75
Fig. 4.15. Time response after WOA-based DLQR controller with initial $q_1=1^\circ$, $q_2=-0.9^\circ$, $q_3=1^\circ$. (a) Relative angle 1, 2, 3 (b) Control effort 1, 2.....	78
Fig. A.1.1 Range of Q1. (a) Relative angle 1 (b) Control effort 1.	91
Fig. A.2.1 Range of Q2. (a) Relative angle 1 (b) Control effort 1(c) Control effort2.....	92
Fig. A.3.1 Range of Q3. (a) Relative angle 3 (b) Control effort 2.	93
Fig. A.4.1 Range of Q4. (a) Relative Angular Velocity 1(b) Control effort 1.	94
Fig. A.5.1 Range of Q5. (a) Relative Angular Velocity 2 (b) Control effort 1(c) Control effort 2.....	95
Fig. A.6.1 Range of Q6. (a) Relative Angular Velocity 3 (b) Control effort 2.	96
Fig. A.7.1 Range of R1. (a) Control effort 1 (b) Control effort 2 (c) Relative angle 1 (d) Relative angle 2.....	97
Fig. A.8.1 Range of R2. (a) Control effort 2 (b) Control effort 1 (c) Relative angle 2 (d) Relative angle 3.....	99

List of Abbreviations

AO	Aquila Optimization
BA	Bees Algorithm
DOF	Degrees of Freedom
DLQR	Discrete Linear Quadratic Regulator
DLIP	Double Link Inverted Pendulum
ENN	Elman Neural Network
FLC	Fuzzy Logic Control
FSC	Fuzzy Supervisor Control
GWO	Gray Wolf Optimization
IWO	Invasive Weed Optimization
IP	Inverted Pendulum
LQR	Linear Quadratic Regulator
PSO	Particle Swarm Optimization
PID	Proportional Integral Derivative
TIP	Triple Inverted Pendulum
WOA	Whale Optimization Algorithm

List of Symbols

θ_i	Angle of i_{th} link from the vertical line
a_i	Centre of gravity of i_{th} link
D	Dissipation Energy
K	Gain matrix
T_i	Generalized torque at an angle θ_i
g	Gravitational acceleration (9.81 m/s ²)
K	Kinetic Energy
L_i	Length of the i_{th} link
m_i	Mass of the i_{th} link
I_i	Moment of inertia of i_{th} link around its center of gravity
J	Performance index
P	Potential Energy
K_i	Ratio of the i_{th} gearbox
q_i	Relative angle of i_{th} link
T_s	Sampling period
G_i	Static gain of i the motor/gearbox
C_{pi}	The gearbox's output shaft reflected the viscous friction coefficient of the motor and gearbox in the form of energy loss
I_{pi}	The moment of inertia of i_{th} motor/ gearbox reflected at the output shaft of the gearbox
T_{mi}	Torque generated by the i_{th} motor
C_i	Viscous friction coefficient of the i_{th} joint
Q, R	Weight matrices

Chapter One: **Introduction**

1.1 **Background**

The technological advances of the information technology age have fueled a growing interest in humanoid robotics. Consequently, various humanoid robots, such as ASIMO [1], Hubo [2], and Atlas [3], have been developed. These robots showcase diverse capabilities, including walking on uneven terrain, climbing ladders, synchronized dancing, assisting elders with household tasks, participating in sports, manufacturing and assembly, search-and-rescue missions, and tasks in hazardous environments, among other applications.

A humanoid robotic system is a robot with a body form constructed to mimic the human body. Humanoid robots simulate human movements and consist of rigid links coupled to each other; a joint is the connection point between two links. Robots possess one degree of freedom when equipped with a single joint. A robot is considered to have 'n' degrees of freedom if it incorporates 'n' joints. Therefore, the complexity of a robot is directly tied to the number of degrees of freedom it possesses. Using actuators simulates human muscles and cartilage to achieve full-body motions like running, jumping, crawling, etc. Since these robots are anticipated to work alongside humans and perform challenging tasks, it is crucial that the control algorithms and planning are efficient, robust, and capable of real-time execution [2].

This thesis develops a three-degree-of-freedom under-actuated gymnastic robot based on a triple inverted pendulum (three links and three joints).

1.2 Problem Statement

Technological developments have spurred an increased interest in humanoid robotics in recent years. A Robogymnast, a specific humanoid robot, is designed to mimic and execute gymnastic maneuvers. Previous literature presented simulations of robot gymnastics in swing and balance motions [4], [5], [6], [7]. However, it did not emphasize the balancing stage and the proportionality of results for practical application. The balancing results exhibited a high overshoot, requiring more settling time to reach the vertical position, impacting the stability of the gymnastics robot, especially when exposed to external disturbances, which may cause it to swing again and return to the downward position.

The main objective is to construct a control system capable of keeping the pendulum standing with time response specifications more suitable for practical application and to create a control system capable of handling the complex, dynamic movements involved in gymnastics. Robogymnast applications are in various fields, such as: validating different control algorithms and developing entertainment and sports training. They offer a valuable tool for studying biomechanics and control systems, performing tasks that could be dangerous or impossible for humans. Tasks requiring real-time reaction necessitate control software that operates in real-time with a short settling time to accomplish these actions reliably. Thus, practical operation is the most essential specification when developing humanoid robot control software.

A Robogymnast is a multi-link underactuated system that is a benchmark system that illustrates various control techniques and is commonly used in laboratories to perform and verify emerging technologies in control

engineering. Studying these systems can help researchers explore solutions for addressing motion problems suffered by disabled and injured individuals experiencing limb issues.

1.3 Aims and Objectives of the Thesis

The research aims to model, simulate, and develop several methods of controlling the balancing of a triple-link gymnastic robot.

The above aims will be achieved by performing the research goals as follows:

1. Evaluate the mathematical models used to describe the gymnastic robot.
2. Design and simulate controllers to balance the Robogymnast vertically.
3. Design an adaptable controller to maintain the Robogymnast balanced upright.
4. Applying swarm-based (WOA and AO) optimization approaches, determine the controllers' optimum parameters.
5. Evaluate and validate the suggested controllers using a different swarm-based optimization method.
6. Develop hybrid swarm-based optimization controllers to design an intelligent, robust controller.

1.4 Research Methodology

The following approach was used to achieve the goals mentioned above:

- Review and understand the most relevant research in the control field for complex multi-link mechanisms, considering various control methods to identify and address weaknesses or improve the results obtained using the most suitable approach to solve the problem.

- The Euler-Lagrange method and Artificial Neural Network Modeling are used for derived the dynamic equations and obtain the mathematical model of the Robogymnast.
- The issue focuses on balancing the Robogymnast in an inverted position. The Whale Optimization Algorithm and Aquila Optimization are used to develop the DLQR controller by selecting its optimal parameters.
- The optimal DLQR parameters are obtained and applied to the Artificial Neural Network Modelling of the Robogymnast, and the result is compared with the Euler-Lagrange model results.
- Apply the fuzzy logic controller to the optimal model achieved online tuning.
- A combined FLC controller with optimal DLQR parameters to the optimal model to obtain a response close to the demand in a real application.

1.5 Thesis Organization

The remaining chapters of the thesis are structured as follows:

Chapter 2 reviews the problems related to complex multi-link mechanisms and their applications, such as balancing control, which are discussed with different control systems.

Chapter 3 covered the Robogymnast system description and derivation of the mathematical model equation of the mathematical model utilizing the Euler-Lagrange technique and presented the Artificial Neural Network Model.

Chapter 4 tests the system's performance with several control strategies intended to stabilize the system. The controller types employed include

DLQR, WOA-based DLQR, AO-based DLQR, FLC, and hybrid FLC controllers with WOA-based DLQR controllers.

Furthermore, a comparison between the suggested control systems was provided, as well as a comparison with previous research.

In Chapter 5, the thesis results are summarized, the study's contributions are explained, and recommendations for further research are made.

Chapter Two: Backgrounds and Literature Review

2.1 Introduction

This chapter provides an overview of the relevant literature related to the Inverted Pendulum system and its association with under-actuated control systems. The reviewed literature encompasses a variety of control systems and incorporates diverse control techniques, with a focus on single, double, and triple inverted pendulum systems. These systems have found numerous applications aimed at addressing human challenges. Researchers have explored and implemented various control techniques specifically to address the upright balancing issues of inverted pendulum systems.

2.2 Balancing Control Problem

All The control problem of balancing an inverted pendulum represents a classic problem in the discipline of control. The main objective is to construct a control system capable of keeping the pendulum standing. To address this problem, different control strategies have been utilized by the researchers, as follows:

(A. Z. Alassar, 2010) [8] the focus was on modeling and controlling a robot arm with five degrees of freedom. The study compared the outcomes of Fuzzy Logic Controllers (FLC) and Fuzzy Supervisor Controllers (FSC) with Proportional Integral Derivative (PID) responses. The FSC adjusts PID gains since PID does not perform effectively in nonlinear systems. FLC outperforms classical PID controllers regarding time response, and FSC outperforms classical approaches like Ziegler-Nichols when adjusting PID parameters.

(S. Sehgal and S. Tiwari, 2012) [9] utilized the Lagrange equation to elucidate the Triple Inverted Pendulum (TIP) model of the automobile. Subsequently, the triple inverted pendulum is linearized to provide a Linear Quadratic Regulator (LQR) controller, which maintains the pendulum in its unstable equilibrium position on a cart with just one control input. Simulation results demonstrate successful stabilization by the LQR controller.

(V. R. Molazadeh, A. Banazadeh, and I. Shafieenejad, 2014) [10] applied intelligent tools for the TIP to tune LQR parameters such as Genetic Algorithm (GA), Genetic Algorithm with Practical Swarm Optimization (GA-PSO), and FLC. According to simulation findings, FLC performs much better in terms of overshoot, settling time, and parameter change response.

(H. G. Kamil, E. E. Eldukhri, and M. S. Packianather, 2014) [11] used a Discrete-time Linear Quadratic Regulator (DLQR) to balance the Robogymnast. The simulation results indicated successful stabilization and balancing of the robot gymnast.

(H. A. Ismail, M. S. Packianather, R. I. Grosvenor et al., 2015) [12] used Invasive Weed Optimization (IWO) to find the optimal Q matrix for the LQR controller. The optimized parameters yielded a substantially shorter settling period compared to control action without optimization, although there was a risk of control voltage reaching a saturation limit that could damage the system.

(D. C. Dracopoulos and B. D. Nichols, 2017) [13] utilized a method to address the Acrobot's swinging and balancing issues, yielding favorable outcomes,

especially in the swing-up task. However, the balance controller could not stabilize the acrobot after the swing-up controller was applied.

(N. F. Jamin and N. A. Ghani, 2017) [14] discussed using FLC-PSO to model a wheelchair with two wheels. The model was tested using an FLC controller, and the results suggest that the system can be stable in an upright posture with satisfactory simulation results. The outcome demonstrates that the system performs better when used with PSO to obtain the system's ideal values for settling time, rising time, peak overshoot, and peak undershoot.

(H. A. Ismail, M. S. Packianather, and R. I. Grosvenor, 2017) [15] discussed the efficacy of the multi-objective Invasive Weed Optimization (IWO) in producing an LQR controller that considers both the cost function and settling time. They optimized the cost function and settling time values of the Weight Criteria Method (WCM) using IWO in the initial optimization technique. The second optimization approach involved a hybrid IWO, which incorporates fuzzy logic to determine a membership value as the fitness criterion. Then, trained controllers were subjected to limited disturbances. Despite being subjected to external disturbances; all controllers could balance the Robogymnast upright.

(T. Yaren and S. Kizir, 2018) [16] applied LQR and Linear-quadratic-Gaussian (LQG) to the TIP. Simulation results indicate successful noise reduction, making LQG control significantly superior to the LQR technique.

(R. Banerjee, N. Dey, U. Mondal et al., 2018) [17] the PID and LQR controllers were used to describe and control a Double Inverted Pendulum

(DIP) on the cart dynamic system. Comparing the effectiveness of these two control systems is the objective of the study. The outcomes confirm the LQR control approach's relative superiority over the traditional PID control strategy for the DIP on the cart system.

(X. Xia, J. Xia, M. Gang et al, 2020) [18] suggested a new algorithm using Logistic chaotic variables to simplify selecting quantization and proportion factors in fuzzy controllers. Integrating chaotic variables in the search process helps find suboptimal solutions faster, improving control of a double-inverted pendulum model, as shown in simulation results.

(A. F. Ghalib and A. A. Oglah ,2020) [19] focused on applying a fuzzy-PID (FPID) controller to control an IP to maintain the pendulum arm's upright position by regulating the cart's location. Several evolutionary optimization techniques are used to optimize the controller's parameters, such as the GA, Social Spider Optimization, and ant colony optimization. Results show that FPID with Social Spider Optimization performs better than conventional PID.

(M. A. Ebrahim, M. E. Mousa, E. M. Said et al, 2020) [20] applied a new Grey Wolf optimizer (GWO) and PSO for the IP system. The Reduced Linear Quadratic Regulator and Variable Structure Adaptive Fuzzy controller parameters are adjusted using the suggested GWO/PSO approach to stabilize the cart posture and the pendulum angle. Compared to conventional LQR, the Reduced Linear Quadratic Regulator performs well for both the cart position and the pendulum angle with fewer parameters needed to achieve the required response.

(M. K. Habib and S. A. Ayankoso, 2020) [21] presented the LQR controller and pole placement design for a DIP model's stabilization. The Q and R matrices of the LQR controller were tuned using GA and PSO algorithms, GA and PSO algorithms were used to tune the Q and R matrices of the LQR controller. When comparing the transient performance of the manually and GA-tuned LQR and pole-placement controllers, the PSO-tuned LQR controller performed the best.

(H. G. Kamil, O. T. Makki, and H. M. Umran ,2020) [22] focused on using the PSO technique to identify the ideal LQR control parameters. These parameters were used to calculate the state feedback gains, and the IP was balanced in the upright equilibrium position using the best gain.

(N.-K. Nguyen, V.-N. Pham, T.-C. Ho et al. 2022) [23] addressed the control of an IP to maintain the rods vertically while regulating the cart to follow a desired trajectory within an acceptable tolerance. The suggested control approach combines PSO with two traditional PID controllers. Results from experiments on a working prototype of the inverted pendulum system and simulations on a Simulink model indicate the performance and viability of the control technique.

(N.-K. Nguyen et al, 2022) [24] introduced a novel control approach for balancing of an IP. The suggested approach combines a modified genetic algorithm (mGA) with a PD-like fuzzy logic architecture to maximize the fuzzy logic controller's scaling factors. To improve the IP's balancing control system, the mGA is used to optimize six important scaling factors that correspond to two fuzzy logic controllers. The findings from numerical

simulations and actual tests done on a genuine IP system illustrate the prospective application and efficacy of the suggested control approach compared to PID and current fuzzy logic alternatives.

(A. Mourad, Y. Zennir, and C. Tolba, 2022) [25] presented a comparison of integral sliding mode control adjusted utilizing the WOA, radial basis function neural network, and FLC for the control of the angle position and velocity of the inverted pendulum system. WOA was used to adjust all the parameters and effects of those controllers. According to comparison data, integral sliding mode control based-WOA performs better than other approaches regarding settling time and overshooting.

(M. Mohamed, F. Anayi, M. Packianather, et al. 2022) [26] designed and simulated PID and LQR controllers for the Robogymnast. Contrasting the PID controller's performance with that of the well-established LQR controller, indicating the PID controller's superior appropriateness for the particular robot under investigation in contrast to a controller intended for broad usage.

(B. A. Samad, F. Anayi, Y. Melikhov et al, 2023) [27] simulated an LQR/FLC for the Robogymnast, comparing the performance of a Fuzzy Linear Quadratic Regulator (FLQR) controller with a conventional LQR controller, demonstrating the FLQR's more outstanding suitability for the specific robotic system examined.

(B. A. Samad, M. Mohamed, and G. S. Member, 2023) [28] aimed to improve the performance of an FLQR by stabilizing the triple-link "Robogymnast" robotic system through the use of Teaching-Learning-Based Optimization

(TLBO) and PSO techniques. The study's findings suggest that, when used in conjunction with the Robogymnast robotic system, the FLQR controller and TLBO algorithm perform better than other controllers and combinations of algorithms.

(O. Saleem and J. Iqbal, 2023) [29] introduced a fuzzy-immune adaptive system that modifies the Degree-of-Stability (DoS) of a LQR process to improve a self-balancing mechatronic system's ability to attenuate disturbances. The closed-loop system's eigenvalues are dynamically relocated in the left half of the complex plane by the system using pre-configured control input-based rules, which modify the LQR gains. This makes it possible to manipulate reaction times and control efforts flexibly as error conditions vary. Hardware-in-the-loop studies on the Quasar rotary inverted pendulum system are used to verify the effectiveness of the system, and the results demonstrate a notable increase in the system's disturbance attenuation capabilities when compared to the DoS-LQR.

(T.ABUT 2023) [30] used optimum LQR control techniques combined with classical methods to model and ideally regulate a DIP system on a Cart (DIPSC). Using the GA, PSO, and GWO algorithms, the Q and R values of the LQR control approach were determined. Mean-Square-Error (MSE) performance criteria and settling time were used to examine and display the graphical results of the evaluation of the DIPSC system using both traditional LQR and optimum LQR approaches. The purpose of the controls was to lead the cart to the predetermined balance position and keep the DIP's arms vertically balanced while it moved. In terms of settling time and MSE error

criterion, the GWO-based LQR control technique fared better than the other approaches.

(S. Erjon, B. Xhevahir, L. Rame et al, 2023) [31] introduced a method to develop a real-time control system for a double-inverted pendulum by combining Proportional-Integral-Derivative (PID) and LQR controllers. Real-time simulation results demonstrated the successful swinging up and stabilization of the double inverted pendulum by both the PID and LQR controllers.

(N. A. Sayer, G. Kamil, and A. A. Al-Moadhen, 2023) [32] addressed the control of a gymnastic robot's balance using DLQR and LQG control techniques. The controllers displayed the ideal values for vertical robot stabilization in a suitable duration of time. DLQR outperformed the LQG controller, as the latter consumed excessive energy to maintain satisfactory performance, leading to saturation of the first motor.

Table 2.1 illustrate various control strategies utilized to stabilize and regulate IP, allowing the pendulum to remain in its upright position, reducing of oscillations, and enhancing of system responsiveness.

Table 2-1 General overview based on Type of System and Controller

Ref	System Type	Type of Controller	Result
[8]	Robot Arm	Fuzzy Logic and Fuzzy Supervisor	Steady state error of motor five was minimized from 0.03 to 0.001, overshoot size reduced

		with Proportional Integral Derivative	from 0.08 to 0.001 and the rising time for the FSC was 50% less than PID controller
[9]	Triple Inverted Pendulum	Linear Quadratic Regulator	Settling time for $\theta_1, \theta_2, \theta_3$ is around 3 seconds
[10]	Triple Inverted Pendulum	Applied intelligent tools to tune Linear Quadratic Regulator parameters such as Genetic Algorithm, Genetic Algorithm with Practical Swarm Optimization, and Fuzzy Logic	Settling Time 4% and Over Shoot 0°
[11]	Robogymnast	Discrete-time Linear Quadratic Regulator	The first and second links reached the steady state after 6 seconds whilst the third link doing that in 9 seconds
[12]	Robogymnast	Invasive Weed Optimization to find the optimal Q matrix for the Linear Quadratic Regulator	Settling time 5.4 sec and overshoot $30^\circ, 68^\circ$ and 10° for first, second and third link, respectively

[13]	Acrobot	Genetic Algorithm	Settling time 1 sec and no overshoot
[14]	wheelchair	Practical Swarm Optimization and Fuzzy Logic	Settling time 2.778 sec for first link and 2.647 sec for second link
[15]	Robogymnast	Invasive Weed Optimization with Fuzzy Logic to find the optimal Q matrix for the Linear Quadratic Regulator	Settling time 6.37 sec and overshoot 30° , 70° and 10° for first, second and third, respectively
[16]	Triple Inverted Pendulum	Linear Quadratic Gaussian and Linear Quadratic Regulator	Settling time 2.5 sec
[17]	Double Inverted Pendulum	Proportional Integral Derivative and Linear Quadratic Regulator	Settling time 4 sec, overshoot 60° for first link and 20° for second link
[18]	Double Inverted Pendulum	Logistic Chaotic Algorithm and Fuzzy Logic	Settling time 4 sec, overshoot 0.12°
[19]	Inverted Pendulum	Used Genetic Algorithm, Social Spider Optimization, and ant colony optimization to	Settling time 2.3803 sec and Overshoot 400920%

		optimize the Fuzzy Logic with Proportional Integral Derivative	
[20]	Inverted Pendulum	Linear Quadratic Regulator and Fuzzy parameters are adjusted using the Grey Wolf Optimizer and Practical Swarm Optimization	Settling time 4.36 sec and Overshoot 0.0942°
[21]	Double Inverted Pendulum	Genetic Algorithm and Practical Swarm Optimization tuned Linear Quadratic Regulator and pole-placement	Settling time 0.7512sec and 1.0245sec, overshoot 0.3094° and 0.0969° for first and second link respectively
[22]	Inverted Pendulum	Practical Swarm Optimization tuned Linear Quadratic Regulator	Settling time 2sec and overshoot 0.05°
[23]	Inverted Pendulum	Practical Swarm Optimization tuned Proportional Integral	Settling time 2sec and overshoot 0.2°

		Derivative parameters	
[24]	Inverted Pendulum	Modified genetic algorithm with a Proportional Derivative -like Fuzzy Logic	Settling time 2sec and overshoot 4°
[25]	Inverted Pendulum	Integral sliding mode control adjusted utilizing the Whale Optimization Algorithm, radial basis function neural network	Settling time 2sec and overshoot 0°
[26]	Robogymnast	Proportional Integral Derivative and Linear Quadratic Regulator	Settling time 15.519sec, 4.914sec and 3.331sec, overshoot 8.02°, 1.32° and 0.41° for first, second and third, respectively
[27]	Robogymnast	Linear Quadratic Regulator and Fuzzy Logic	Settling time 11.1823sec, 4.1694sec and 2.4428sec, overshoot 2.88°, 1.44° and 0.4° for first, second and third, respectively
[28]	Robogymnast	Teaching-Learning-Based Optimization	Settling time 5.577sec, 2.1668sec and 1.9675sec,

		and Practical Swarm Optimization tuned Linear Quadratic Regulator with Fuzzy Logic parameters	overshoot 5.67° , 1.32° and 0.4° for first, second and third, respectively
[29]	Quasar Rotary Inverted Pendulum	Linear Quadratic Regulator	-
[30]	Double Inverted Pendulum	Genetic Algorithm, Practical Swarm Optimization, and Gray Wolf Optimization tuned Linear Quadratic Regulator parameters	Settling time 1.5sec for first link and 1.2sec for second link
[31]	Double Inverted Pendulum	Proportional Integral Derivative with Linear Quadratic Regulator	Settling time 2sec and no overshoot
[32]	Robogymnast	Discrete Linear Quadratic Regulator and Linear Quadratic Gaussian	overshoot (-4° , 9° , 1.1°), and settling time (3.208s, 3.233s, 4.16s) for the three links (first, second, third)

Table. 2.1 overviews relevant publications on single, double, and triple IP systems and their applications, focusing on different control strategies. The research closely related to robot gymnastics includes references [11], [12], [15] and [32].

2.3 Discrete Linear Quadratic Regulator

A discrete linear quadratic regulator is a vital control unit that controls the system's response and energy consumption to achieve the reaction with the lowest control voltage and obtain a stable system. It uses a state space approach, and since it is not a dynamic system, the system's order is the same as that of a closed-loop feedback system. DLQR is used to option feedback gain state [33].

The principle of a linear quadratic regulator is to minimize a cost function as follows:

$$J = \int_0^{\infty} (x^T Q x + u^T R u) dt \quad (2.1)$$

Q and R are the weighted diagonal matrix that must be positive definite.

$$\dot{x} = (A - BK)x \quad (2.2)$$

Where A and B State Space matrix derived in section 3.3 and equal to the ENN weights Wcx and Who respectively that defined in section 3.4 when using the ENN model.

The feedback control law is as follows:

$$u(t) = -Kx(t) \quad (2.3)$$

K is the gain vector and is given by:

$$K = R^{-1}B^T P \quad (2.4)$$

Using the Algebraic Riccati Equation below, the value of P can be obtained:

$$A^T P + PA - PBR^{-1}P + Q = 0 \quad (2.5)$$

2.4 Whale Optimization Algorithm (WOA)

In 2016, Seyed Ali Mirjalili proposed the WOA meta-heuristic schema [34]. This algorithm simulates hunting humpback whales using the bubble net strategy. It specifies using a basic mathematical model. The suggested method may be executed in three primary phases: encircling prey, exploitation phase (attacking prey), and exploration phase (searching prey). During the search and encircling phase, the humpback whales select the agent with the most incredible score of a random agent (to avoid the local minimum) as a target point, updating their position with the surroundings of this point [25]. The humpbacks travel in a circular or spiral pattern toward the target position after selecting the best agent (as shown in section 2.4.1).

2.4.1 Pseudo-code of the WOA algorithm

1. Input

- objective function (fitness function), search space boundary, population size N , number of iterations T_{max} , number of Variable No. Variable.

2. Initialization

- Initialize random whale's population X_i ($i = 1, 2, \dots, n$) with search space boundary.
- Calculate the fitness of each search agent
- X^* =the best search agent.

3. while (While $T < T_{max}$)

for $i=1: N$

for $j=1: \text{No. Variable}$

 Update a, A, C, l , and p

if₁ ($p < 0.5$)

if₂ ($|A| < 1$)

 Update the position of the current search agent by the Eq. (2.6)

else if₂ ($|A| > 1$)

 Select a random search agent (X_{rand})

```

        Update the position of the current search agent by the Eq. (2.12)
    end if2
        else if1 (p >0.5)
            Update the position of the current search by the Eq. (2.10)
        end if1
    end for
    Check X* within the boundary and amend it
    Calculate the fitness of each search agent
    Update X* if there is a better solution
    t=t+1
    end while
4. print X*
    
```

Step 1: Initialization

Initially, WOA variables include the number of search agents (No-whales set to 50 agents). Then, DLQR controller parameters such as Q and R are set with random values selected by the predefined search space to start the optimization process, as shown in Table 2.2.

Table 2-2: Parameters of Optimization

Variable	Description	Value
Number of Variables	Number of parameters that should be adjusted by optimization.	8
Number of agents	Number of agent population of solutions	50
Initial Standard Deviation Value	Initial input states	[3°3°3°]
Search Range	The search range is founded depending on the designer’s experience.	Q= 1- 10000 R = 0.1- 5

Table. 2.1 defines the WOA algorithm's important parameters specified by the designer's experience.

Step 2: fitness function calculation

The WOA algorithm obtains the Q and R that achieve the best performance to balance the system upright. The calculated gain by using DLQR for each search agent and choosing the one with a minimum deviation of the first link from the upright balancing point with minimum overshoot and settling time as a leader agent for the next step (update the position).

Step 3: update the position

a) Encircling

The humpback whales first begin the encircling process after discovering their prey, but the ideal location inside a specific search space is unknown; thus, this algorithm selects the current prey position as the best prey position (optimal solution), after which the other agents will update their positions following the best solution. Equation (2.6) and (2.7) provide the mathematical equation of the encircling process [34].

$$D = |CX_{best}(t) - X(t)| \quad (2.6)$$

$$X(t+1) = X_{best}(t) - A.D \quad (2.7)$$

A and C are coefficient vectors.

t: denote the current iteration.

X: location vector.

X_{best} : is the location vector of the best currently-found solution that should be adjusted in each loop if there is a better solution.

$$A = 2a \cdot r - a \quad (2.8)$$

$$C = 2r \quad (2.9)$$

a: decrease linearly during the period of iterations from 2 to 0.

r: a random vector in [0,1].

b) Attacking prey

This step explains the bubble-net method. Humpback whales attack and surround their prey simultaneously with a narrowing circle and a spiral movement shape toward the prey (target solution). This algorithm suggests a 50% chance of picking between the two types of movement to mimic this type of behavior. Equation (2.10) provides the bubble-net technique.

$$X(t + 1) = \begin{cases} X_{best}(t) - A \cdot D & \text{if } p < 0.5 \\ D e^{bl} \cos(2\Pi l) + X_{best}(t) & \text{if } p \geq 0.5 \end{cases} \quad (2.10)$$

l : a random number between [-1 1].

P : a random number with a range [0 1].

b : a constant used to describe the logarithmic spiral form.

Where $D = |X_{best}(t) - X(t)|$ that described the distance between the current agent and target.

c) Exploration (searching for prey)

This step discusses the exploration capability of the WOA algorithm. For describing the exploration, the search agent moves far away from the goal (Reference whale) if the value of A is more significant than 1 or less than -1. In contrast to exploitation, each agent's location is updated concerning an agent that is selected randomly (when $|A| > 1$); this is explained in the equation. (2.11) and (2.12).

$$D = |CX_{rand} - X| \quad (2.11)$$

$$X(t + 1) = X_{rand} - A \cdot D \quad (2.12)$$

X_{rand} ; is a random position vector (a random whale selected from the present population).

Fig. 2.1 demonstrates the flow chart of WOA steps.

2.5 Aquila Optimization (AO) Algorithm

Abualigah, L. et al. proposed the AO algorithm in 2021 as a typical SI method that mimics the hunting behavior of the Aquila [35]. Aquila Optimizer (AO) focuses on three significant steps, another population-based algorithm: initialization, exploration, and exploitation, as shown in section 2.5.1 [36].

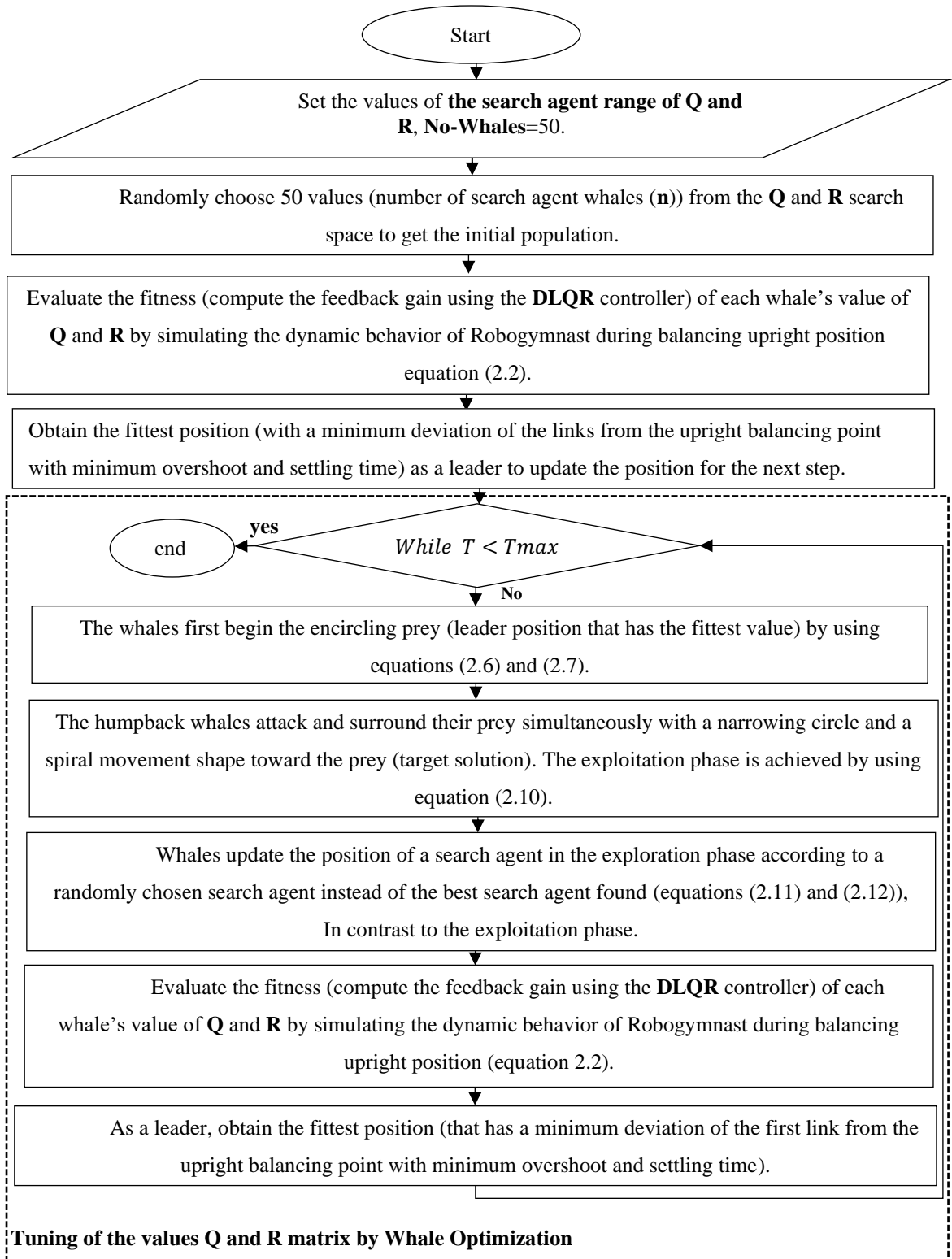


Fig. 2.1. Flow chart of the whale optimization.

2.5.1 Pseudo-code of the AO algorithm

1. Input

- objective function (fitness function), search space boundary, population size N , number of iterations T_{max} , number of Variable No. Variable.

2. Initialization

- Initialize random Aquila's population X_i ($i = 1, 2, \dots, n$) with search space boundary.
- Calculate the fitness of each search agent
- X_{best} =the best search agent.

3. while (*While* $T < T_{max}$)

for $i=1: N$

for $j=1: \text{No. Variable}$

Update parameters $X_m, Levy(D), G_1, G_2, X,$ and Y .

Step 1: X_1 : **Expanded Exploration**

Update the position of the current search agent by the Eq. (2.16)

$$X_{best}(t)=X_1(t+1)$$

Step 2: X_2 : **Narrowed Exploration**

Update the position of the current search agent by the Eq. (2.18)

$$X_{best}(t)=X_2(t+1)$$

Step 3: X_3 : **Expanded Exploitation**

Update the position of the current search agent by the Eq. (2.25)

$$X_{best}(t)=X_3(t+1)$$

Step 4: X_4 : **Narrowed Exploitation**

Update the position of the current search agent by the Eq. (2.26)

$$X_{\text{best}}(t) = X_4(t+1)$$

End for

End for

Check Xbest within the boundary, amend it, and calculate the fitness of each search agent

Update X_{best} if there is a better solution

$$t = t + 1$$

end while

4. print X_{best}

Step 1: Initialization

The population of possible solutions (X_{ij}) is generated randomly between the upper UB_j and lower LB_j boundary as shown in equations (2.13) and (2.14) [35] and initialed another parameter of AO as shown in Table 2.2.

$$X = \begin{bmatrix} x_{1,1} & \cdots & x_{1,j} & x_{1, \text{Dim}-1} & x_{1, \text{Dim}} \\ x_{2,1} & \cdots & x_{2,j} & \cdots & x_{2, \text{Dim}} \\ \cdots & \cdots & x_{i,j} & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{N-1,1} & \cdots & x_{N-1,j} & \cdots & x_{N-1, \text{Dim}} \\ x_{N,1} & \cdots & x_{N,j} & x_{N, \text{Dim}-1} & x_{N, \text{Dim}} \end{bmatrix} \quad (2.13)$$

$$X_{ij} = \text{rand} (UB_j - LB_j) + LB_j, i = 1, 2, \dots, N, j = 1, 2, \dots, \text{Dim} \quad (2.14)$$

Where m is the number of agents of the population, Dim is the number of variables that should be adjusted by optimization (Q and R matrix), and $rand$ is the random number between 0 and 1.

Step 2: Exploration

In this step, the optimization generates random agents for exploring different search space regions. Equation (2.15) described expanded exploration where the Aquila identifies the area of the prey and chooses the optimal hunting area:

$$X_1(t + 1) = X_{best}(t) \left(1 - \frac{t}{T}\right) + (X_M(t) - X_{best}(t) * rand) \quad (2.15)$$

$$X_M(t) = \frac{1}{N} \sum_{i=1}^N X_i(t), \forall j = 1, 2, \dots, Dim \quad (2.16)$$

Where X_{best} is the best solution of the adjusted variable until t^{th} iteration, $(1 - t/T)$ controls the search during the exploration phase, T is the total number of generations, and X_M is the average of the search agent.

In the second strategy of exploration (narrowed exploration), the Aquila flies in a spiral above the prey before attacking through a quick glide. The narrowed exploration is described as follows:

$$X_2(t + 1) = X_{best}(t) Levy(D) + X_R(t) + (y - x) * rand \quad (2.17)$$

At the i^{th} iteration, $X_R(t)$ is a random solution selected from the interval $[1 N]$, the dimension of space is D , and $Levy(D)$ is the distribution function for levy flights and calculated as follows:

$$\text{Levy } (D) = s \frac{u \sigma}{|v|^{\frac{1}{\beta}}} \quad (2.18)$$

Where u and v are random numbers between 0 and 1, s is a constant value set to 0.01, β is a constant set to 1.5, and σ is determined by applying equation (2.18):

$$\sigma = \left(\frac{\Gamma(1 + \beta) \sin \left(\frac{\pi\beta}{2} \right)}{\Gamma \left(\frac{1 + \beta}{2} \right) \beta 2^{\left(\frac{\beta - 1}{2} \right)}} \right) \quad (2.19)$$

The spiral shape in the search is shown in equation (2.16) using the variables y and x :

$$y = r \cos (\theta) \quad (2.20)$$

$$x = r \sin (\theta) \quad (2.21)$$

$$r = r_1 + U + D_1 \quad (2.22)$$

$$\theta = -\omega D_1 + \theta_1 \quad (2.23)$$

$$\theta_1 = \frac{3\pi}{2} \quad (2.24)$$

The value of U is a small number fixed to 0.00565, and the value of r_1 ranges from 1 to 20 for a predetermined number of search cycles. D_1 consists of integers from 1 to the search space's length (Dim), and ω is a small number set to 0.005.

Step 3: Exploitation

In this step, AO exploits the target's selected region to get close and attack the prey, as described in the following equation.

$$X_3(t + 1) = (X_{\text{best}}(t) - X_M(t)) \alpha - \text{rand} + ((UB - LB) \text{rand} + LB) \delta \quad (2.25)$$

Where α and δ are small values between 0 and 1.

When the Aquila approached the prey, it attacked it over the land following its stochastic movements. Equation (2.25) provides a mathematical representation of this behavior.

$$X_4(t + 1) = Q F X_{\text{best}}(t) - (G_1 X(t) \text{rand}) - G_2 \text{Levy}(D) + \text{rand } G_1 \quad (2.26)$$

Equation (2.26) is used to calculate a quality function called QF that is utilized to balance the search strategies.

$$Q_f = t \frac{2 \text{rand} - 1}{(1-T)^2} \quad (2.27)$$

Equation (2.27) is utilized to generate G_1 , which represents various AO motions used for tracking the prey through the elope.

$$G_1 = 2 \text{rand} - 1 \quad (2.28)$$

G_2 represents a parameter that decreases from 2 to 0, and the following equation is used to update it:

$$G_2 = 2 \left(1 - \frac{t}{T}\right) \quad (2.29)$$

Fig. 2.2 demonstrates the flow chart of the Aquila optimization algorithm.

2.6 Fuzzy Logic Control (FLC)

In 1965, Lofti Zadeh suggested fuzzy logic, essentially based on the concept of the fuzzy set. Fuzzy logic builds a controller based on expressions instead of equations [37]. It mimics human expertise by defining a fuzzy set as a class of objects with a membership grade continuum that may be described by a membership function that assigns an actual number between [0,1] to each point, where 1 denotes a greater degree of membership, and 0 is a lower degree, making it a desirable method of control for issues that are challenging to measure mathematically [38], [39].

To build a fuzzy controller, first specify input variables; after understanding the system, we found that the output consists of three angles, three angular velocities, and two inputs, represented by the voltages of the two motors. When designing the DLQR controller, we observed that the responses to the three output angles change between -3 and 3 and that the angular velocities of these angles change between -10 and 10. whereas the input voltages change from -10 to 10. Therefore, the range of inputs and outputs in the design of the fuzzy system was chosen based on these results. Then, define the membership functions' number and shape for inputs and output. The two inputs, as well as the output, may have the following linguistic variables: Positive-Big (PB), Positive-Medium (PM), Positive-Small (PS), Zero (Z), Negative-Big (NB), Negative-Medium (NM), and Negative- Small (NS). There are several different kinds of membership functions, including Gaussian, Triangle, Gaussian two, and Trapezoidal membership functions [40]; all of these types have been applied to the system, but the triangle membership function was

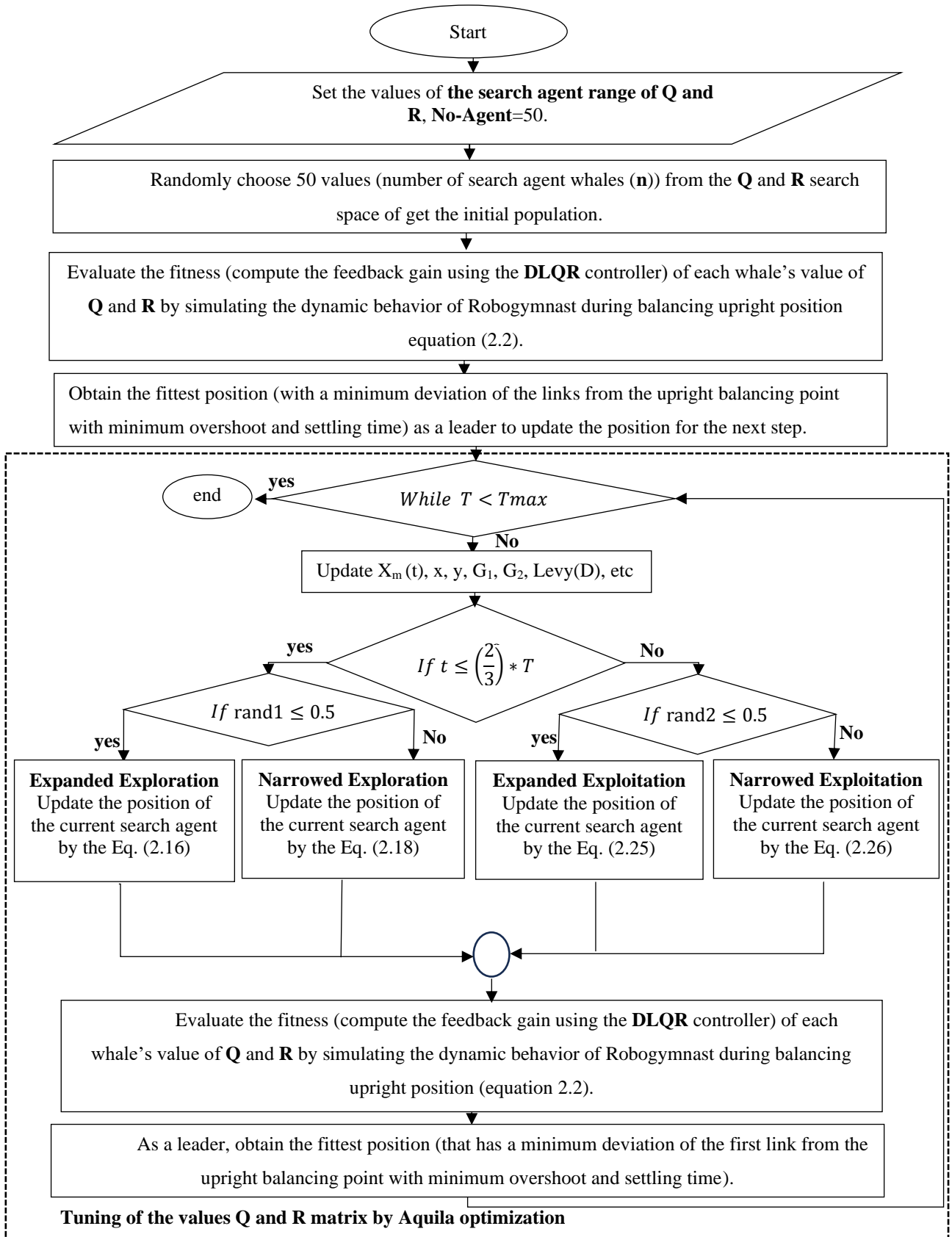


Fig. 2.2 Flow chart of Aquila optimization algorithm

selected above the other forms of membership functions because it produces smooth inputs and outputs with less control effort required at each stage (as shown in Fig. 2.3). So, the fuzzy controllers were created using 49 rules (as shown in Table 2.3). Seven membership functions with a base of 49 rules were used to develop the appropriate system tuning of the FLC to stabilize the model.

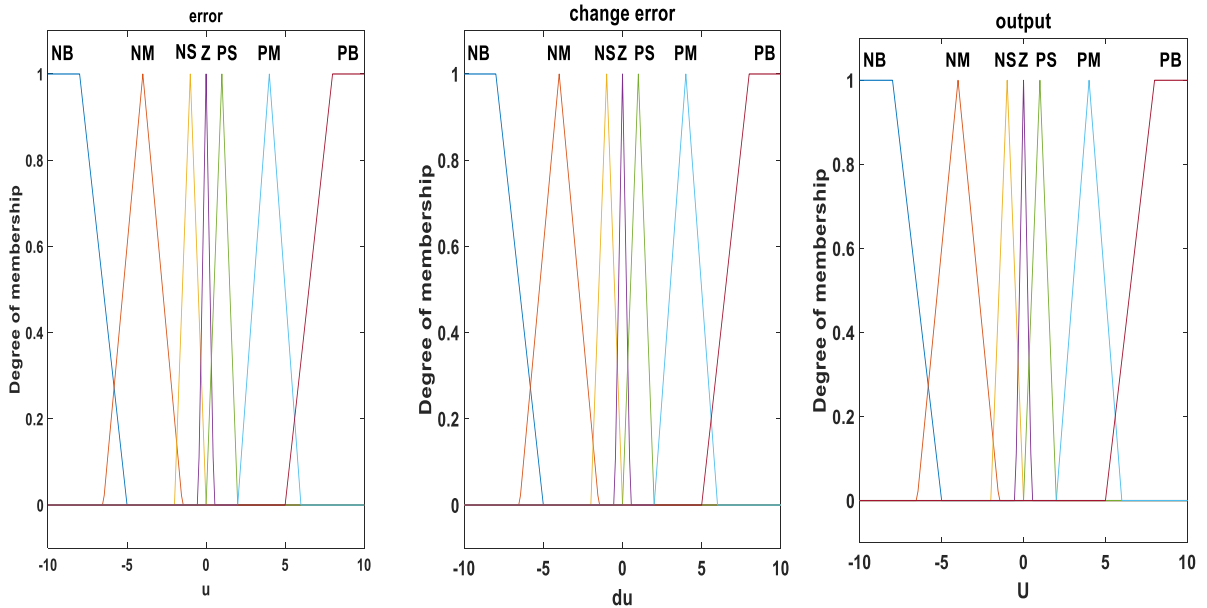


Fig. 2.3 Membership function with two inputs and one output.

Table 2-3 Fuzzy Logic rules base

$e \hat{e}$	NB	NM	NS	Z	PS	PM	PB
NB	NB	NB	NB	NM	NM	NS	Z
NM	NB	NB	NB	NM	NS	Z	PS
NS	NB	NM	NM	NS	Z	PS	PM
Z	NM	NM	NS	Z	PS	PM	PB
PS	NM	NS	Z	PS	PM	PM	PB
PM	NS	Z	PS	PM	PM	PB	PB
PB	Z	PS	PM	PM	PB	PB	PB

2.7 Summary

A review of different designs of single and multi-link underactuated systems has been presented. This chapter gave an overview of various controller methods that have been used to satisfy different types of complex n-link robot system locomotion focusing on the literature for balancing control. In addition, the literature of using an optimization technique to achieve the control performance has been presented. In the next chapter, the description of the Robogymnast system is given and a mathematical model for the Robogymnast will be derived.

Chapter Three: **The System Description and Mathematical Model**

3.1 Introduction

The Robogymnast system description and its mathematical model derivation will be discussed in this chapter. According to the actuation level, the mechanical system is divided into three essential categories. The first one is called a completely actuated system when the system has the same number of actuators as the DoFs, and each is controlled independently [41], [42], [43]. The second one is called an over-actuated system when it has more actuators than the DoFs [44]. The third one is called an underactuated system when it has fewer actuators than the DoFs [45]. The Robogymnast has three links and three joints, so it has three degrees of freedom (three angles). The first joint is passive, while the second and third are actuated. The underactuated mechanical system provides several benefits, such as less weight, less tendency to break down, and lower energy use [46], [47].

The rest of this chapter is organized as follows: A description of the system is presented in Section 3.2. Section 3.3 demonstrate the system's state space model derivation from the Euler-Lagrange equation. Section 3.4 determines the Elman neural network modeling. Finally, section 3.5 provides a summary of the chapter.

3.2 System Description

The Robogymnast system, illustrated in Fig. 3.1. It is designed to mimic human acrobatics and modeled as a three-link under-actuated pendulum. The

physical specifications of Robogymnast are carefully tailored to closely mimic the movements of a human gymnast gripping a freely rotating high bar with a tight grip. Each link in the system corresponds to a specific human body part or group of body parts. The first link represents the arms, excluding the elbows and wrists, while the second represents the head and torso. The third link represents the legs. A potentiometer is mounted on each joint's steel shaft to measure the links' relative angles. The second section of joints 2 and 3 consists of the output shaft of the power unit, which is a DC motor with a gearbox [4], [5].

3.3 Robogymnast Mathematical Model in the Upward Position Derived by Euler-Lagrange

The Robogymnast is shown schematically in Fig. 3.2. It is considered a TIP with an unstable balance state and derived using the Euler-Lagrange formula. The Euler-Lagrange method is the most commonly used technique for obtaining the dynamical equations of several dynamic systems. The essential part of the Lagrange equation is achieving the total system's potential, dissipation energy, and kinetic energy.

The Euler-Lagrange equations [48] and [49] are used to derive the mathematical model:

$$\frac{d}{dt} \left(\frac{\partial K}{\partial \dot{\theta}_i} \right) - \frac{\partial K}{\partial \theta_i} + \frac{\partial D}{\partial \dot{\theta}_i} + \frac{\partial P}{\partial \theta_i} = T_i \quad i = \text{integer number} \quad (3.1)$$

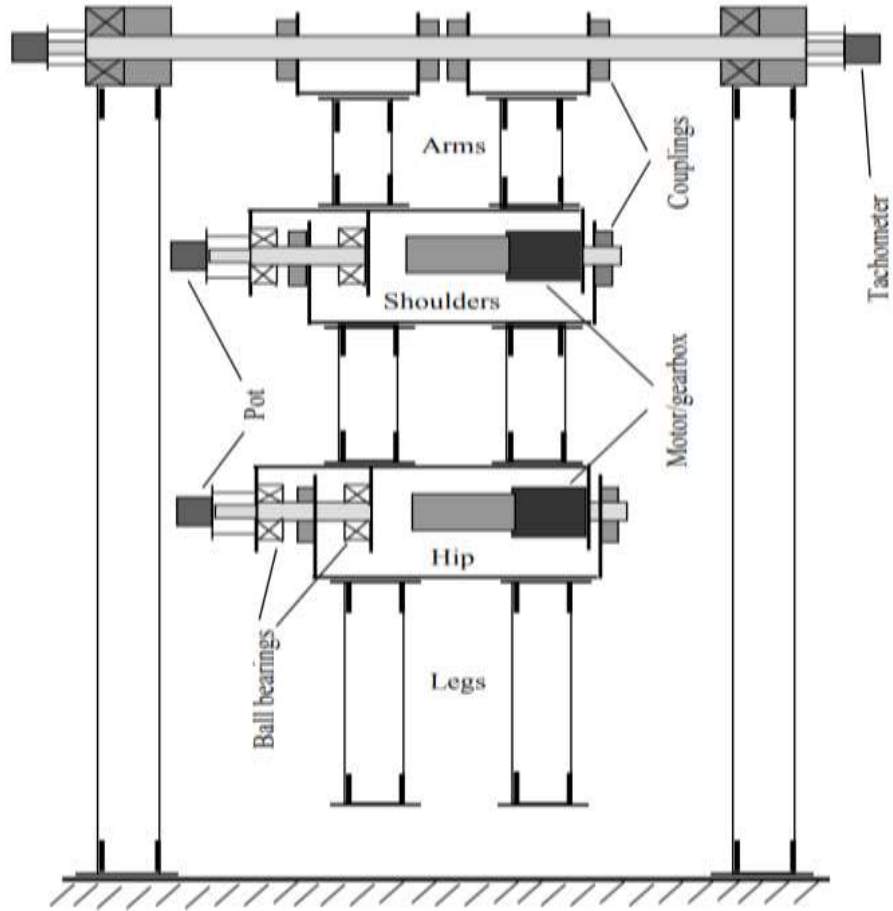


Fig. 3.1 Hardware components of Robogymnast [5]

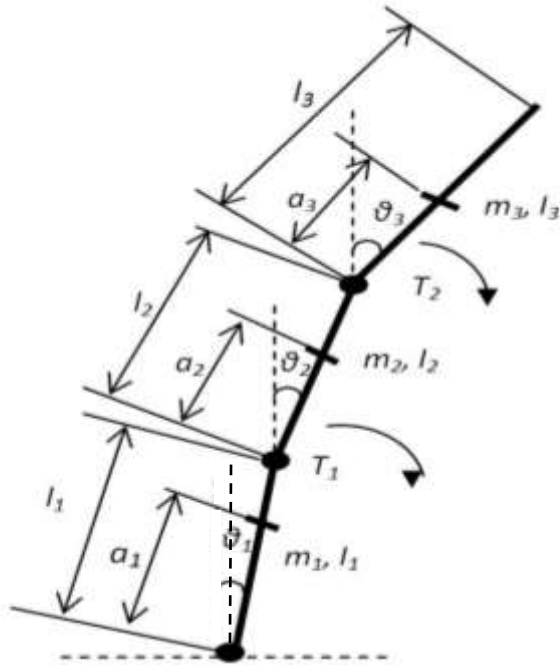


Fig. 3.2 Robogymnast schematic in the upright posture [5]

K is the kinetic energy, P is the potential energy, and D is the dissipation energy. It stands for the angle of the respective link, calculated concerning the vertical line, and T_i denotes the torque related to it.

$$K = \frac{1}{2} \sum_{i=1}^3 \left\{ \begin{array}{l} I_i \dot{\theta}_i^2 + m_i \left[\frac{d}{dt} \left(\sum_{k=i-3}^{i-1} l_k \sin(\theta_k) + a_i \sin(\theta_i) \right) \right]^2 + \\ \left[\frac{d}{dt} \left(\sum_{k=i-3}^{i-1} l_k \cos(\theta_k) + a_i \cos(\theta_i) \right) \right]^2 \end{array} \right\} \quad (3.2)$$

$$P = \sum_{i=1}^3 m_i g \left(a_i \cos(\theta_i) + \sum_{k=i-3}^{i-1} l_k \cos(\theta_k) \right) \quad (3.3)$$

$$D = \frac{1}{2} \sum_{i=1}^3 (c_i (\theta_i - \theta_{i-1})^2) \quad (3.4)$$

l_i is the i^{th} link's length, m_i is the i^{th} link's mass, I_i is the moment of inertia around its center of gravity, a_i is the i^{th} link's center of gravity, c_i is the i^{th} joint's viscous friction coefficient, and the acceleration brought on by gravity is known as g .

The torques given to the second and third actuated joints have an impact on the first joint, which is underactuated. The following equation represents the torques of the two DC motors at the second and third joints:

$$T_{m_1} = G_1 u_1 - I_{p_1} (\ddot{\theta}_2 - \ddot{\theta}_1) - C_{p_1} (\dot{\theta}_2 - \dot{\theta}_1) \quad (3.5)$$

$$T_{m_2} = G_2 u_2 - I_{p_2} (\ddot{\theta}_3 - \ddot{\theta}_2) - C_{p_2} (\dot{\theta}_3 - \dot{\theta}_2) \quad (3.6)$$

were

$$T_1 = -T_{m_1}, T_2 = T_{m_1} - T_{m_2}, T_3 = T_{m_2}$$

The input voltage to the DC motors is represented by u_1 and u_2 ($|u_1|$ and $|u_2| \leq 10V$). G_i is the i_{th} motor's static gain, I_{p_i} is the i_{th} motor's moment of inertia reflected at the gearbox's output shaft, and C_{p_i} is the i_{th} motor's viscous friction coefficient reflected at the gearbox's output shaft.

The total potential energy of the system is described as follows:

$$P = g[m_1 a_1 + m_2 l_1 + m_3 l_1] \cos(\theta_1) + g[m_2 a_2 + m_3 l_2] \cos(\theta_2) + m_3 g a_3 \cos(\theta_3) \quad (3.7)$$

In addition, the total dissipation energy of the system is described as follows:

$$D = \frac{1}{2} [C_1 + C_2] \dot{\theta}_1^2 + \frac{1}{2} [C_2 + C_3] \dot{\theta}_2^2 + \frac{1}{2} C_3 \dot{\theta}_3^2 - C_2 \dot{\theta}_1 \dot{\theta}_2 - C_3 \dot{\theta}_2 \dot{\theta}_3 \quad (3.8)$$

The total kinetic energy of the system is as follows:

$$\begin{aligned}
 K = & \frac{1}{2} [I_1 \dot{\theta}_1^2 + I_2 \dot{\theta}_2^2 + I_3 \dot{\theta}_3^2] + \frac{1}{2} [m_1 a_1^2 + m_2 l_1^2 + m_3 l_1^2] \dot{\theta}_1^2 \cos^2 (\theta_1) \\
 & + \frac{1}{2} [a_1^2 + 2l_1^2] \dot{\theta}_1^2 \sin^2 (\theta_1) \\
 & + \frac{1}{2} [m_2 a_2^2 + m_3 l_2^2] \dot{\theta}_2^2 \cos^2 (\theta_2) \\
 & + [m_2 l_1 a_2 + m_3 l_1 l_2] \dot{\theta}_1 \dot{\theta}_2 \cos (\theta_1) \cos (\theta_2) \\
 & + \frac{1}{2} [a_2^2 + l_2^2] \dot{\theta}_2^2 \sin^2 (\theta_2) \\
 & + [l_1 a_2 + l_1 l_2] \dot{\theta}_1 \dot{\theta}_2 \sin (\theta_1) \sin (\theta_2) \\
 & + \frac{1}{2} m_3 a_3^2 \dot{\theta}_3^2 \cos^2 (\theta_3) + \frac{1}{2} a_3^2 \dot{\theta}_3^2 \sin^2 (\theta_3)
 \end{aligned} \tag{3.9}$$

By resolving (3.1) for each system coordinate $[\theta_1 \theta_2 \theta_3]$, the motion equations of a system can be obtained as follows:

$$\frac{d}{dt} \left(\frac{\partial K}{\partial \dot{\theta}_1} \right) - \frac{\partial K}{\partial \theta_1} + \frac{\partial D}{\partial \dot{\theta}_1} + \frac{\partial P}{\partial \theta_1} = T_1 \tag{3.10}$$

$$\frac{d}{dt} \left(\frac{\partial K}{\partial \dot{\theta}_2} \right) - \frac{\partial K}{\partial \theta_2} + \frac{\partial D}{\partial \dot{\theta}_2} + \frac{\partial P}{\partial \theta_2} = T_2 \tag{3.11}$$

$$\frac{d}{dt} \left(\frac{\partial K}{\partial \dot{\theta}_3} \right) - \frac{\partial K}{\partial \theta_3} + \frac{\partial D}{\partial \dot{\theta}_3} + \frac{\partial P}{\partial \theta_3} = T_3 \tag{3.12}$$

Equations (3.7), (3.8), and (3.9) can be substituted for Equations (3.10), (3.11), and (3.12) to produce three differential equations that describe the system dynamics. The differential equations are nonlinear and can be linearized about the upward position ($\theta_i = 0$) to simplify the control system

analysis and design. Hence, the motion equations of the system are expressed as follows:

$$\begin{aligned}
 & [I_1 + m_1 a_1^2 + m_2 l_1^2 + m_3 l_1^2 + I_{p1}] \ddot{\theta}_1 + [m_2 l_1 a_2 + m_3 l_1 l_2 - I_{p1}] \ddot{\theta}_2 \\
 & \quad + [m_3 l_1 a_3] \ddot{\theta}_3 + [C_1 + C_2 + C_{p1}] \dot{\theta}_1 + [-C_2 - C_{p1}] \dot{\theta}_2 \\
 & \quad + [-gm_1 a_1 - gm_2 l_1 - gm_3 l_1] \theta_1 + G_1 u_1 = 0
 \end{aligned} \tag{3.13}$$

$$\begin{aligned}
 & [m_2 l_1 a_2 + m_3 l_1 l_2 - I_{p1}] \ddot{\theta}_1 + [I_2 + m_2 a_2^2 + m_3 l_2^2 + I_{p1} + I_{p2}] \ddot{\theta}_2 \\
 & \quad + [-I_{p2} + m_3 l_2 a_3] \dot{\theta}_3 + [-C_2 - C_{p1}] \dot{\theta}_1 \\
 & \quad + [C_2 + C_3 + C_{p1} + C_{p2}] \dot{\theta}_2 + [-C_3 - C_{p2}] \dot{\theta}_3 \\
 & \quad + [-gm_2 a_2 - gm_3 l_2] \theta_2 - G_1 u_1 + G_2 u_2 = 0
 \end{aligned} \tag{3.14}$$

$$\begin{aligned}
 & [I_3 + m_3 a_3^2 + I_{p2}] \ddot{\theta}_3 + [m_3 l_2 a_3 - I_{p2}] \ddot{\theta}_2 + [m_3 l_1 a_3] \ddot{\theta}_1 \\
 & + [-C_{p2} - C_3] \dot{\theta}_2 + [C_3 + C_{p2}] \dot{\theta}_3 + [-m_3 a_3 g] \theta_3 - G_2 u_2 = 0
 \end{aligned} \tag{3.15}$$

The linearized continuous time model is represented as follows:

$$\tilde{M} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix} + \tilde{N} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix} + \tilde{P} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} + \tilde{G} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Where

$$\tilde{M} = \begin{bmatrix} J_1 + I_{p1} & l_1 M_2 - I_{p1} & l_1 M_3 \\ l_1 M_2 + I_{p1} & J_2 + I_{p1} + I_{p2} & l_2 M_3 - I_{p2} \\ l_1 M_3 & l_2 M_3 - I_{p2} & J_3 - I_{p2} \end{bmatrix}$$

$$\tilde{N} = \begin{bmatrix} C_1 + C_2 + C_{p1} & -C_2 - C_{p1} & 0 \\ -C_2 - C_{p1} & C_2 + C_3 + C_{p1} + C_{p2} & -C_3 - C_{p2} \\ 0 & -C_3 - C_{p2} & C_3 - C_{p2} \end{bmatrix}$$

$$\tilde{P} = \begin{bmatrix} -M_1 g & 0 & 0 \\ 0 & -M_2 g & 0 \\ 0 & 0 & -M_3 g \end{bmatrix}$$

$$\tilde{G} = \begin{bmatrix} G_1 & 0 \\ -r_1 & f_2 \\ 0 & -G_2 \end{bmatrix}$$

Were

$$M_1 = m_1 a_1 + m_2 l_1 + m_3 l_1$$

$$M_2 = m_2 a_2 + m_3 l_2$$

$$M_3 = a_3 m_3$$

$$J_1 = I_1 + m_1 a_1^2 + (m_2 + m_3) l_1^2$$

$$J_2 = I_2 + m_2 a_2^2 + m_3 l_2^2$$

$$J_3 = I_3 + m_3 a_3^2$$

The linearized continuous model can then be rewritten using the relative angle q_i . Potentiometers are used to measure these angles. The relationship between the relative angle q_i and the angle θ_i is described below:

$$W = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix}, \theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} \text{ and } q = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix}$$

Then

$$q = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} \theta_1 \\ \theta_2 - \theta_1 \\ \theta_3 - \theta_2 \end{bmatrix} = w\theta$$

Each θ in (3.16) is changed to $W^{-1}q$ in the following step. Hence, we may rewrite this equation as follows:

$$MW^{-1} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} + NW^{-1} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} + PW^{-1} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} + G \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (3.17)$$

And then:

$$\begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} = -WM^{-1}NW^{-1} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} - WM^{-1}PW^{-1} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} - WM^{-1}G \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad (3.18)$$

The state space representation that is derived based on the relative angle is defined as follows using (3.18):

$$\begin{aligned} \dot{x} &= Ax + Bu = \begin{bmatrix} 0_3 & I_3 \\ -WM^{-1}PW^{-1} & -WM^{-1}NW^{-1} \end{bmatrix} x + \begin{bmatrix} 0_{3 \times 2} \\ -WM^{-1}G \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \\ y &= Cx = [I_4 \quad 0_{4 \times 2}]x \end{aligned} \quad (3.19)$$

Were

$$I_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, 0_{4 \times 2} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, 0_{3 \times 2} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

Here, the output vector is $y = q$.

The numerical model of the Robogymnast is calculated by substituting the values of the parameters described in Tables 3.1 and 3.2 into (3.19).

Table 3-1: Values of parameters for Robogymnast [5]

Link 1	Link 2	Link 3	Units
$a_1 = 0.0426$	$a_2 = 0.138$	$a_3 = 0.065$	m
$l_1 = 0.155$	$l_2 = 0.180$	$l_3 = 0.242$	m
$C_1 = 0.0172$	$C_2 = 0.0272$	$C_3 = 0.035$	$Nm\ s$
$m_1 = 2.625$	$m_2 = 0.933$	$m_3 = 0.372$	Kg
$I_1 = 0.014$	$I_2 = 0.018$	$I_3 = 0.002$	Kgm^2

Table 3-2: Values of parameters for Motor [5]

Motor1	Motor2	Units
$C_{p1} = 7.73$	$C_{p2} = 7.73$	Nms
$I_{p1} = 0.0358$	$I_{p2} = 0.0358$	Kgm^2
$k_1 = 246:1$	$k_2 = 110.6:1$	Unit less
$G_1 = 1.333$	$G_2 = 0.625$	Nm/V

Were

$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 36.42 & 0.35 & -0.21 & -0.2 & 88.38 & 9.17 \\ -13.10 & 22.06 & 2.23 & 0.20 & -168.29 & 7.70 \\ -2.14 & 1.50 & 5.68 & 0.02 & 7.69 & -201.45 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ -15.19 & -0.74 \\ 28.92 & -0.62 \\ -1.32 & 16.21 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

The continuous time model's upright eigenvalues are depicted in (3.19). as follows:

$$[-166.8506 \quad -203.1990 \quad -5.4598 \quad 5.3762 \quad 0.1662 \quad 0.0270]$$

The system looks unstable based on the eigenvalues above because three positive characteristic roots exist.

The system controllability matrix is:

$$CO = [B \quad AB \quad A^2B \quad A^3B \quad A^4B \quad A^5B]$$

The CO matrix's rank, as determined by the MATLAB command $\text{rank}(CO)$, is 6, meaning that all six states are achievable with the correct input provided to the system via $u(t)$ [50]. Therefore, total control over the system is possible.

The system's observability matrix is:

$$OB = [C \quad CA \quad CA^2 \quad CA^3 \quad CA^4 \quad CA^5]^T$$

With the MATLAB command $\text{rank}(OB)$, the OB matrix's rank is 6, meaning that each of the six states can be seen via linear combinations of the output

variables $y(t)$ [6]. Thus, the entire system can be observed. Based on the preceding analysis, it can be inferred that the system's linearized model, as shown in equation (3.19), is both observable and controllable. Consequently, the system can be controlled by applying the controller.

By discretized (3.19) using MATLAB® software with a sampling period of 25 milliseconds (obtained practically in [47]) to produce the discrete-time model of the Robogymnast:

$$\begin{aligned} x(k+1) &= A_d x(k) + B_d u(k) \\ y(k) &= C_d x(k) \end{aligned} \quad (3.20)$$

were

$$A_d = \begin{bmatrix} 1.0100 & 0.0024 & 0.0002 & 0.0250 & 0.0101 & 0.0012 \\ -0.0015 & 1.0025 & 0.0003 & 0.0000 & 0.0059 & 0.0002 \\ -0.0003 & 0.0002 & 1.0006 & 0.0000 & 0.0002 & 0.0049 \\ 0.7761 & 0.2334 & 0.0240 & 1.0069 & 0.5232 & 0.0646 \\ -0.0771 & 0.1298 & 0.0143 & -0.0003 & 0.0158 & 0.0021 \\ -0.0134 & 0.0122 & 0.0285 & -0.0001 & 0.0020 & 0.0068 \end{bmatrix}$$

$$B_d = \begin{bmatrix} -0.0017 & -0.0001 \\ 0.0033 & -0.0000 \\ -0.0000 & 0.0016 \\ -0.0895 & -0.0052 \\ 0.1696 & -0.0001 \\ -0.0003 & 0.0800 \end{bmatrix}, C_d = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

Three unstable eigenvalues (outside the unit circle) for the discrete-time model shown in (3.20) are listed below:

[0.0156 0.0062 1.1439 0.8724 1.0042 1.0007]

3.4 Mathematical Model in the Upward Position Derived by Elman Neural Network (ENN)

Robogymnast is regarded as a TIP in an unstable position for balancing model functions. The Euler-Lagrange formula was used to derive mathematical modeling. The Euler-Lagrange method is the most common technique for obtaining dynamical equations of several rigid systems [47]. Mathematical modeling presents an estimation of real-world systems. However, as the system's complexity rises, a mathematical model loses accuracy because modeling is a procedure of simplifying and deducting, and information about a system is lost by simplification. The gymnastic robot is a nonlinear multi-link under-actuated mechanism that demands a complicated mathematical model with the accuracy of information taken into account. Neural networks are commonly used in nonlinear system modeling applications. Ismail et al. existing model was enhanced using an artificial neural network, as shown in Fig.3.4.

The following equation can be used to represent the Elman neural network (ENN) model:

$$X(k) = A_d X(k - 1) + B_d U(k - 1) \quad (3.21)$$

$$Y(k) = C_d X(k) \quad (3.22)$$

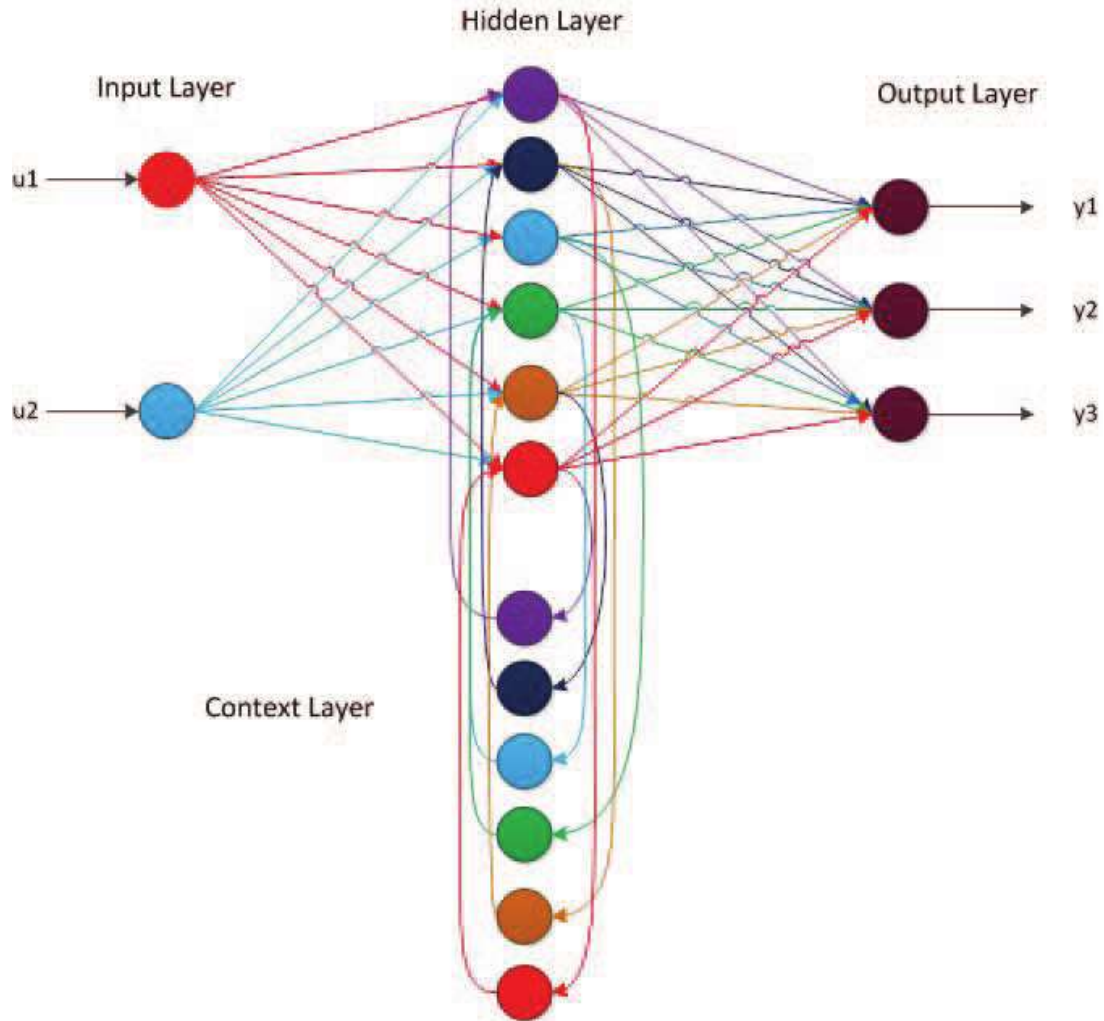


Fig. 3.3 Elman Neural Network model of the Robogymnast

X represents the state, while Y represents the output, precisely the relative angular positions (q_1, q_2, q_3) of links 1, 2, and 3 from the balancing point. Additionally, we have U, which represents the input for the model. U consists of two components, namely the input voltages u_1 and u_2 . These voltages, limited to a range of $|u_1, u_2| \leq 10V$, are the control effort input for the DC motors at joints 2 and 3.

$$A_d = \begin{bmatrix} 0.99 & -2.43e^{-3} & -2.34e^{-4} & 2.49e^{-2} & 1.01e^{-2} & 1.20e^{-3} \\ 1.49e^{-3} & 0.99 & -2.72e^{-4} & 3.78e^{-5} & 5.88e^{-3} & 2.15e^{-4} \\ 2.55e^{-4} & -2.22e^{-4} & 1.00 & 5.30e^{-6} & 2.15e^{-4} & 4.95e^{-3} \\ 0.77 & 0.23 & 2.39e^{-2} & 0.99 & 0.52 & 6.37e^{-2} \\ -7.59e^{-2} & 0.13 & 1.43e^{-2} & 2.64e^{-3} & 1.55e^{-2} & 2.01e^{-3} \\ -1.32e^{-2} & 1.21e^{-2} & 2.85e^{-2} & 3.97e^{-4} & 2.05e^{-3} & 6.55e^{-3} \end{bmatrix}$$

$$B_d = \begin{bmatrix} -2.91e^{-3} & -1.62e^{-4} \\ -5.51e^{-3} & -2.94e^{-5} \\ -6.26e^{-5} & 2.71e^{-3} \\ -0.15 & -8.61e^{-3} \\ 0.283 & -3.08e^{-4} \\ -6.55e^{-4} & 0.13 \end{bmatrix}, C_d = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

The results obtained in research [5] were applied to the system practically; when input and output data were available for several cases, the researcher [6] was able to extract the state space of the system; this is the simple concept of a Neural Network (for the different types of networks used).

Chapter four applies the controller to the stat space derived from the Euler-Lagrange and Elman Neural network and compares the results.

3.5 Summary

A description of the system is discussed in this chapter. The mathematical model of the system's derivation is illustrated in the upright position. This chapter demonstrated the step-by-step derivation of the system's state space model from the Euler-Lagrange equations and the Elman neural network modeling. The calculated state space is used in the simulation of the system

to understand the behavior before being implemented in real; this is discussed in Chapter Four, where different control methods will be applied to the system.

Chapter Four: **Controller Design with Different Control Strategy**

4.1 Introduction

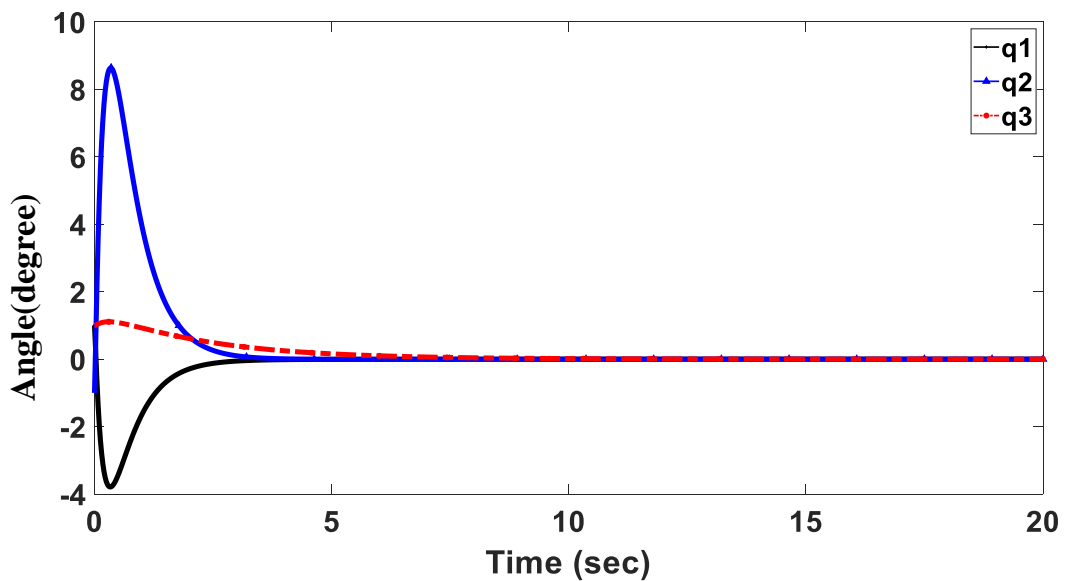
This chapter discussed the design of different controllers to control both the states and control effort voltage, including three angles and two control input voltages. Several proposed controllers, including DLQR, WOA-based DLQR, AO-based DLQR, FLC, and FLC with self-tuning-gain via WOA, all intended to achieve system stability.

4.2 Results of DLQR Applying a Mathematical Model Derived from the Euler-Lagrange

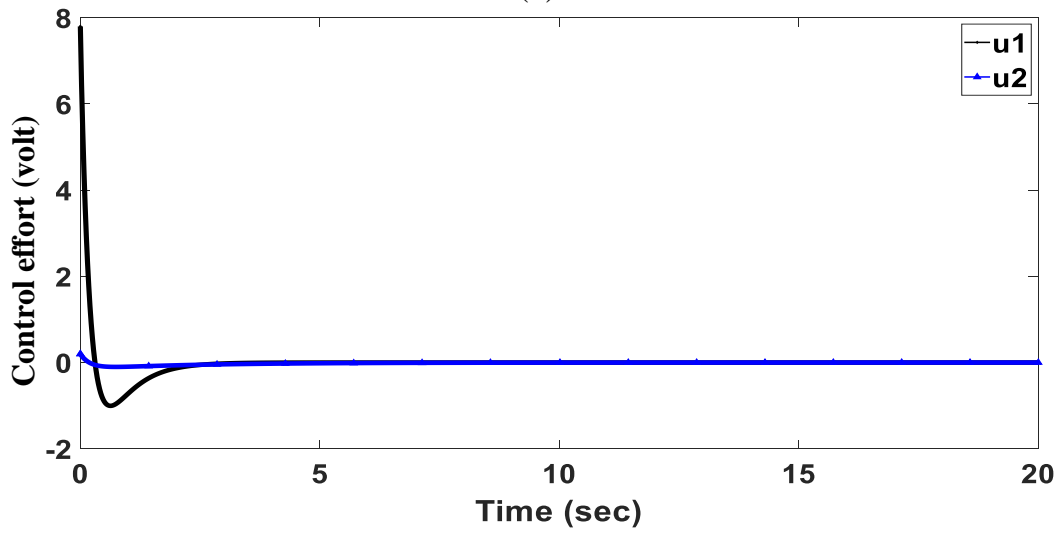
The construction of DLQR parameters is based on weight matrices, and the most challenging part of designing a DLQR controller is the adjustment of the weight matrices parameters, which is often adjusted by repeating trial and error method. The matrices Q and R are configured as diagonal matrices. Q and R matrixes obtained by the trial-error method (see Appendix A) as below:

$$Q = \begin{bmatrix} 50 & 0 & 0 & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 & 0 & 0 \\ 0 & 0 & 70 & 0 & 0 & 0 \\ 0 & 0 & 0 & 60 & 0 & 0 \\ 0 & 0 & 0 & 0 & 20 & 0 \\ 0 & 0 & 0 & 0 & 0 & 30 \end{bmatrix}, R = \begin{bmatrix} 0.1 & 0 \\ 0 & 2 \end{bmatrix}$$

The controller is applied when the initial deflection of angles equal to $q_1=1^\circ$, $q_2=-1^\circ$, and $q_3=1^\circ$. These angles represent the estimated maximum deflection that the Robogymnast can achieve before the system loses the ability to restore it to a balanced upright configuration [5]. Fig. 4.1.(a) Demonstrates that the settling time of three relative angles is 5 seconds. The deviation reaches 8.7 degrees, -3.8 degrees, and 1.2 for the first, second, and third link. Fig. 4.1.(b) shows the control effort consumed by two motors, observed that the value of two motors is satisfactory and under limited voltage.



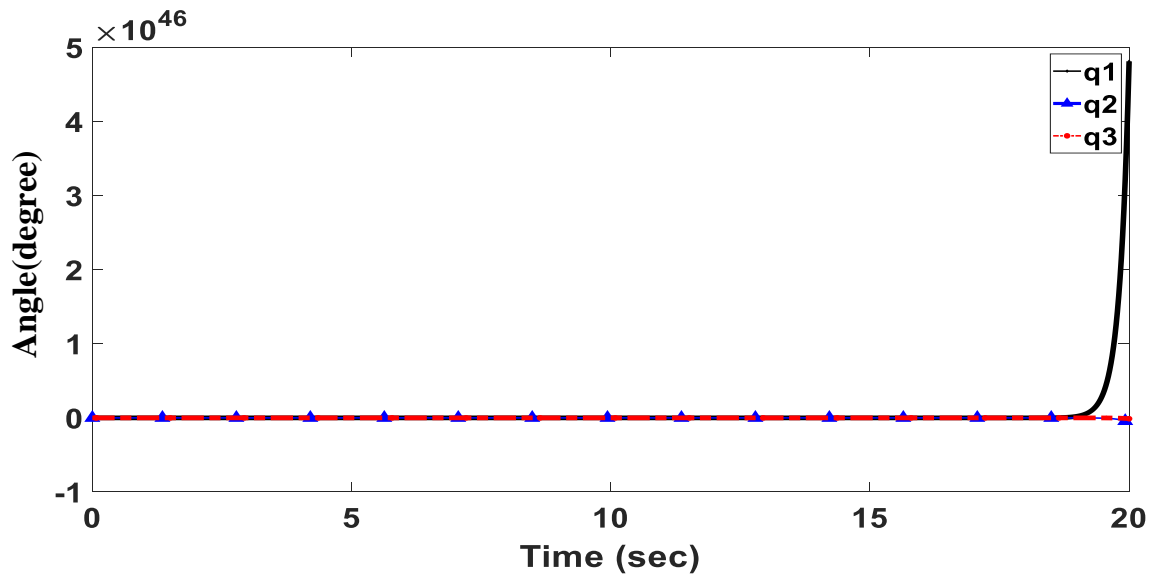
(a)



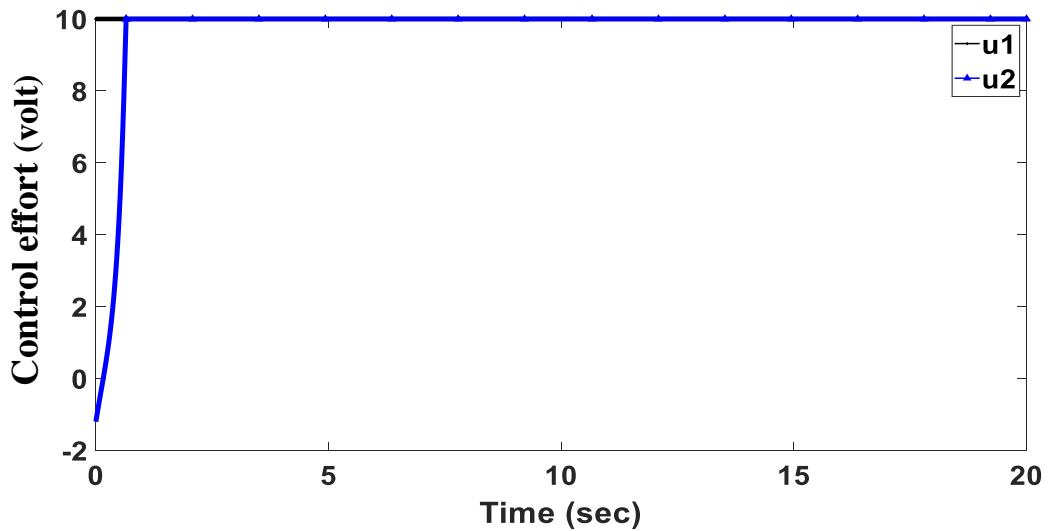
(b)

Fig. 4.1 Time response with initial $q_1=1^\circ$, $q_2=-1^\circ$, $q_3=1^\circ$ of the Euler-Lagrange formula. (a) Relative angle 1, 2, 3 (b) Control effort 1, 2.

The system becomes unstable when the initial is set to 3° , as shown in Fig 4.2.



(a)

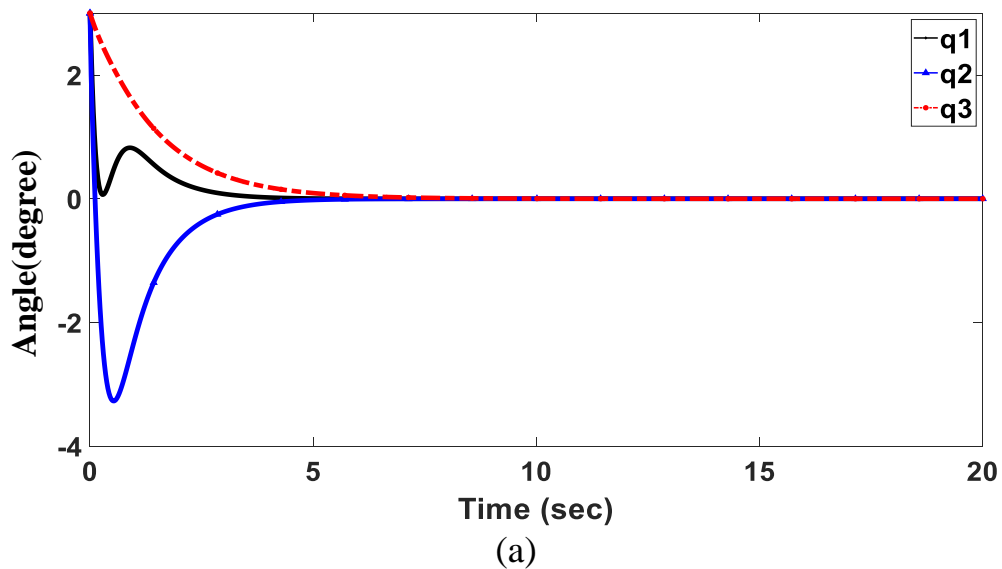


(b)

Fig. 4.2 Time response with initial $q_1=3^\circ$, $q_2=3^\circ$, $q_3=3^\circ$ of the Euler-Lagrange formula. (a) Relative angle 1, 2, 3 (b) Control effort 1, 2.

4.3 Results of DLQR Applying a Mathematical Model Derived from the Elman Neural Network (ENN)

The project's observable results are the state response and the control effort (control input voltage). The adjusting process is employed when the initial deflection of absolute angles equals $q_1=3^\circ$, $q_2=3^\circ$, and $q_3=3^\circ$. These angles represent the estimated maximum deflection that the Robogymnast can achieve before the system loses the ability to restore it to a balanced upright configuration [6]. As shown in Fig. 4.3(a), the first, second, and third links consumed a setting time of 7 seconds to reach the steady state. The first link has a deviation of 0.8 degrees and no deviation from the third link. However, the second link has a deviation of -3.3 degrees. The first motor consumed 3V, and the second motor consumed 0.3V to achieve the desired response, which was less than the limited voltage (12V), as shown in Fig. 4.3 (b).



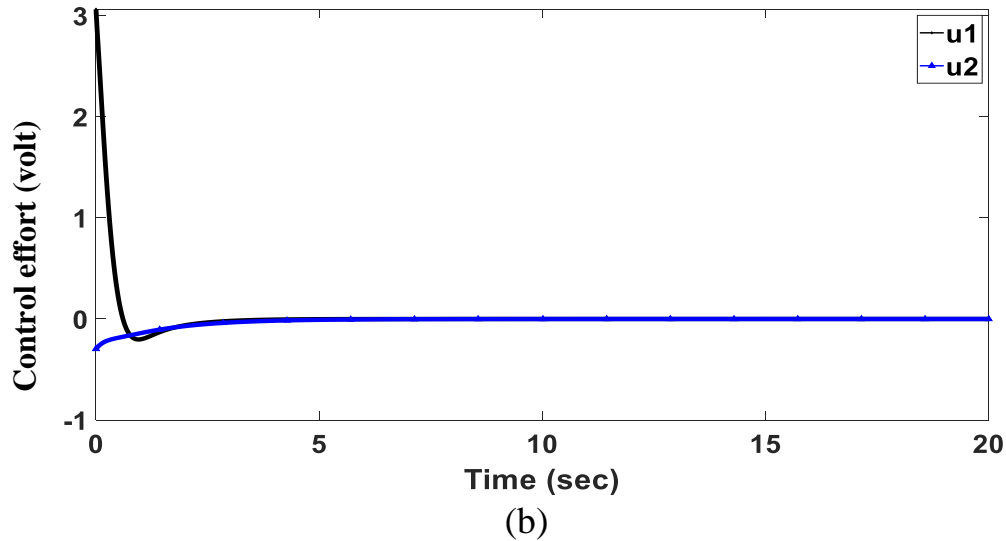


Fig. 4.3 Time response with initial $q_1=3^\circ$, $q_2=3^\circ$, $q_3=3^\circ$ of the ENN model. (a) Relative angle 1, 2, 3 (b) Control effort 1, 2.

The conclusion from Fig. 4.1, Fig. 4.2, and Fig. 4.3 is that the state space specific by the ENN model is much better than the state space derived by the Euler-Lagrange formula because the ENN model accepted initial deflection of absolute angles reach 3 degrees where the state space that derived by Euler-Lagrange formula accepted initial deflection one degree also ENN model is considered closer to the actual practical application.

The Robogymnast system's dynamic model shows that the system is in sixth order and contains two inputs. So, the size of the matrix Q is 6×6 , and matrix R is 2×2 . The cost function clarifies that Q and R define the priority level assigned to each state and input. That is, a big valued Q matrix (with an accepted range specific to the designer's experience) and a small valued R matrix indicates that the changes in the state matrix will be amplified compared to the changes in the input matrix. This decision results in a controller responding more sensitively to system states than to control input. The rationale for this decision is that stability is the primary design criterion,

so stability is determined by the system states. Since the input values are given a small weight, this could result in actuator saturation. Extensive controller adjusting is therefore required to increase the controller's stability range. Although the system is stabilized (using trial and error), several reasons exist to increase the controller's capabilities. The link's displacement may go over the limitations of stability range in actual application. The second reason is that noise has yet to be incorporated into the models during operation, and the system will become more unstable by adding noise. The tests were carried out by individually varying the Q matrix while keeping the other Q and R parameters constant at 1, which makes it possible to calculate the individual effects of every value in the Q matrix on the maximum angular displacement. Then, to modify control efforts, vary R and set Q at 1 (see Appendix A). This process consumed Time, inaccurately, and effort [51].

Based on the previously mentioned reasons that have been proven through experience and to solve them to get a good response, Optimization will be used (this discussion is in the next section). The Optimization technique is responsible for finding a suitable value for Q and R (see Fig. 4.4).

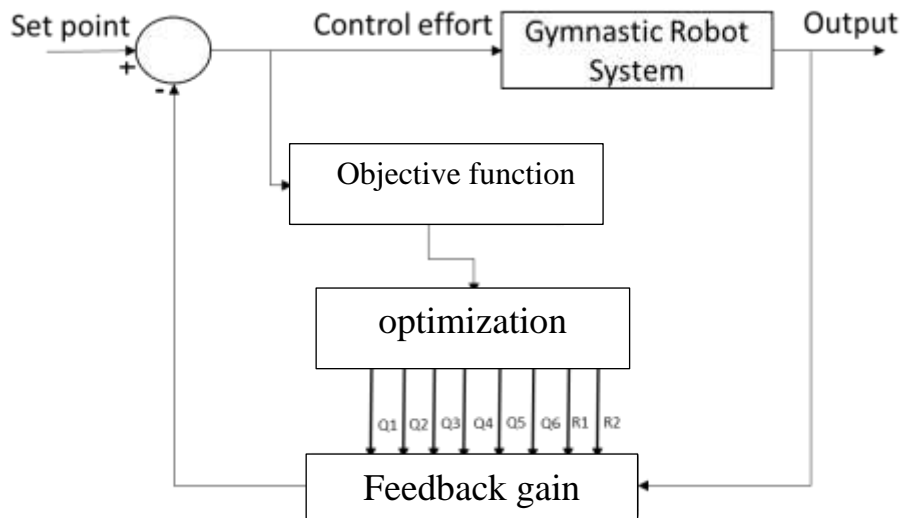


Fig. 4.4 Block diagram for optimization-based DLQR controller.

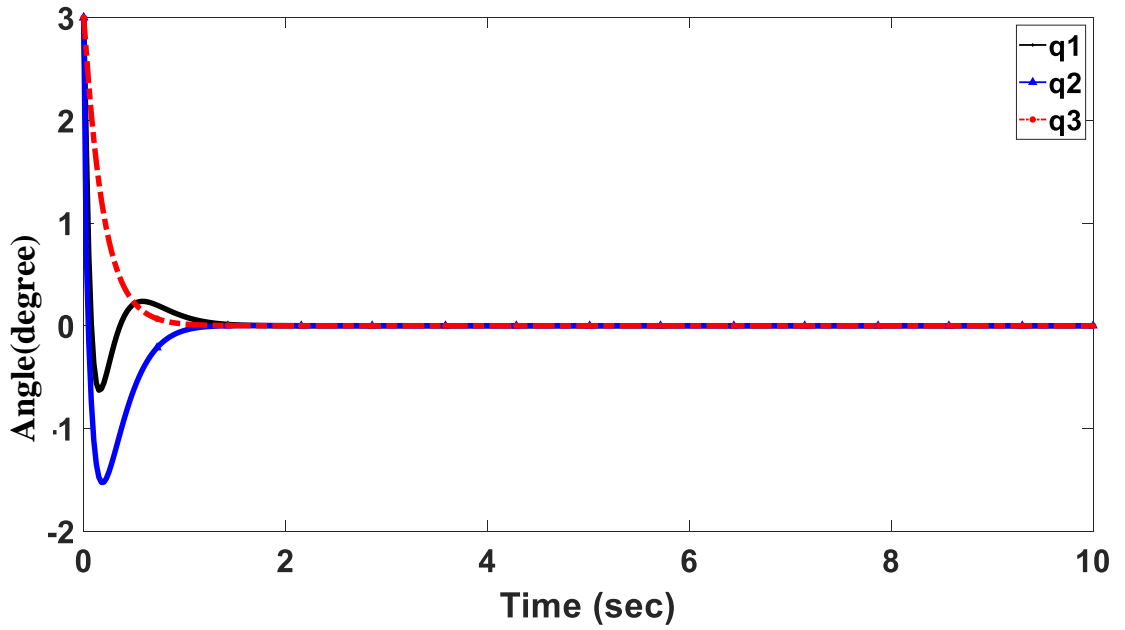
4.4 Results of Whale Optimization Algorithm

The WOA algorithm is utilized to discover the best possible solution globally for the Linear Quadratic Regulator (LQR) controller. The objective is to minimize both the time it takes for the Robogymnast to transition from an unbalanced inverted state to a balanced upright state and the voltage required for this transition. The matrices Q and R are configured as diagonal matrices. Q and R matrixes obtained by the WOA algorithm and the adjusted value after optimization as below:

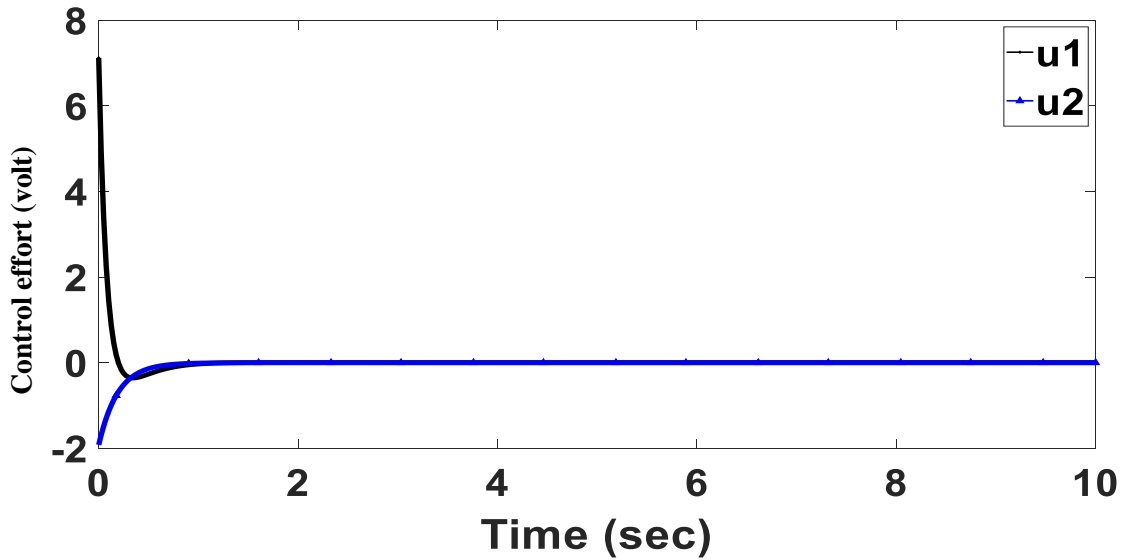
$$Q = \begin{bmatrix} 362.02735 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1000 & 0 & 0 & 0 & 0 \\ 0 & 0 & 7605.4042 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 54.048495 \end{bmatrix}$$

$$R = \begin{bmatrix} 0.1 & 0 \\ 0 & 3.9882462 \end{bmatrix}$$

The optimization is employed when the initial deflection equals $q_1=3^\circ$, $q_2=3^\circ$, and $q_3=3^\circ$. As shown in Fig. 4.5 (a), the first, second, and third links reached steady after 1.825 seconds with minimum deviation. Moreover, as demonstrated, the control voltage of the first motor consumed 7.12V, and the second motor consumed 2V to achieve the desired response, and this is less than the limited voltage (12V), as shown in Fig. 4.5 (b).



(a)



(b)

Fig. 4.5. Time response after WOA with initial $q_1=3^\circ$, $q_2=3^\circ$, $q_3=3^\circ$.

(a) Relative angle 1, 2, 3 (b) Control effort 1, 2.

The convergence curve in Fig. 4.6 is getting closer to 1.825 sec time to reach 0.001 degree steady-state error of all relative angles after 13 iterations, which indicates the speed WOA of reaching a steady state.

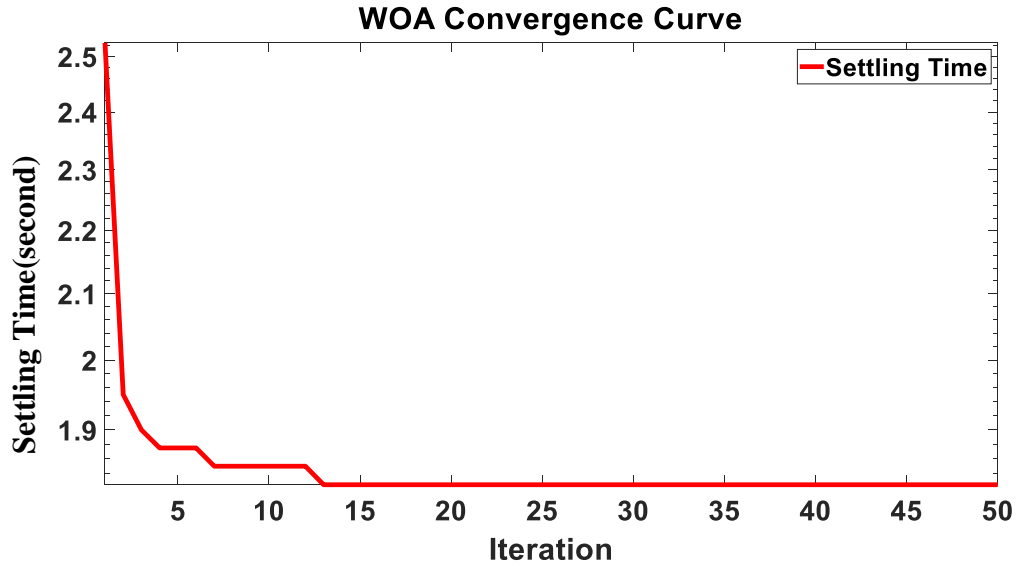


Fig. 4.6 Convergence curve of WOA

4.5 Results of Aquila Optimizer

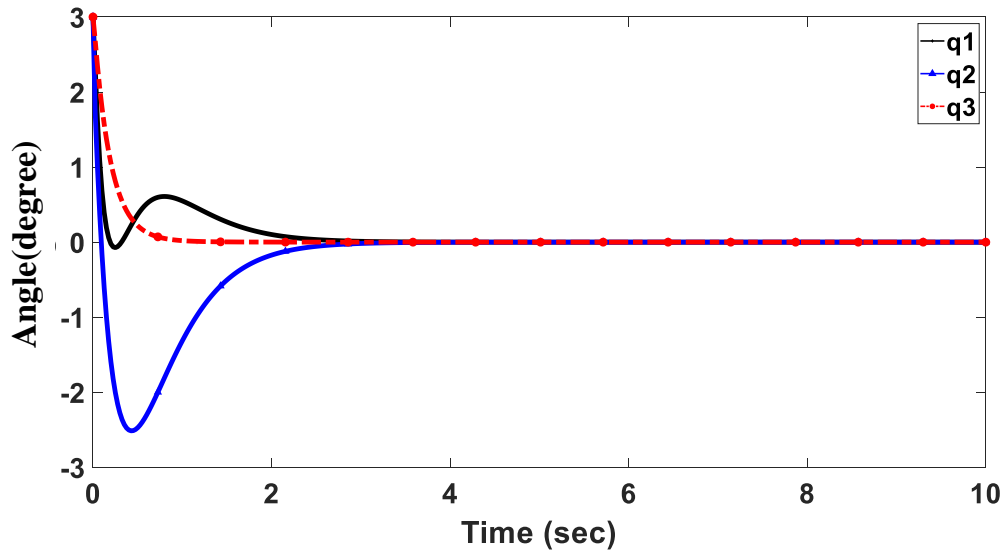
The AO algorithm is utilized to discover the best solution globally for the DLQR controller. Q and R matrixes obtained by the AO algorithm and the adjusted value after optimization as below:

$$Q = \begin{bmatrix} 929.32223 & 0 & 0 & 0 & 0 & 0 \\ 0 & 397.93419 & 0 & 0 & 0 & 0 \\ 0 & 0 & 5715.4537 & 0 & 0 & 0 \\ 0 & 0 & 0 & 13.736104 & 0 & 0 \\ 0 & 0 & 0 & 0 & 35.812884 & 0 \\ 0 & 0 & 0 & 0 & 0 & 9.6405528 \end{bmatrix}$$

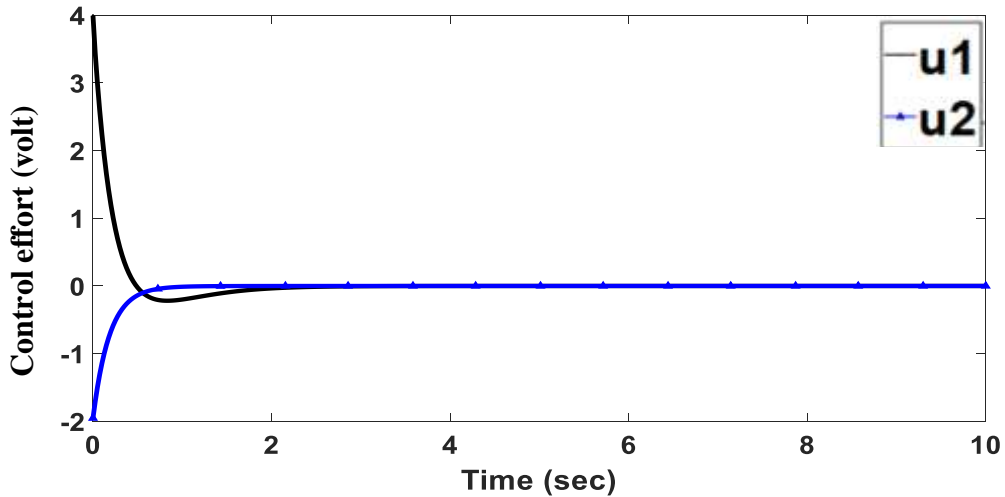
$$R = \begin{bmatrix} 0.76390393 & 0 \\ 0 & 3.3013285 \end{bmatrix}$$

The first, second, and third links reached a steady state after 3 seconds, with the maximum deviation in the second link reaching -2.75 degrees, as shown

in Fig. 4.7 (a). Moreover, the first motor consumed 4V, and the second consumed -2V to achieve the desired response, as shown in Fig. 4.7 (b).



(a)



(b)

Fig. 4.7 Time response after Aquila Optimization with initial $q_1=3^\circ$, $q_2=3^\circ$, $q_3=3^\circ$. (a) Relative angle 1, 2, 3 (b) Control effort 1, 2.

Figure 4.8 illustrates the fuzzy logic control simulation system; there are three angles and two input voltages. So, there are three fuzzy logic controls for each

input. Each fuzzy has two input errors in angle and a change of error angle. The output gain of fuzzy optioned using trial and error.

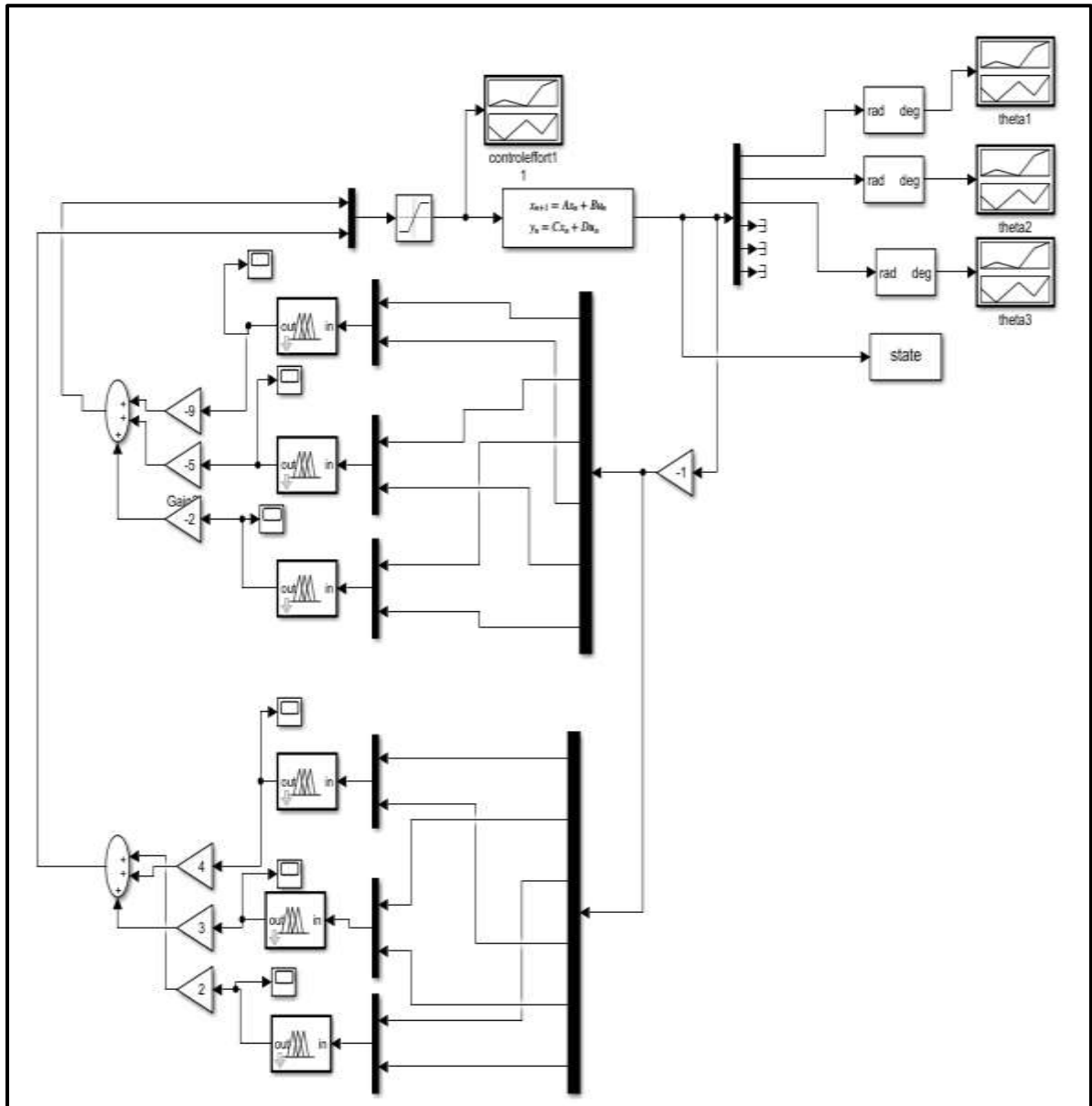
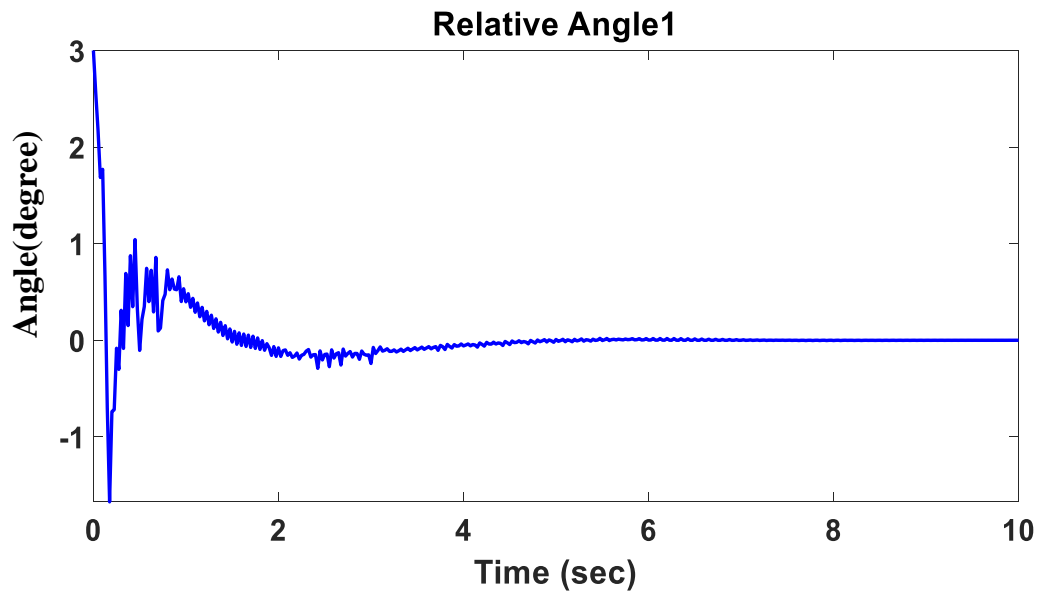


Fig. 4.8 Simulation System of Fuzzy Logic

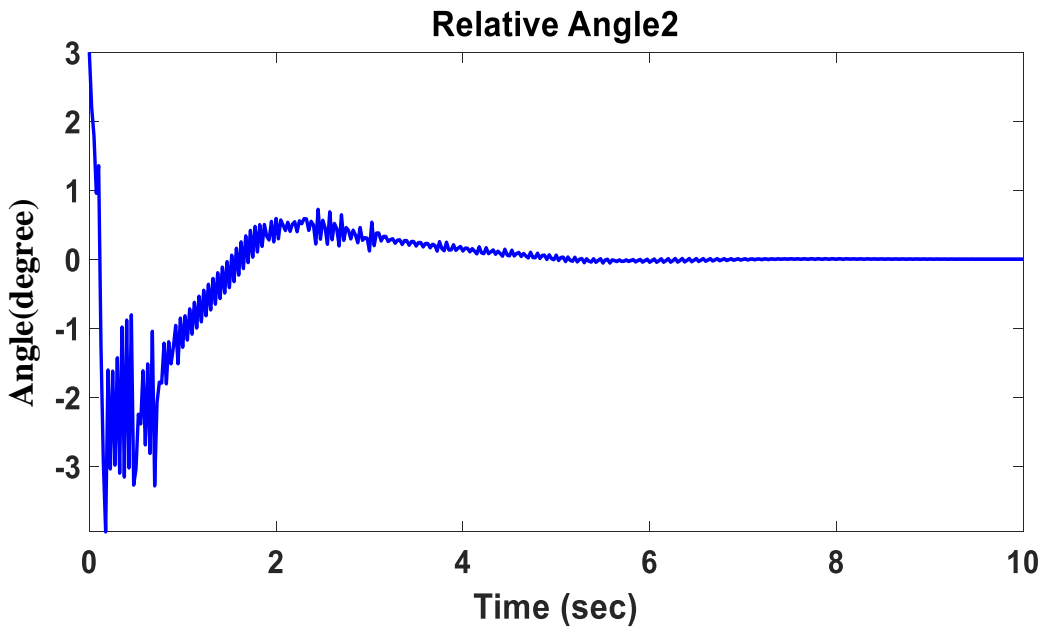
4.6 Results of Fuzzy Logic Control

Fig. 4.9 presents the results of the transient and steady-state responses of the relative angular position; the first and third link's relative angular positions deviation about -1.8° and -0.5° , respectively, while the second link's relative

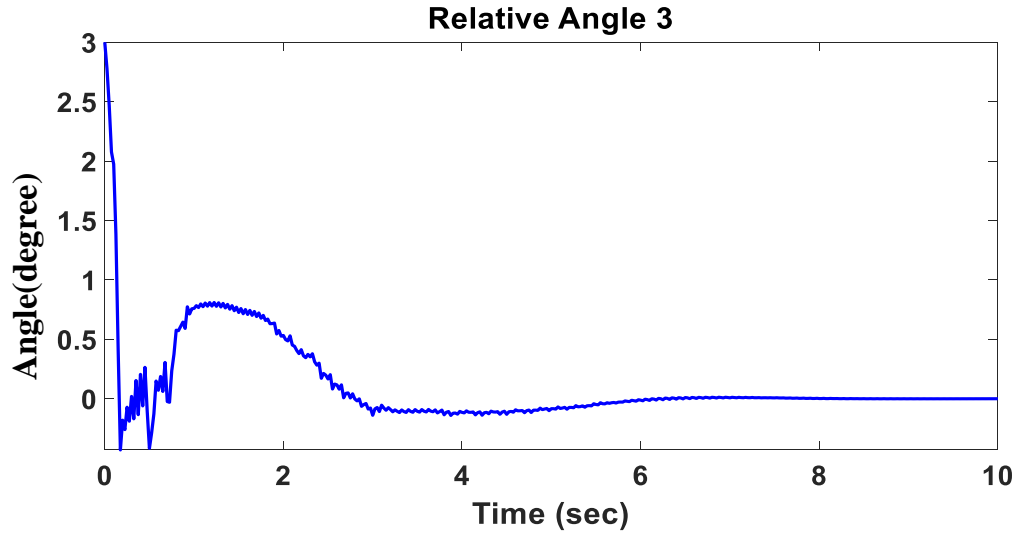
angle deviation reaches approximately -4° . As observed, the three links required 6 seconds to reach the steady state. The control effort of the first motor reached 8 volts, and the second motor control effort consumed -4 volts; these voltages were within the limited voltages. As a result, the gymnastic robot can stabilize in an inverted position with a satisfactory transient response and voltage.



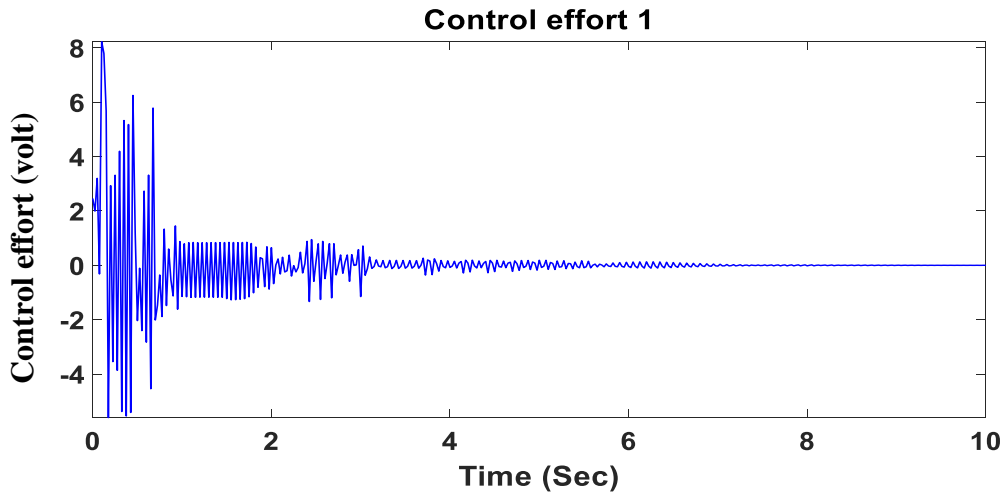
(a)



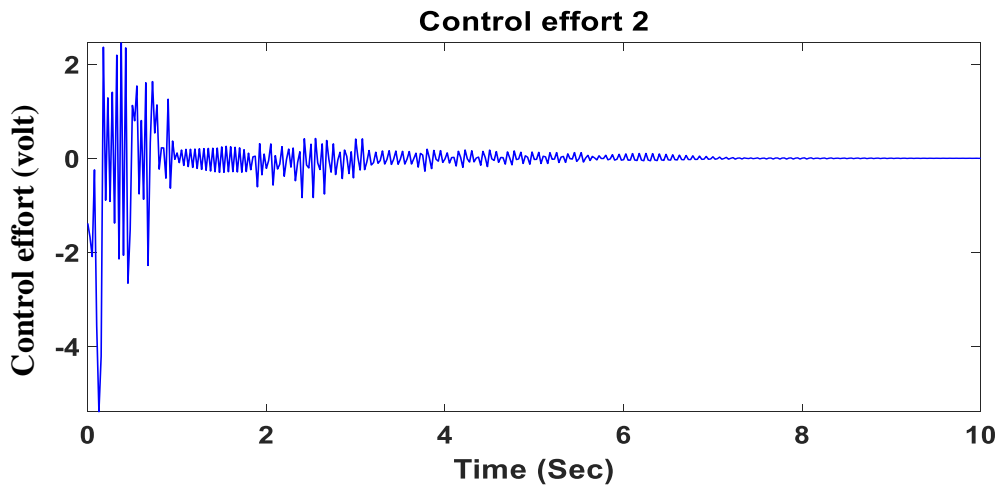
(b)



(c)



(d)



(e)

Fig. 4.9 FLC time response with initial deflection $q_1=3^\circ$, $q_2=3^\circ$, $q_3=3^\circ$. (a) Relative angle 1 (b) Relative angle 2 (c) Relative angle 3 (d) Control effort 1 (e) Control effort 2.

4.7 Hybrid System Fuzzy Logic and Whale Optimization

In this section, a hybrid controller combining two control systems is designed. The objective of the hybrid control system is to enhance system performance by combining the valuable specifications from the many control systems. Naturally, different hybrid controller structures, including fuzzy hybrid with other controllers, have been presented by researchers [52], [53], [54].

WOA-based DLQR and FLC controllers are used in this study to improve outcomes. When designing the WOA-based DLQR, the results were good, and the system could stabilize within a short time. Still, when implementing the system practically, the optimization takes time that is not commensurate with the practical application because it performs calculations in each case. These calculations take time, and the controller becomes off-line tuning. FLC was designed to achieve online tuning to stabilize and balance the system. The result of the FLC was accepted, although the system consumed more settling time to be stable in an inverted position. So, we used a hybrid controller FLC with WOA-based DLQR to achieve online tuning with less settling time.

Fig. 4.10 shows the simulation system of the hybrid controller that combines the WOA-based DLQR parameter with the FLC system specification discussed in the previous sections.

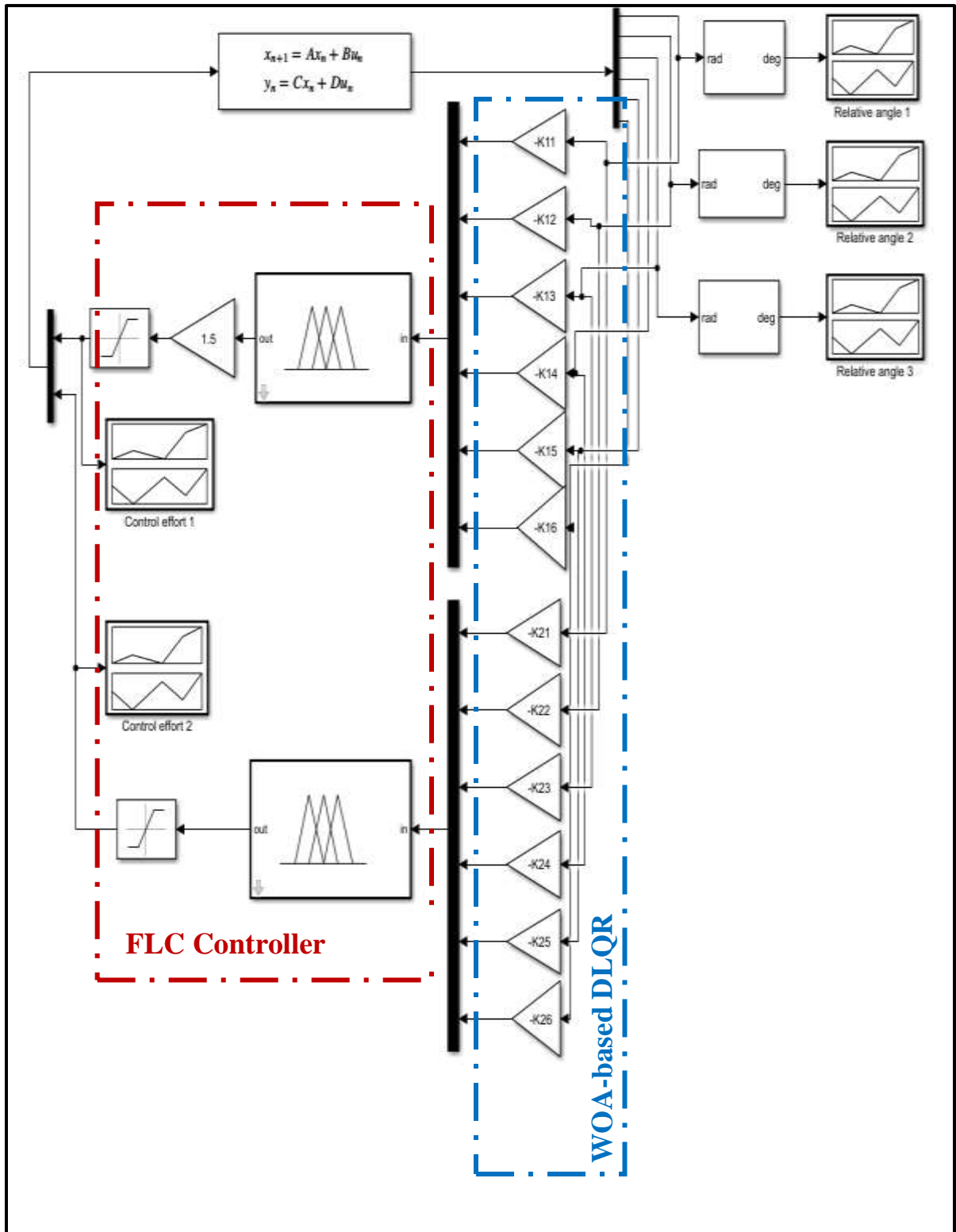
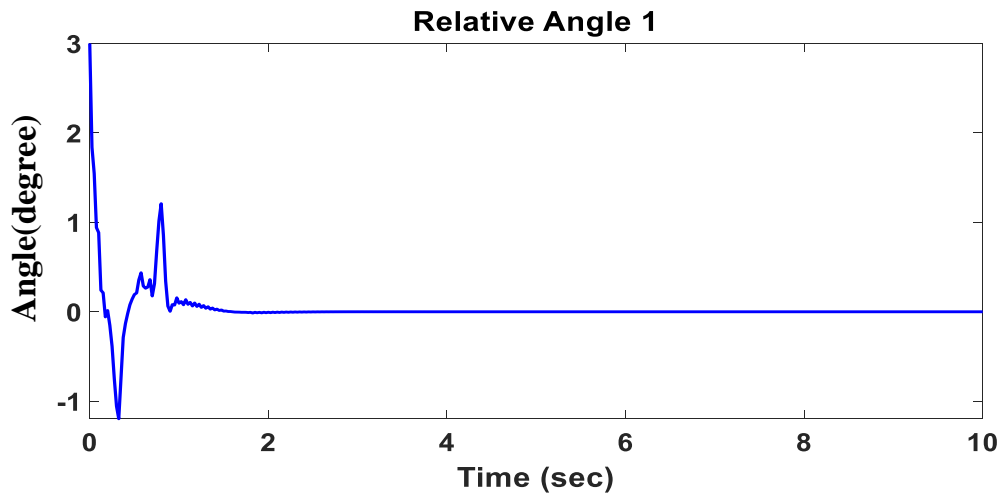


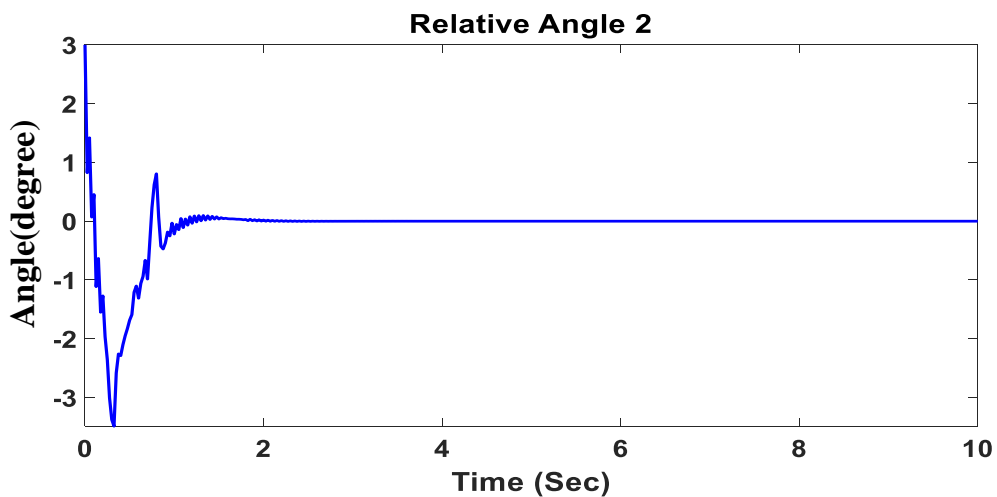
Fig. 4.10. Simulation System of Hybrid Controller.

4.7.1 Results of Hybrid Control

Fig. 4.11 presents the results of the transient and steady-state responses of the relative angular position; the first and second relative angles consumed 2 seconds to reach the steady state. In contrast, the third relative angle consumed one second, a suitable settling time to balance the RoboGymnast in an upright position. The deviation of the first and second links reached -1.15° and -3.4° , respectively, but there was no deviation in the third link from a vertical position. The first motor consumed 6.7 volts of control effort, but the second motor consumed only -1.5 volts; this was considered satisfactory voltage to become the RoboGymnast in an inverted position.



(a)



(b)

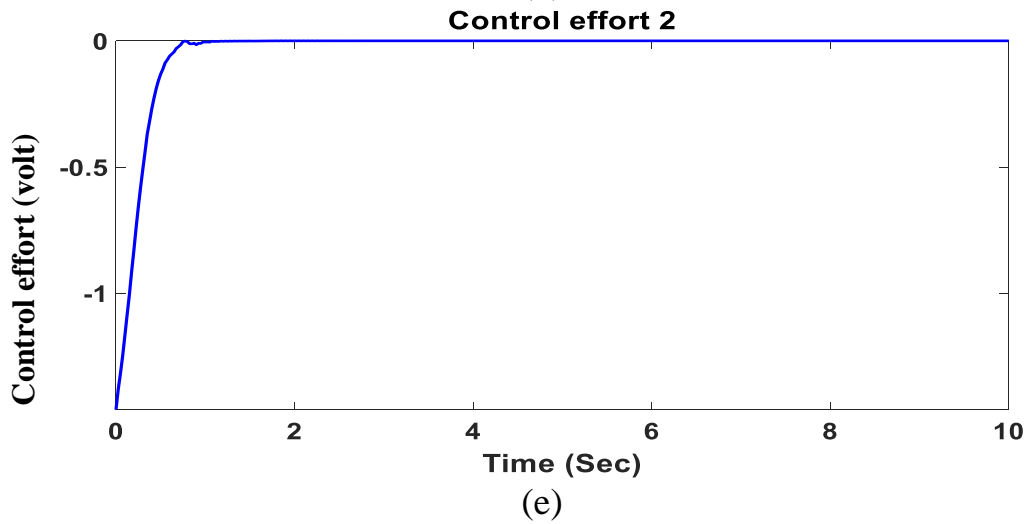
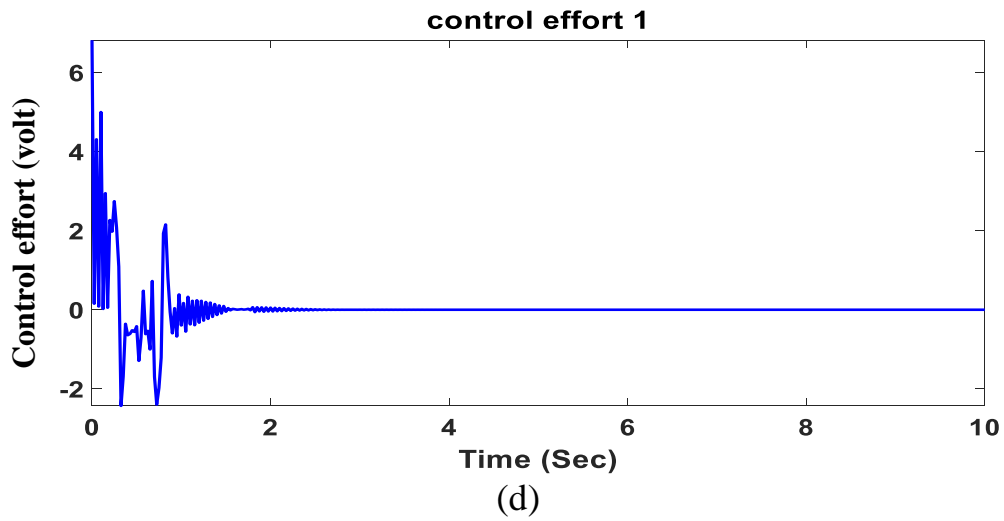
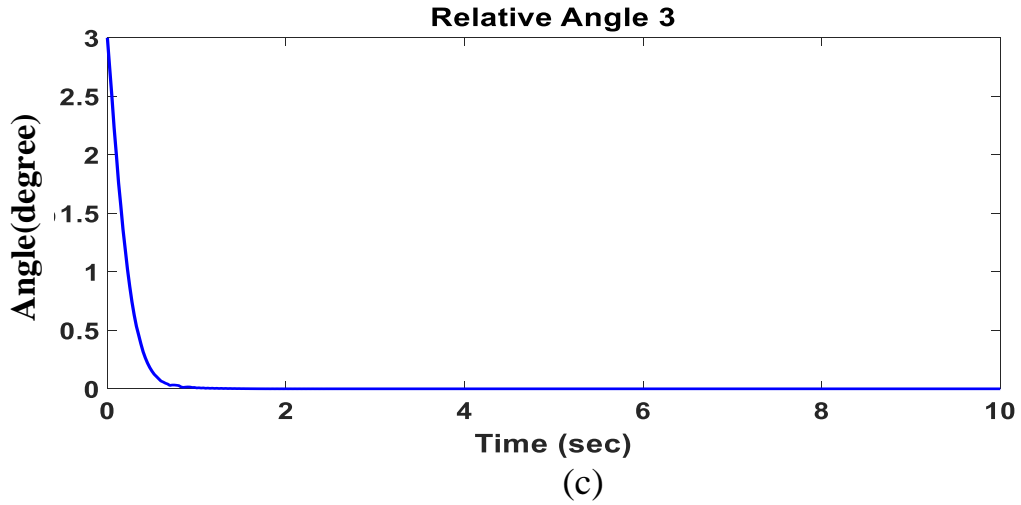
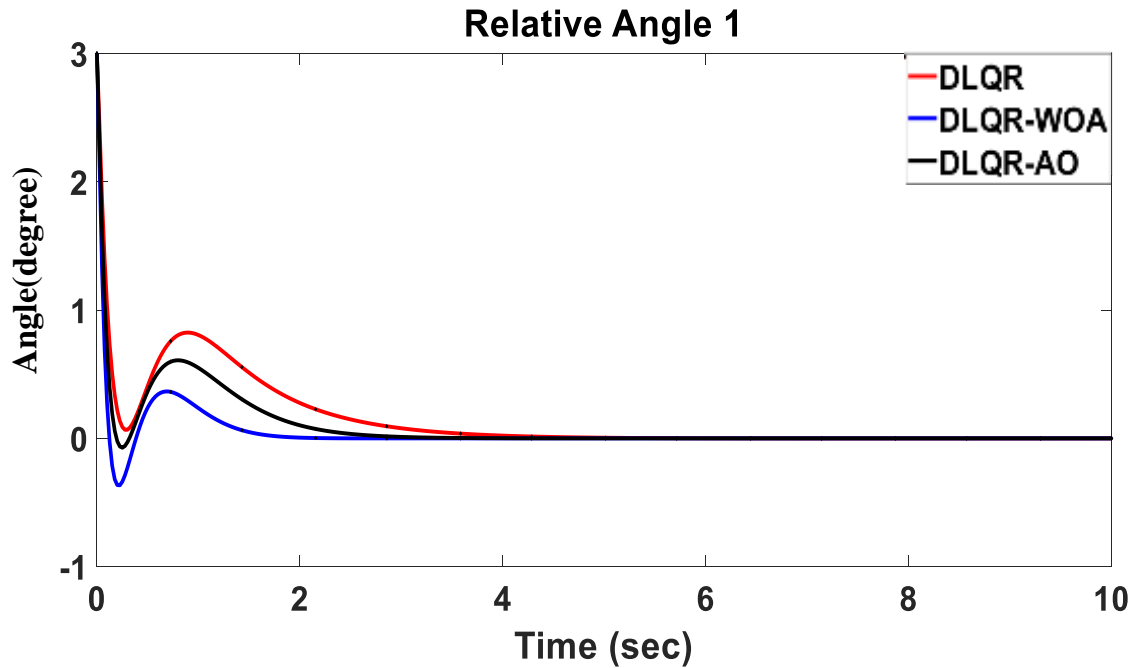


Fig. 4.11 Hybrid control time response with initial deflection $q_1=3^\circ$, $q_2=3^\circ$, $q_3=3^\circ$. (a) Relative angle 1 (b) Relative angle 2 (c) Relative angle 3 (d) Control effort 1 (e) Control effort 2.

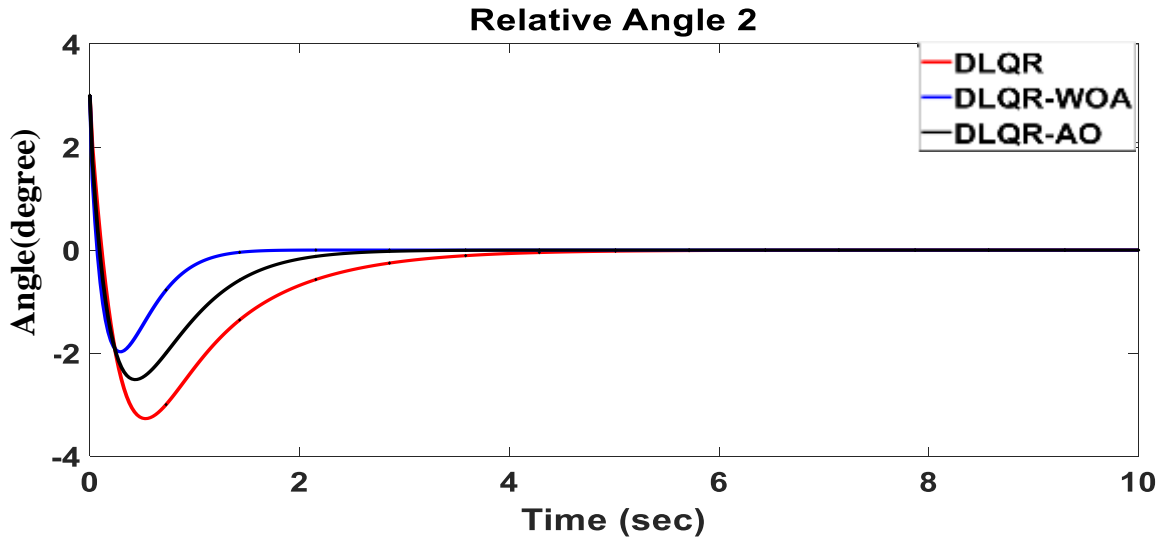
4.8 Comparative results

4.8.1 Comparison Between Before and After Optimization Techniques

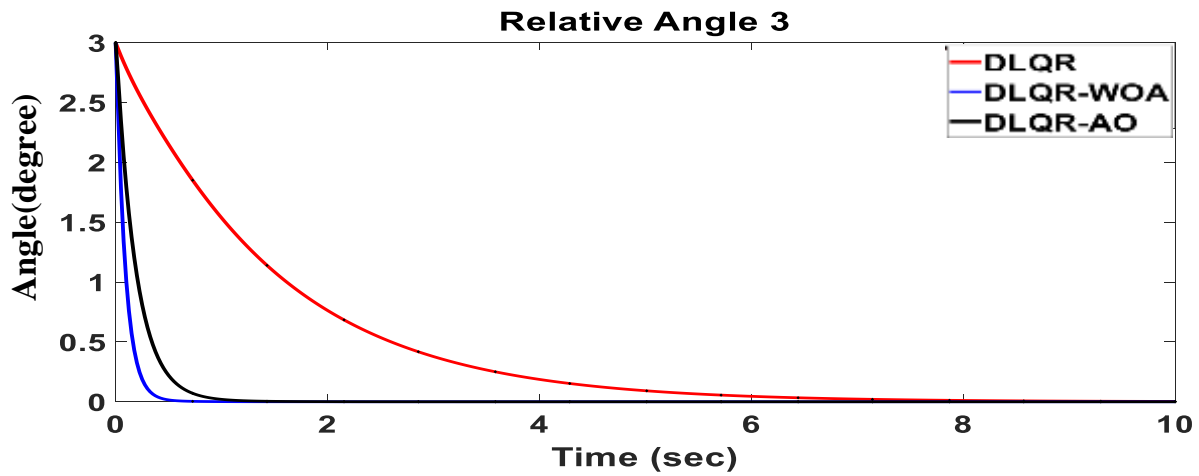
Fig. 4.12 compares the DLQR, WOA-based DLQR, and AO-based DLQR controllers. It is clear that the WOA-based DLQR controller gets the best result according to the transient response of the relative angles, but it consumes the higher voltage of two motors between the other type controllers. However, the WOA-based DLQR controller is considered the best controller because the system's stability has higher priority than the consumed voltage since the voltages are within limits.



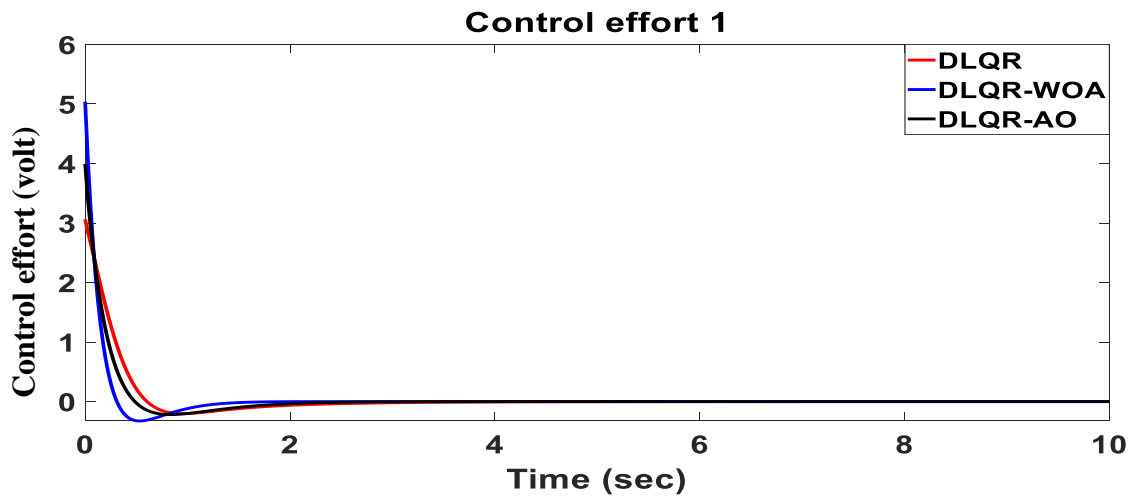
(a)



(b)



(c)



(d)

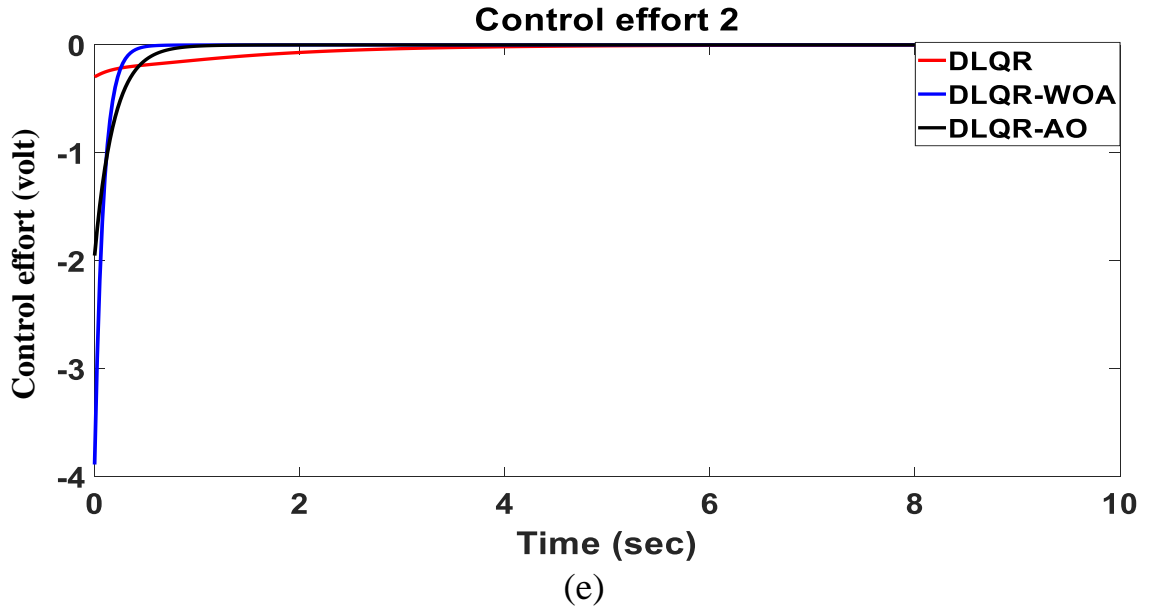
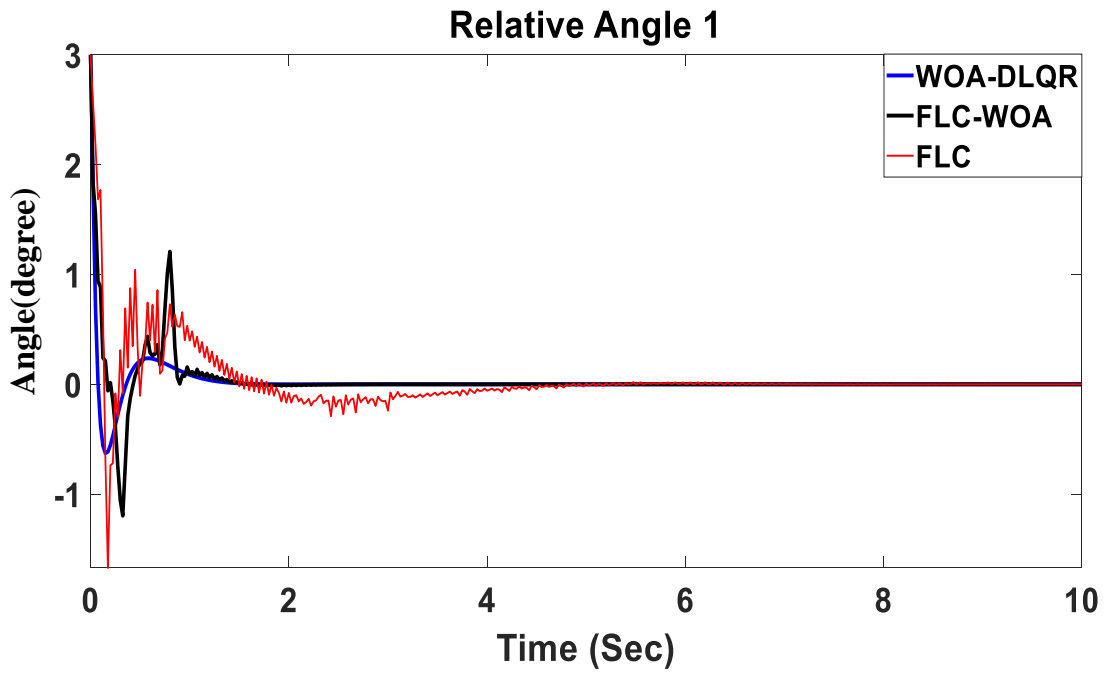


Fig. 4.12 Comparison time response between DLQR, WOA-based DLQR, and AO-based DLQR. (a) Relative angle 1 (b) Relative angle 2 (c) Relative angle 3 (d) Control effort 1 (e) Control effort 2.

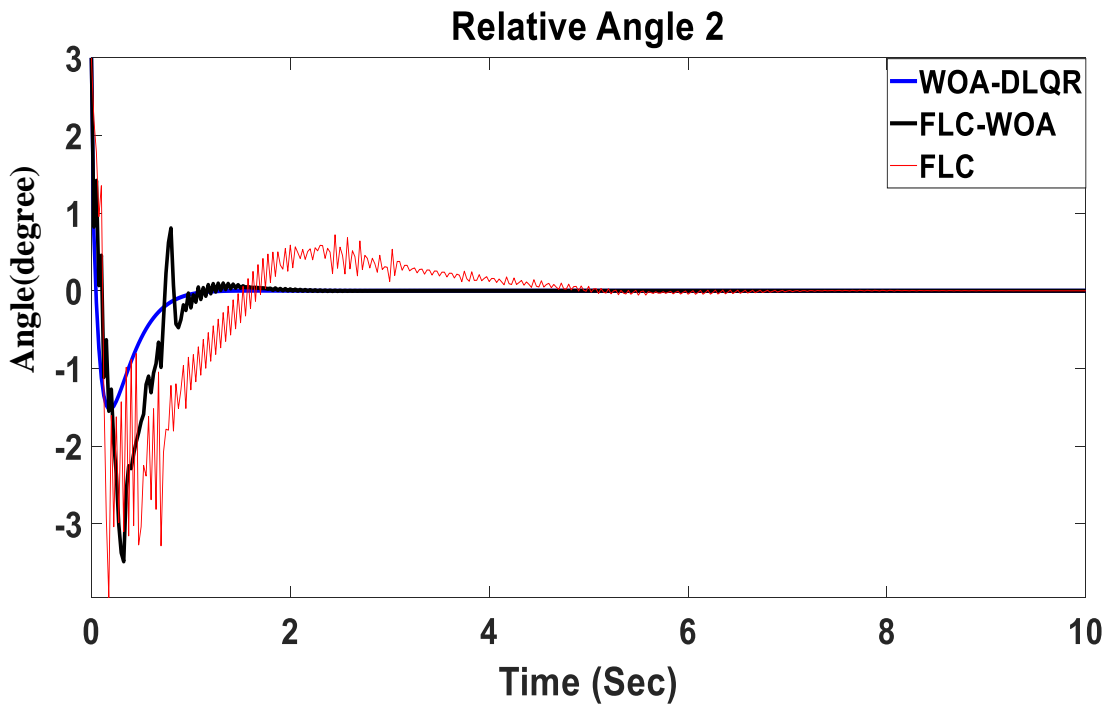
4.8.1 Comparison Between WOA-based DLQR, FLC, and Hybrid Controller

Fig. 4.13 compares the WOA-based DLQR, FLC, and FLC hybrid with WOA-based DLQR controllers. The results clearly show that the WOA-based DLQR controller gets the best result according to the transient response of the relative angles. The FLC response was the worst because it had higher settling time and control efforts. The hybrid controller consumed less voltage from two motors than the other type of controller. The settling time of the relative angles is equal in the WOA-based DLQR and hybrid controller, but the WOA-based DLQR has less deviation. Although the WOA-based DLQR controller is considered the best controller, when implemented practically, the WOA-based DLQR controller consumes time to perform the calculations required in

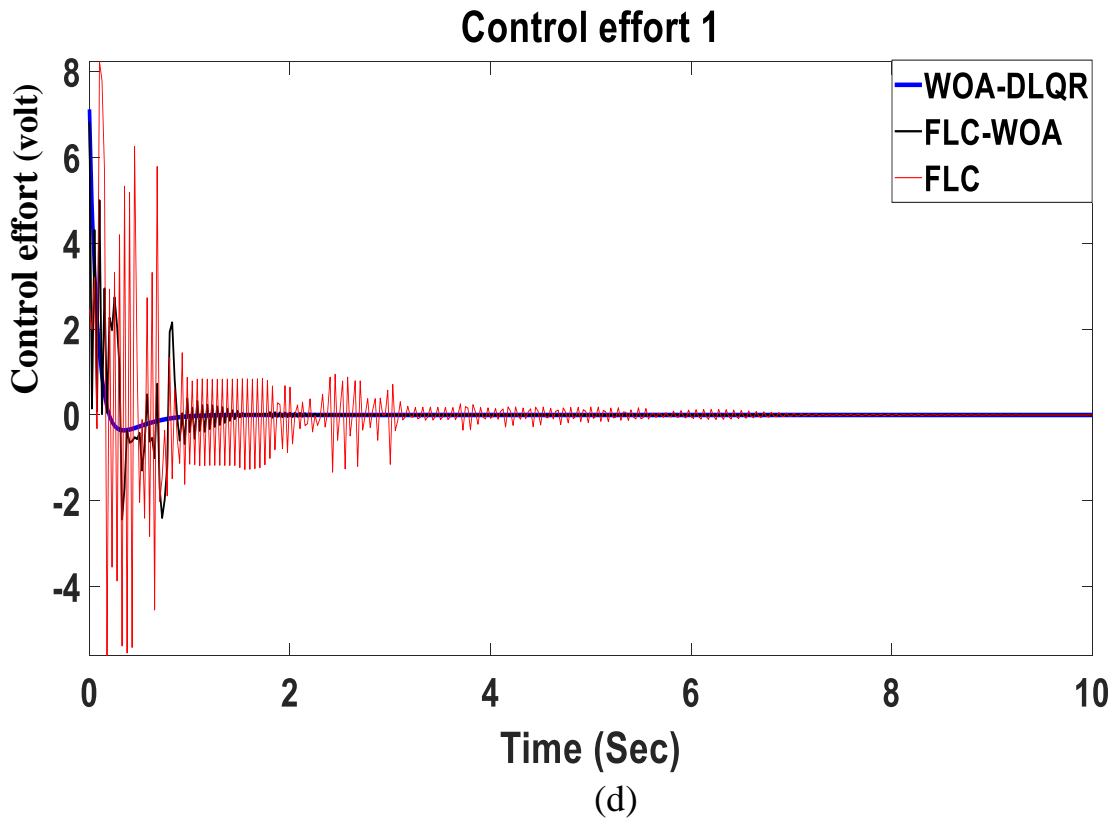
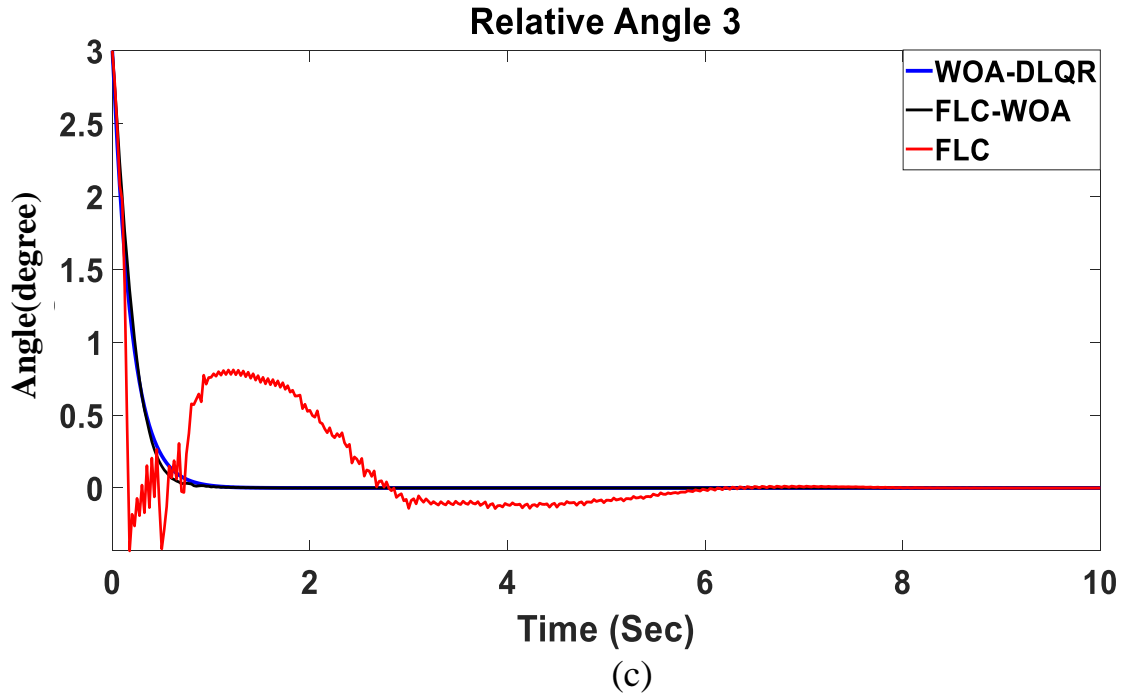
each case (off-line tuning). So, the hybrid system achieves online tuning with a satisfactory response to stabilize the gymnastic robot vertically.



(a)



(b)



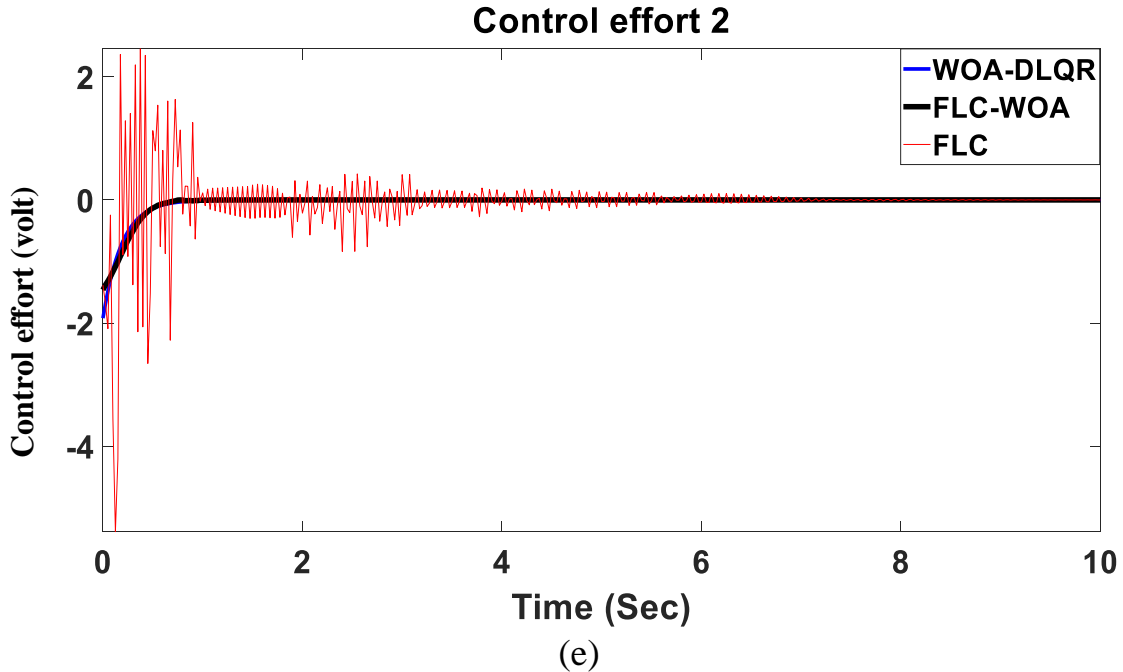


Fig. 4.13. Comparison time response between WOA-based DLQR, FLC and Hybrid controllers. (a) Relative angle 1 (b) Relative angle 2 (c) Relative angle 3 (d) Control effort 1 (e) Control effort 2.

4.9 Comparison with other work

In [5], to balance the Robogymnast, Kamil et al. developed a Discrete-time Linear Quadratic Regulator (DLQR). They adjusted the Q and R matrix using a trial-and-error method depending on the designer's experience. This method consumed 3.5827 seconds to achieve an upright position, and the simulation achieved a stable response. However, the robot consumed more settling time and overshoot than achieved when using WOA-based DLQR optimization to adjust the Q and R matrix. The first control effort rose to 10V (saturation limit), and the second control effort was 5.2 V, but the control effort that the author achieved ($u_1=8.5V$, $u_2=1.2V$) is less than this, as shown in Fig. 4.14 (b).

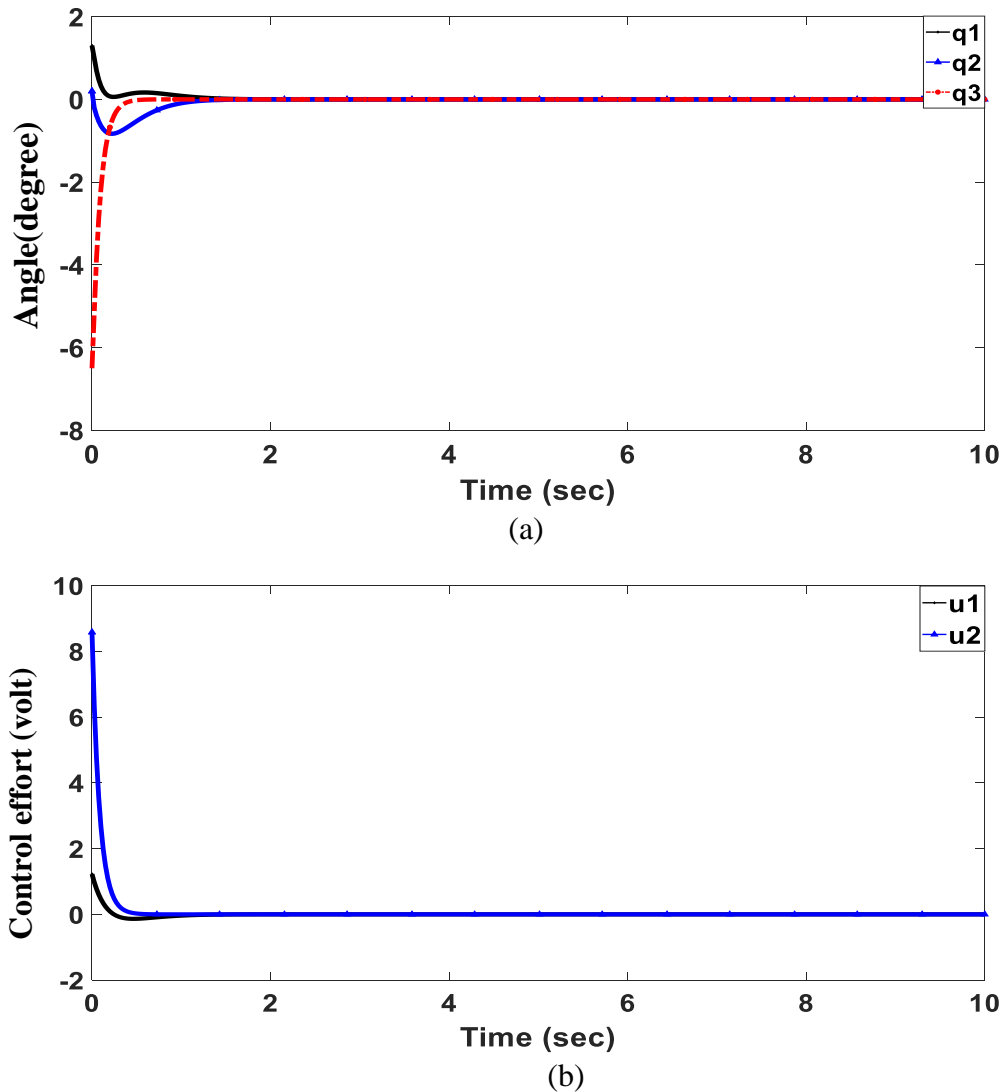


Fig. 4.14. Time response after WOA-based DLQR controller with initial $q_1=1.3^\circ$, $q_2=0.2^\circ$, $q_3=-6.5^\circ$. (a) Relative angle 1, 2, 3 (b) Control effort 1, 2.

The result achieved by the author, which is explained in Fig. 4.14 (a), is better than the simulation result achieved by another researcher in [5] (with initial relative angles $q_1=1.3^\circ$, $q_2=0.2^\circ$, $q_3=-6.5^\circ$). The comparison is according to the time response characteristics regarding the settling time, maximum deviation, and other criteria, as shown in Table 4.1.

Table 4-1: Compared performance analysis for angles response DLQR+WOA with DLQR

Relative angles	Controllers	Rise time (s)	Overshoot (degree)	Settling time(s)
q ₁	DLQR[5]	0.0201	-3.5832	1.4831
	DLQR+WOA	1.8	0	1.2760
q ₂	DLQR[5]	0.0106	7.8998	1.4771
	DLQR+WOA	0.013	-0.8	0.9334
q ₃	DLQR[5]	1.7482	0.0991	3.5827
	DLQR+WOA	0.5	0	0.7583

The result achieved by the author, which is explained in Fig. 4.5 (a), is better than the simulation result achieved by another researcher in [12] (with initial relative angles $q_1=3^\circ$, $q_2=3^\circ$, $q_3=3^\circ$) when compared according to transient response characteristics in terms of settling time and other criteria as shown in Table 4.2.

Table 4-2 : Compared performance analysis for relative angles response DLQR+WOA with DLQR+IWO

Relative angles	Controllers	Rise time (s)	Overshoot(degree)	Settling time(s)
q ₁	DLQR+IWO[12]	4	-29	5.4
	DLQR+WOA	0.0584	-0.6278	1.0949
q ₂	DLQR+IWO[12]	4	67	5.4
	DLQR+WOA	0.0339	-1.5255	0.9530
q ₃	DLQR+IWO[12]	2	10	3
	DLQR+WOA	0.4196	0.2883	0.7495

In [12], Ismail et al. applied invasive weed optimization (IWO) to determine the best possible Q matrix. According to an investigation, the 12V control effort consumed from two motors and this voltage above the control signal's saturation limit (10V) and the control effort ($u_1=2V$, $u_2=7.12V$) that the author achieved is less than this, as shown in Fig. 4.5 (b).

In [15], a multi-objective fuzzy logic hybrid invasive weed optimization (FLIWOH) method is presented. Fuzzy logic and the IWO are combined in this method. IWO is used in search and seed generation processes, and fuzzy logic is utilized to specify the fitness seeds by checking the JT (cost and settling time) criteria. The author's result in Fig. 4.11 is better than the result simulation in [15] (with initial relative angles $q_1=3^\circ$, $q_2=3^\circ$, $q_3=3^\circ$) compared to transient response characteristics as shown in Table 4.3. It is noticeable that the three relative angles take 6.37 seconds to achieve a stable upright position. Motor1 (u_1) and Motor2 (u_2) have maximum voltages of 12 and 5.8159 volts, respectively. Motor 1 (u_1) is above the control signal's saturation limit (10V), and the control effort ($u_1 = 6.7 \text{ V}$, $u_2 = -1.5\text{V}$) that the author achieved is less than this, as shown in Fig. 4.11.

Table 4-3: Compared performance analysis for relative angles response FLC+WOA with FLIWOH.

Relative angles	Controllers	Rise time (s)	Overshoot (degree)	Settling time(s)
q_1	FLIWOH [15]	3	-30	6.375
	FLC+WOA	0.0584	-1.2	1.8
q_2	FLIWOH [15]	3	67	6.375
	FLC+WOA	0.0339	-3.4	2
q_3	FLIWOH [15]	1.5	13	4
	FLC+WOA	0.4196	0	0.7495

In [32], N. A. Sayer et al. utilized DLQR and LQG controllers to stabilize a nonlinear triple inverted pendulum, represented by a gymnastic robot in the vertical plane. The simulation results demonstrated that LQR outperformed LQG, achieving superior results in terms of overshoot (-4, 9, 1.1), rise time (0.05013s, 0.07519s, 0.02506s), and settling time (3.208s, 3.233s, 4.16s) for the three links (first, second, third). However, the results presented in Fig. 4.15 surpassed those of the simulation in [32], as shown in Table 4.4. When a

WOA-based DLQR controller was employed, the three relative angles reached settling times of (1.2708s, 0.9362s, 0.7579s) and overshoots of (0.425, -1.902, 0), with rise times of (0.0771s, 0.4242s, 0.4234s) for the first, second, and third links, respectively. The system achieved quicker stabilization in the vertical position and reduced overshoot. Motor1 and Motor2 consumed 1.3 volts, significantly less than the consumption reported in [32] (Motor1 and Motor2 consumed 9 volts and 0.7 volts, respectively, in [32]).

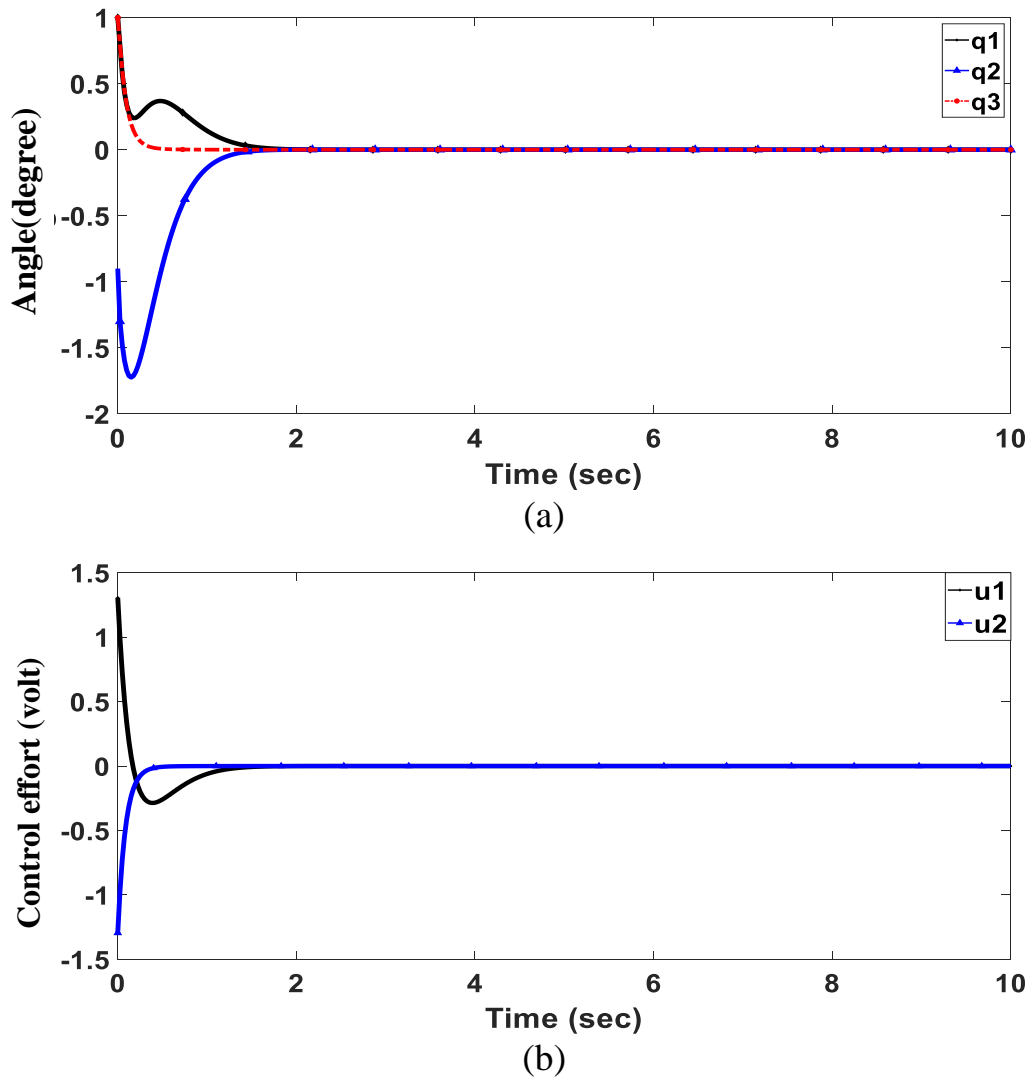


Fig. 4.15. Time response after WOA-based DLQR controller with initial $q_1=1^\circ$, $q_2=-0.9^\circ$, $q_3=1^\circ$. (a) Relative angle 1, 2, 3 (b) Control effort 1, 2.

Table 4-4: Compared performance analysis for angles response
DLQR+WOA with DLQR

Relative angles	Controllers	Rise time (s)	Overshoot (degree)	Settling time(s)
q ₁	DLQR [32]	0.05013	-4	3.208
	DLQR+WOA	0.0771	0.425	1.2708
q ₂	DLQR [32]	0.07519	9	3.233
	DLQR+WOA	0.4242	-1.902	0.9362
q ₃	DLQR [32]	0.02506	1.1	4.16
	DLQR+WOA	0.4234	0	0.7579

4.10 Summary

In this chapter, the balance problem of the gymnastics robot was addressed by applying several types of controllers. The second section discusses the application of DLQR to the Euler-Lagrange method and Artificial Neural Network Modeling. The results of applying DLQR to ENN were better as they provided stability with a starting angle of 3 degrees. The third section explores the application of WOA-based DLQR, with an evaluation against another optimization method, AO-based DLQR, showing that WOA-based DLQR yielded the best results. In the fourth section, FLC was applied, demonstrating the successful balancing of the gymnast robot in the vertical position. The fifth section combines FLC with WOA-based DLQR, resulting in the best response and lowest voltage consumption. Finally, in the sixth and seventh sections, a comparison is made between the methods used and previous research, revealing that WOA-based DLQR with FLC produced the best outcomes.

Chapter Five: Conclusion, Contributions, and Future Work

5.1 Conclusion

This study has aimed to develop three-degree-of-freedom control strategies for a gymnastic robot to balance in an inverted position with different Robogymnast situations. Robogymnast mimics human acrobatics with triple links and triple joints. Mathematical modeling presents an estimation of real-world systems. The Euler-Lagrange formula is used to derive the mathematical dynamics of the system. The gymnastic robot is an under-actuated, nonlinear multi-link mechanism requiring a complex mathematical model considering information accuracy. So, the ENN model is used in nonlinear system modeling.

DLQR controller is used to balance the gymnastic robot in an upright position. DLQR is applied to the Euler-Lagrange model and the ENN model. The comparison results show that the dynamic model obtained by the ANN is significantly better than the model derived from the Euler-Lagrange formula because the ANN model accepted the initial deviation of absolute angle for each link up to 3 degrees. In contrast, the dynamic model derived from the Euler-Lagrange formula accepted 1 degree and becomes unstable at 3 degrees.

The WOA is applied to the DLQR controller to adjust its parameters and self-adjust the weight matrices. The simulation results after optimization demonstrated that the first, second, and third links' overshoot angular positions and settling time are less than that achieved before optimization. The Robogymnast could be stabilized in the upright balancing point within a

suitable time, and motors consumed less voltage. Also, the AO is utilized to discover the best matrix for the DLQR controller and successfully achieve stabilizing transient response characteristics and control effort. The comparison between WOA-based DLQR and AO-based DLQR shows the WOA-based DLQR controller gets the best result according to the transient response characteristic with minimum deviation from the upright balancing point and settling time. The first, second, and third links reached a steady state after 1.825 seconds with a minimum deviation (-0.5° and -1.5° for the first and second links, respectively, and no deviation for the third link). Moreover, the control voltage of the first motor consumed 7.12V, and the second motor consumed 2V to achieve the desired response, which is less than the limited voltage (12V). However, because the optimization process involves computations for each case, it takes longer to implement practically and causes the controller to become offline.

After that, FLC was designed to achieve online tuning to stabilize and balance the system. The result of the FLC showed that the system consumed more settling time to be stable in an inverted position. So, a hybrid controller FLC with WOA-based DLQR to achieve online tuning with less settling time for the relative angular position (1.5 seconds) and acceptable undershoot of the links from the upright balancing point (-1.15° and -3.4° for the first and second links, respectively, and no undershoot of the third link). The first motor consumed 6.7 volts of control effort, but the second motor consumed only 1.5 volts; this was considered satisfactory voltage to become the Robogymnast in an inverted position.

Finally, the comparison with previous research shows that the FLC with the WOA-based DLQR method achieves less overshoot, settling time, and control effort than the other methods.

5.2 Contributions

- Analyze and understand the system performance by applying a DLQR controller with various Q and R matrix ranges.
- Apply the DLQR controller using the mathematical formula derived from Euler-Lagrange equations and the mathematical model calculated from experimental results by applying a neural network. A comparison between both methods has been introduced.
- Achieve self-tuning for the Q and R matrix by applying the WOA and AO evaluated.
- Achieve online tuning gains by applying a Fuzzy Logic controller.
- Design hybrid controllers (FLC and WOA) for achieve the best result.

5.3 Future Work

The study effectively met its defined objectives, and the presented simulations and results offer insights for additional research. This section outlines several worthwhile suggestions for exploring ways to improve system performance and introduce enhanced capabilities for advanced applications:

- Employ various optimization methods to control the transition of the robot's swing from a stable to an unstable position.
- Integrate Swinging-Up and Balancing Control for a comprehensive approach.
- Employ this model as a framework for testing different control algorithms and as a basis for creating diverse applications.
- Explore the adaptation of the designed controller concerning changes in the durations of external disturbances.

References

- [1] S. Shigemi, “ASIMO and Humanoid Robot Research at Honda,” *Humanoid Robotics: A Reference*. pp. 55–90, 2018. doi: 10.1007/978-94-007-6046-2_9.
- [2] M. A. Paldhe, “Software Architecture and Development for Controlling a Hubo Humanoid Robot,” 2014, [Online]. Available: https://docs.lib.purdue.edu/open_access_theses
- [3] A. J. Bautista and S. O. Wane, “ATLAS Robot:A Teaching Tool for Autonomous Agricultural Mobile Robotics,” *ICCAIS 2018 - 7th International Conference on Control, Automation and Information Sciences*. pp. 264–269, 2018. doi: 10.1109/ICCAIS.2018.8570494.
- [4] E. E. Eldukhri and D. T. Pham, “Autonomous swing-up control of a three-link robot gymnast,” *Proc. Inst. Mech. Eng. Part I J. Syst. Control Eng.*, vol. 224, no. 7, pp. 825–833, 2010, doi: 10.1243/09596518JSCE948.
- [5] HG Kamil, “Intelligent Model-Based Control of Complex Three-Link Mechanisms,” no. April, 2015.
- [6] H. Azizi and B. Ismail, “INTELLIGENT MODEL-BASED CONTROL OF COMPLEX MULTI-LINK MECHANISMS A thesis submitted to Cardiff University in the candidature for the,” no. December, 2016.
- [7] N. A. Sayer, H. G. Kamil, and A. A. Al-Moadhen, “Improving of Swing up Motion Control Parameters for a Gymnastics Robot Using the Gray Wolf Algorithm,” *Int. J. Intell. Syst. Appl. Eng.*, vol. 11, no. 6s, pp. 441 – 450, May 2023, Accessed: Jun. 13, 2023. [Online]. Available: <https://www.ijisae.org/index.php/IJISAE/article/view/2869>
- [8] A. Z. Alassar, “Modeling and Control of 5DOF robot Arm Using

- Supervisory Control thesis Submitted in Partial F requirements for the Degree of Master of Science in,” no. March, 2010.
- [9] S. Sehgal and S. Tiwari, “LQR control for stabilizing triple link inverted pendulum system,” *ICPCES 2012 - 2012 2nd Int. Conf. Power, Control Embed. Syst.*, 2012, doi: 10.1109/ICPCES.2012.6508052.
- [10] V. R. Molazadeh, A. Banazadeh, and I. Shafieenejad, “Design of the LQR controller and observer with fuzzy logic GA and GA-PSO algorithm for triple an inverted pendulum and cart system,” *Int. Conf. Adv. Mechatron. Syst. ICAMechS*, pp. 295–300, 2014, doi: 10.1109/ICAMechS.2014.6911560.
- [11] H. G. Kamil, E. E. Eldukhri, and M. S. Packianather, “Balancing Control of Robot Gymnast Based on Discrete-Time Linear Quadratic Regulator Technique,” *Proc. - 2nd Int. Conf. Artif. Intell. Model. Simulation, AIMS 2014*, pp. 137–142, 2014, doi: 10.1109/AIMS.2014.38.
- [12] H. A. Ismail, M. S. Packianather, R. I. Grosvenor, and E. E. Eldhukri, “The application of IWO in LQR controller design for the Robogymnast,” *IntelliSys 2015 - Proc. 2015 SAI Intell. Syst. Conf.*, no. March 2016, pp. 274–279, 2015, doi: 10.1109/IntelliSys.2015.7361154.
- [13] D. C. Dracopoulos and B. D. Nichols, “Genetic programming for the minimum time swing up and balance control acrobot problem,” *Expert Systems*, vol. 34, no. 5. 2017. doi: 10.1111/exsy.12115.
- [14] N. F. Jamin and N. A. Ghani, “Two-wheeled wheelchair stabilization control using fuzzy logic controller based particle swarm optimization,” *Proc. - 2016 IEEE Int. Conf. Autom. Control Intell. Syst. I2CACIS 2016*, no. November, pp. 180–185, 2017, doi: 10.1109/I2CACIS.2016.7885311.

- [15] H. A. Ismail, M. S. Packianather, and R. I. Grosvenor, “Multi-objective invasive weed optimization of the LQR controller,” *Int. J. Autom. Comput.*, vol. 14, no. 3, pp. 321–339, Jun. 2017, doi: 10.1007/S11633-017-1061-3/METRICS.
- [16] T. Yaren and S. Kizir, “Stabilization control of triple pendulum on a cart,” *2018 6th Int. Conf. Control Eng. Inf. Technol. CEIT 2018*, no. October, pp. 1–6, 2018, doi: 10.1109/CEIT.2018.8751818.
- [17] R. Banerjee, N. Dey, U. Mondal, and B. Hazra, “Stabilization of Double Link Inverted Pendulum Using LQR,” *Proc. 2018 Int. Conf. Curr. Trends Towar. Converging Technol. ICCTCT 2018*, pp. 1–6, 2018, doi: 10.1109/ICCTCT.2018.8550915.
- [18] X. Xia, J. Xia, M. Gang, Q. Zhang, J. Wang, and J. Wang, “Discrete Dynamics-Based Parameter Analysis and Optimization of Fuzzy Controller for Inverted Pendulum Systems Based on Chaos Algorithm,” *Discret. Dyn. Nat. Soc.*, vol. 2020, 2020, doi: 10.1155/2020/3639508.
- [19] A. F. Ghalib and A. A. Oglah, “Design and Implementation of a Fuzzy Logic Controller for Inverted Pendulum System Based on Evolutionary Optimization Algorithms,” *Eng. Technol. J. J. homepage engtechjournal.org Technol. J.*, vol. 38, no. 03, pp. 361–374, 2020, doi: 10.30684/etj.v38i3A.400.
- [20] M. A. Ebrahim, M. E. Mousa, E. M. Said, M. Mahmoud Zaky, and S. A. Kotb, “Optimal Design of Hybrid Optimization Technique for Balancing Inverted Pendulum System,” *Wseas Trans. Syst.*, vol. 19, pp. 138–148, 2020, doi: 10.37394/23202.2020.19.19.
- [21] M. K. Habib and S. A. Ayankoso, “Modeling and Control of a Double Inverted Pendulum using LQR with Parameter Optimization through GA and PSO,” *2020 21st Int. Conf. Res. Educ. Mechatronics, REM*

- 2020, pp. 0–5, 2020, doi: 10.1109/REM49740.2020.9313893.
- [22] H. G. Kamil, O. T. Makki, and H. M. Umran, “Optimal tuning of a Linear Quadratic Regulator for Position Control using Particle Swarm Optimisation,” *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 671, no. 1, 2020, doi: 10.1088/1757-899X/671/1/012047.
- [23] N.-K. Nguyen, V.-N. Pham, T.-C. Ho, and T.-M.-P. Dao, “Designing an Effective Hybrid Control Strategy to Balance a Practical Inverted Pendulum System,” *Int. J. Eng. Trends Technol.*, vol. 70, pp. 80–87, 2022, doi: 10.14445/22315381/IJETT-V70I5P210.
- [24] N.-K. Nguyen *et al.*, “Balancing a Practical Inverted Pendulum Model Employing Novel Meta-Heuristic Optimization-based Fuzzy Logic Controllers,” *Artic. Int. J. Adv. Comput. Sci. Appl.*, vol. 13, no. 4, p. 2022, 2022, doi: 10.14569/IJACSA.2022.0130464.
- [25] A. Mourad, Y. Zennir, and C. Tolba, “Intelligent and Robust Controller Tuned with WOA: Applied for the Inverted Pendulum,” *J. Eur. des Syst. Autom.*, vol. 55, no. 3, pp. 359–366, 2022, doi: 10.18280/jesa.550308.
- [26] M. Mohamed, F. Anayi, M. Packianather, B. A. Samad, and K. Yahya, “Simulating LQR and PID controllers to stabilise a three-link robotic system,” *2022 2nd Int. Conf. Adv. Comput. Innov. Technol. Eng. ICACITE 2022*, pp. 2033–2036, 2022, doi: 10.1109/ICACITE53722.2022.9823512.
- [27] B. A. Samad, F. Anayi, Y. Melikhov, M. Mohamed, and E. Altayef, “Modelling of LQR and Fuzzy-LQR Controllers for Stabilisation of Multi-link Robotic System (Robogymnast),” *2022 8th Int. Conf. Autom. Robot. Appl. ICARA 2022*, pp. 33–38, 2022, doi: 10.1109/ICARA55094.2022.9738577.
- [28] B. A. Samad, M. Mohamed, and G. S. Member, “Enhanced the Control

- Strategy of a Triple Link Robotic System (Robogymnast),” *IEEE Access*, vol. 11, no. April, pp. 31997–32005, 2023, doi: 10.1109/ACCESS.2023.3262190.
- [29] O. Saleem and J. Iqbal, “Fuzzy-Immune-Regulated Adaptive Degree-of-Stability LQR for a Self-Balancing Robotic Mechanism: Design and HIL Realization,” *IEEE Robot. Autom. Lett.*, vol. 8, no. 8, pp. 4577–4584, Aug. 2023, doi: 10.1109/LRA.2023.3286176.
- [30] T. ABUT, “Optimal LQR Controller Methods for Double Inverted Pendulum System on a Cart,” *DÜMF Mühendislik Dergisi*. 2023. doi: 10.24012/dumf.1253331.
- [31] S. Erjon, B. Xhevahir, L. Rame, and P. Arbnor, “REAL TIME SWINGING UP AND STABILIZING A DOUBLE INVERTED PENDULUM USING PID-LQR,” vol. 73, no. 1, pp. 159–168, 2023, doi: 10.2478/scjme-2023-0013.
- [32] N. A. Sayer, G. Kamil, and A. A. Al-Moadhen, “Control Design for the Balancing of an Automated Three-Link Gymnastics Robot Based Discrete-Time Linear Quadratic Regular Technique,” vol. 03, no. 04, 2023, Accessed: Jan. 28, 2024. [Online]. Available: <https://kjes.uokerbala.edu.iq>
- [33] K. Ogata and J. W. Brewer, “Modern Control Engineering,” *Journal of Dynamic Systems, Measurement, and Control*, vol. 93, no. 1. pp. 63–63, 1971. doi: 10.1115/1.3426465.
- [34] S. Mirjalili and A. Lewis, “The Whale Optimization Algorithm,” *Adv. Eng. Softw.*, vol. 95, pp. 51–67, 2016, doi: 10.1016/j.advengsoft.2016.01.008.
- [35] L. Abualigah, D. Yousri, M. Abd Elaziz, A. A. Ewees, M. A. A. Al-qaness, and A. H. Gandomi, “Aquila Optimizer: A novel meta-heuristic

- optimization algorithm,” *Comput. Ind. Eng.*, vol. 157, p. 107250, 2021, doi: 10.1016/j.cie.2021.107250.
- [36] B. Sasmal, A. G. Hussien, A. Das, and K. G. Dhal, “A Comprehensive Survey on Aquila Optimizer,” *Arch. Comput. Methods Eng.* 2023 307, vol. 30, no. 7, pp. 4449–4476, Jun. 2023, doi: 10.1007/S11831-023-09945-6.
- [37] L. A. Zadeh, “Fuzzy logic,” *Computational Complexity: Theory, Techniques, and Applications*, vol. 9781461418. pp. 1177–1200, 2013. doi: 10.1007/978-1-4614-1800-9_73.
- [38] “(PDF) Fuzzy swing-up and stabilization of real inverted pendulum using single rulebase.” https://www.researchgate.net/publication/268427647_Fuzzy_swing-up_and_stabilization_of_real_inverted_pendulum_using_single_rulebase (accessed Jan. 11, 2024).
- [39] H. J. Zimmermann, “Fuzzy set theory,” *Wiley Interdiscip. Rev. Comput. Stat.*, vol. 2, no. 3, pp. 317–332, May 2010, doi: 10.1002/WICS.82.
- [40] K. Ong and M. Piovoso, “Fuzzy Logic Control of a Double Inverted Pendulum.” 2018. Accessed: Jan. 11, 2024. [Online]. Available: <http://hdl.handle.net/10066/20450>
- [41] G. Garofalo, C. Ott, and A. Albu-Schäffer, “Walking control of fully actuated robots based on the bipedal SLIP model,” *Proc. - IEEE Int. Conf. Robot. Autom.*, pp. 1456–1463, 2012, doi: 10.1109/ICRA.2012.6225272.
- [42] M. Khadiv, S. A. A. Moosavian, and M. Sadedel, “Dynamics modeling of fully-actuated humanoids with general robot-environment interaction,” *2014 2nd RSI/ISM Int. Conf. Robot. Mechatronics, ICRoM 2014*, pp. 233–238, Dec. 2014, doi: 10.1109/ICROM.2014.6990906.

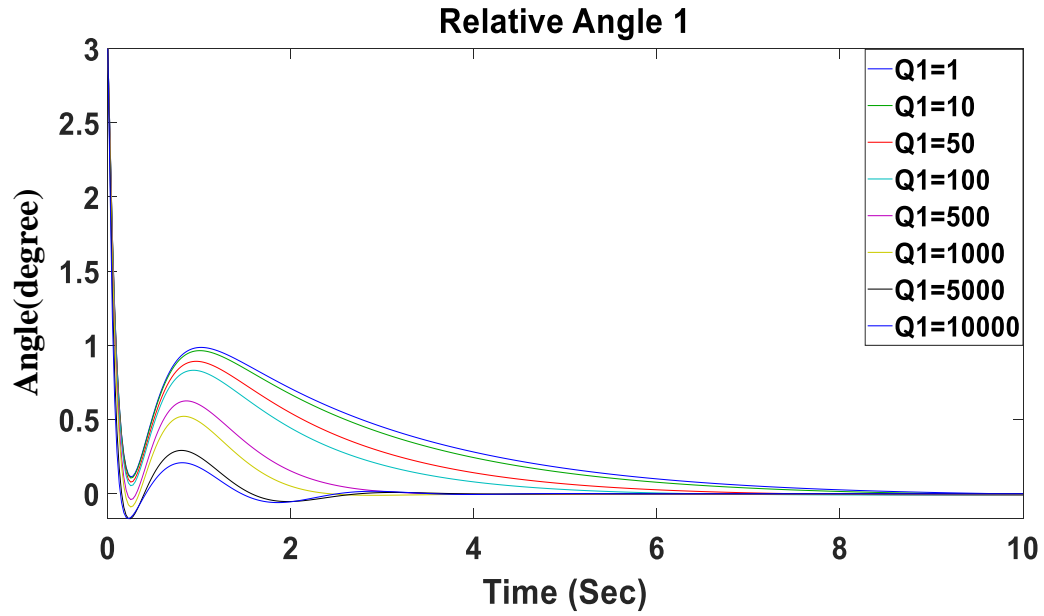
- [43] Y. Gu, B. Yao, and C. S. George Lee, “Exponential Stabilization of Fully Actuated Planar Bipedal Robotic Walking with Global Position Tracking Capabilities,” *J. Dyn. Syst. Meas. Control. Trans. ASME*, vol. 140, no. 5, May 2018, doi: 10.1115/1.4038268/368288.
- [44] J. A. Saglia, N. G. Tsagarakis, J. S. Dai, and D. G. Caldwell, “A high performance 2-dof over-actuated parallel mechanism for ankle rehabilitation,” *Proc. - IEEE Int. Conf. Robot. Autom.*, pp. 2180–2186, 2009, doi: 10.1109/ROBOT.2009.5152604.
- [45] R. Seifried, “Dynamics of underactuated multibody systems: Modeling, control and optimal design,” *Solid Mechanics and its Applications*, vol. 205, pp. 1–260, 2014. doi: 10.1007/978-3-319-01228-5.
- [46] S. Krafes, Z. Chalh, and A. Saka, “A Review on the Control of Second Order Underactuated Mechanical Systems,” *Complexity*, vol. 2018, 2018, doi: 10.1155/2018/9573514.
- [47] G. A. Medrano-Cerda, E. E. Eldukhri, and M. Cetin, “Balancing and Attitude Control of Double and Triple Inverted Pendulums,” <http://dx.doi.org/10.1177/014233129501700306>, vol. 17, no. 3, pp. 143–154, Aug. 1995, doi: 10.1177/014233129501700306.
- [48] K. Furut, T. Ochiai, and N. Ono, “Attitude control of a triple inverted pendulum,” <https://doi.org/10.1080/00207178408933251>, vol. 39, no. 6, pp. 1351–1365, 2007, doi: 10.1080/00207178408933251.
- [49] K. G. Eltohamy and C. Y. Kuo, “Nonlinear optimal control of a triple link inverted pendulum with single control input,” <http://dx.doi.org/10.1080/002071798222811>, vol. 69, no. 2, pp. 239–256, Jan. 2010, doi: 10.1080/002071798222811.
- [50] “Control System By Norman nise Sixth Ed. | Rabbi Hossain - Academia.edu.”

- https://www.academia.edu/35425584/Control_System_By_Norman_ni_se_Sixth_Ed (accessed Jan. 11, 2024).
- [51] L. Erik and J. Perkins, “Comparison of Techniques for Stabilization of a Triple Inverted Pendulum,” *Am. Control Conf. 2006*, no. 2, pp. 1814–1819, 2006.
- [52] C. W. Tao, J. Taur, J. H. Chang, S. Su, and S. Member, “Adaptive Fuzzy Switched Swing-Up and Sliding Control for the Double-Pendulum-and-Cart System,” vol. 40, no. 1, pp. 241–252, 2010.
- [53] Y. S. Hussein, H. G. Kamil, and A. A. Al-Moadhen, “Robust Hybrid Controller Design to Stabilise an Underactuated Robot Vehicle under Various Input,” *J. Robot.*, vol. 2022, 2022, doi: 10.1155/2022/5477391.
- [54] Z. Ben Hazem and Z. Bingul, “Comprehensive Review of Different Pendulum Structures in Engineering Applications,” *IEEE Access*, vol. 11, pp. 42862–42880, 2023, doi: 10.1109/ACCESS.2023.3269580.

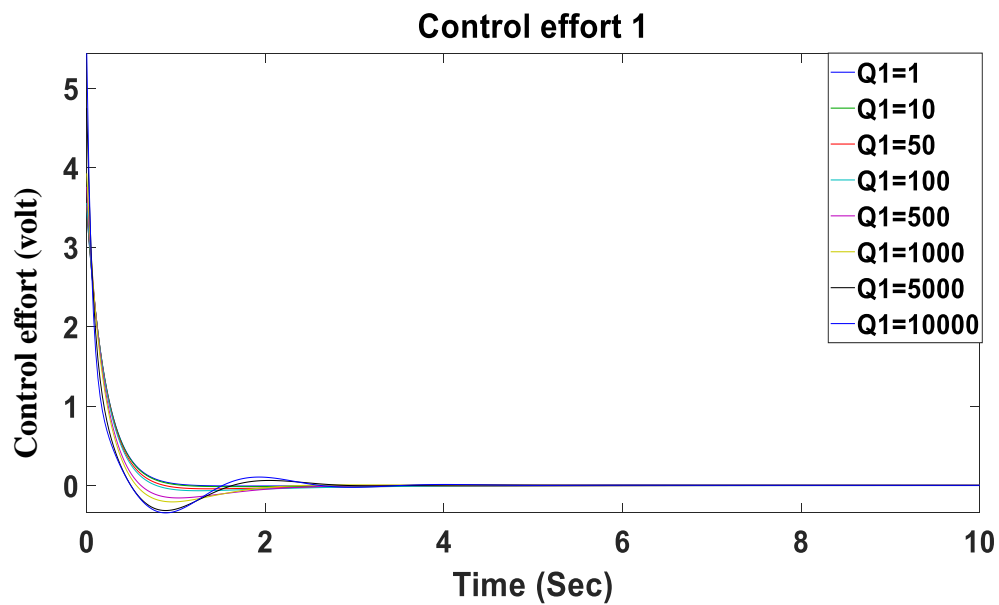
Appendices

Appendix A

A.1. $Q1=1-10000$, $(Q2, Q3, Q4, Q5, Q6, R1, R2) = 1$.



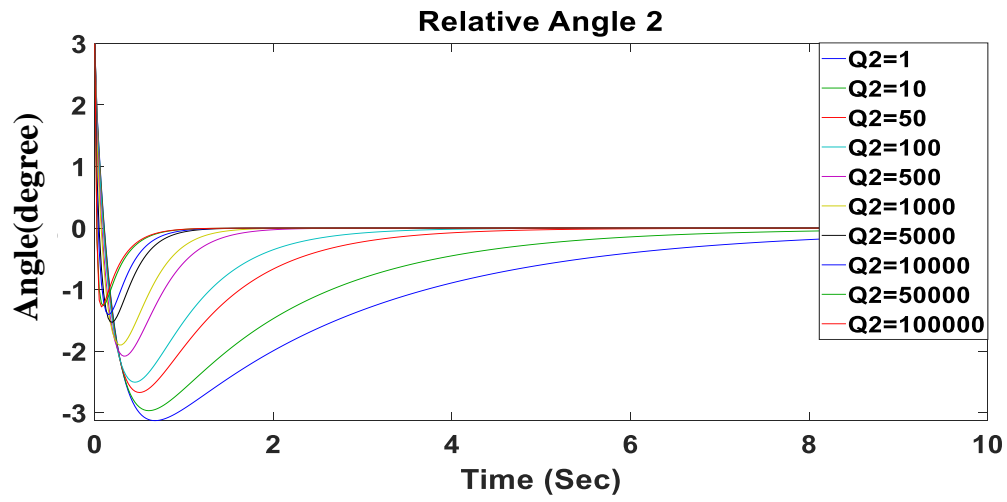
(a)



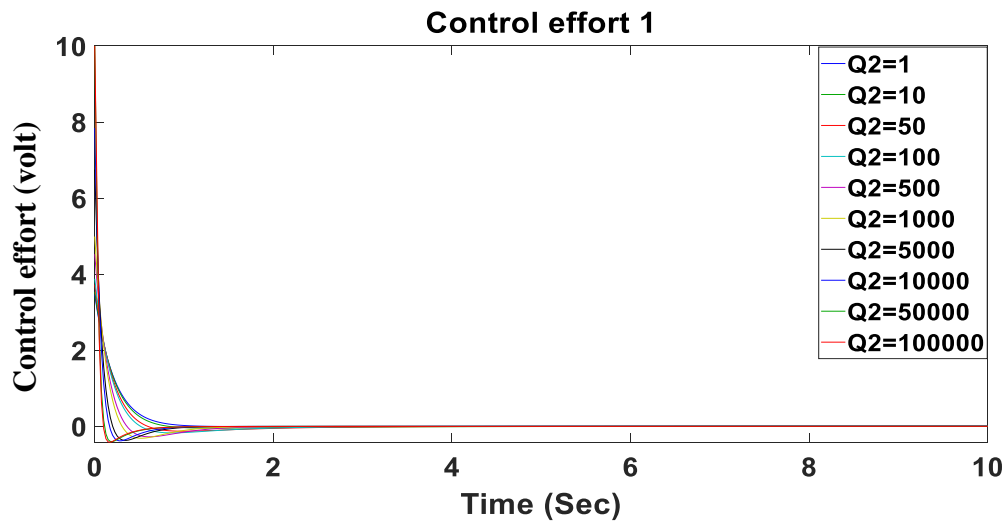
(b)

Fig. 5.1.1 Range of $Q1$. (a) Relative angle 1 (b) Control effort 1.

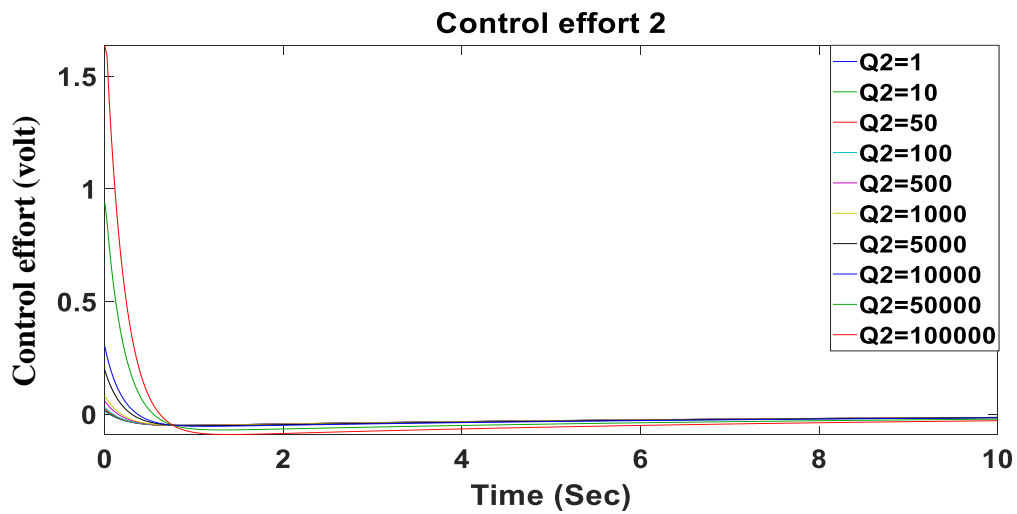
A.2. $Q_2=1-100000$, $(Q_1, Q_3, Q_4, Q_5, Q_6, R_1, R_2) = 1$.



(a)



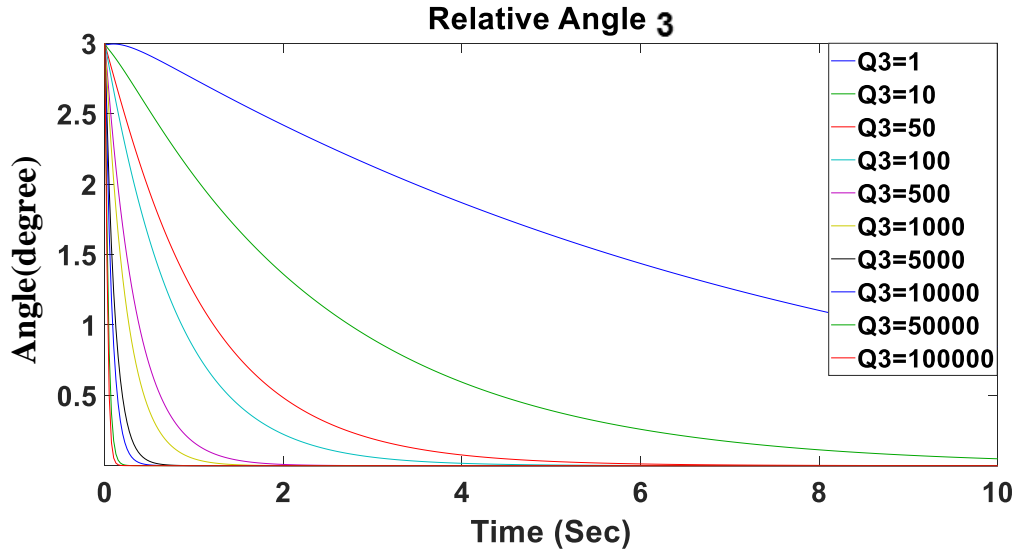
(b)



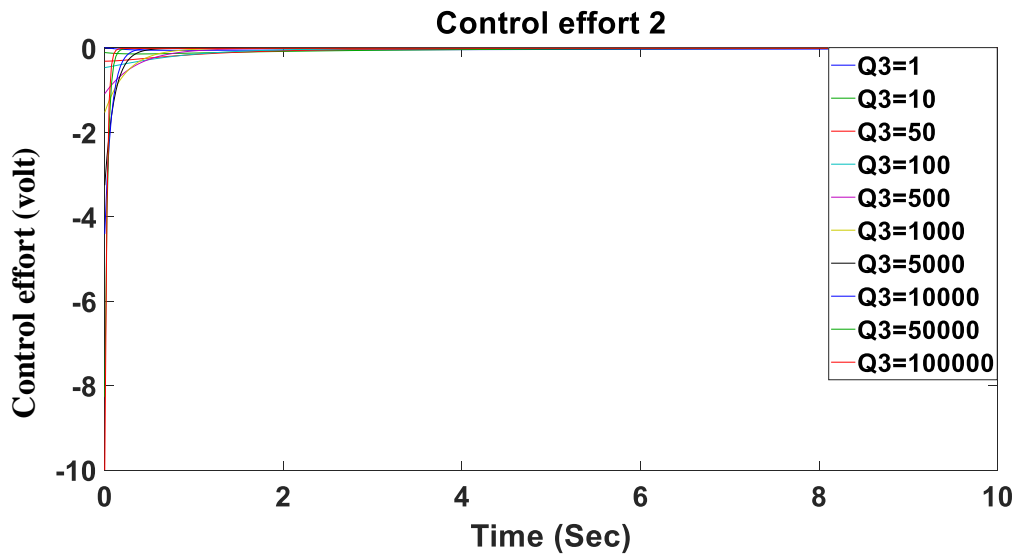
(c)

Fig. 5.2.1 Range of Q2. (a) Relative angle 1 (b) Control effort 1(c) Control effort 2.

A.3.Q3=1-100000, (Q1, Q2, Q4, Q5, Q6, R1, R2) =1.



(a)



(b)

Fig. 5.3.1 Range of Q3. (a) Relative angle 3 (b) Control effort 2.

A.4. $Q_4=1-100000$, $(Q_1, Q_2, Q_3, Q_5, Q_6, R_1, R_2) = 1$.

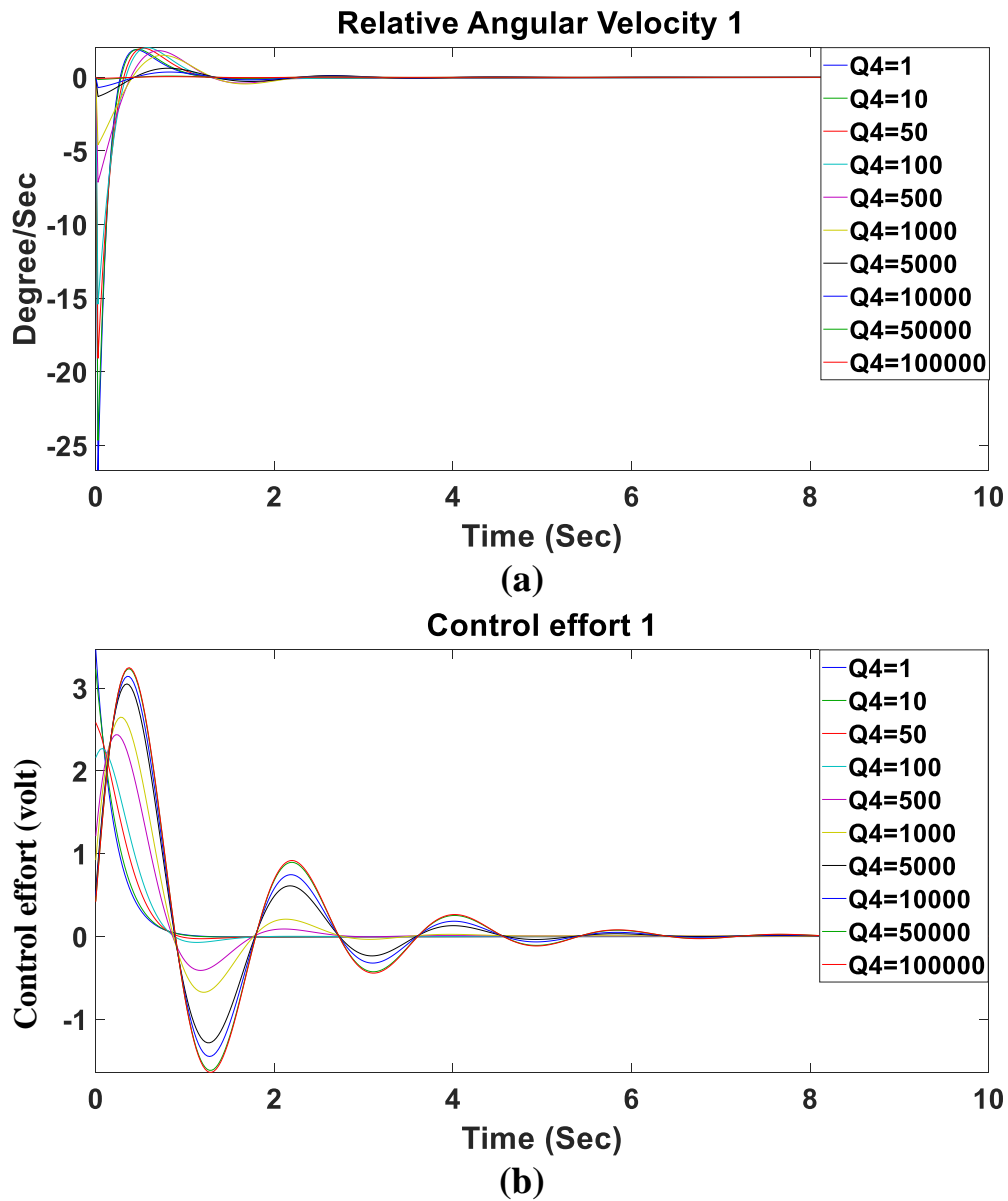
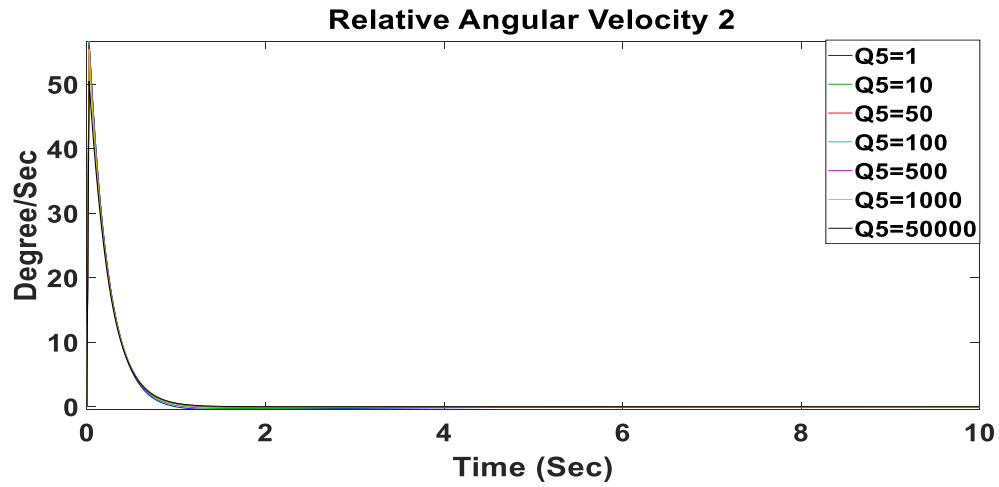
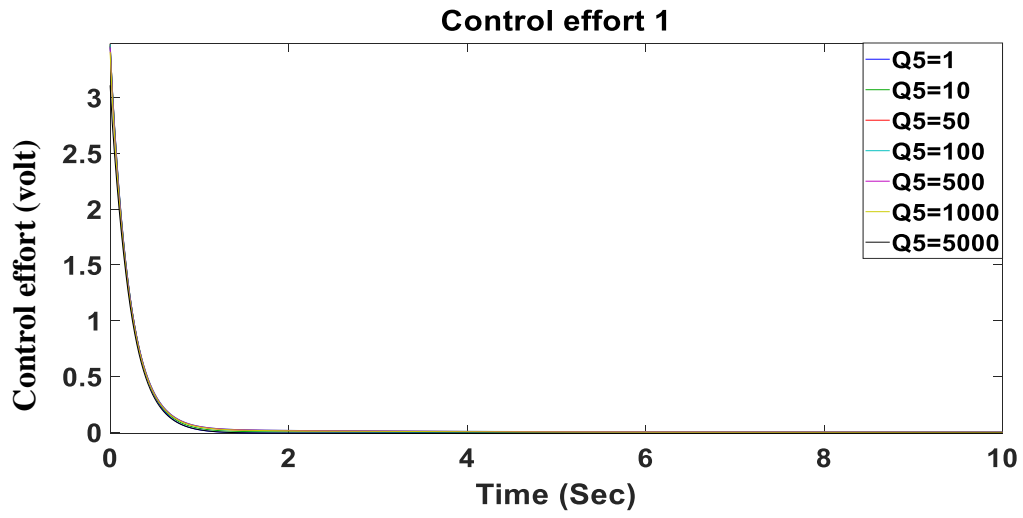


Fig. 5.4.1 Range of Q_4 . (a) Relative Angular Velocity 1 (b) Control effort 1.

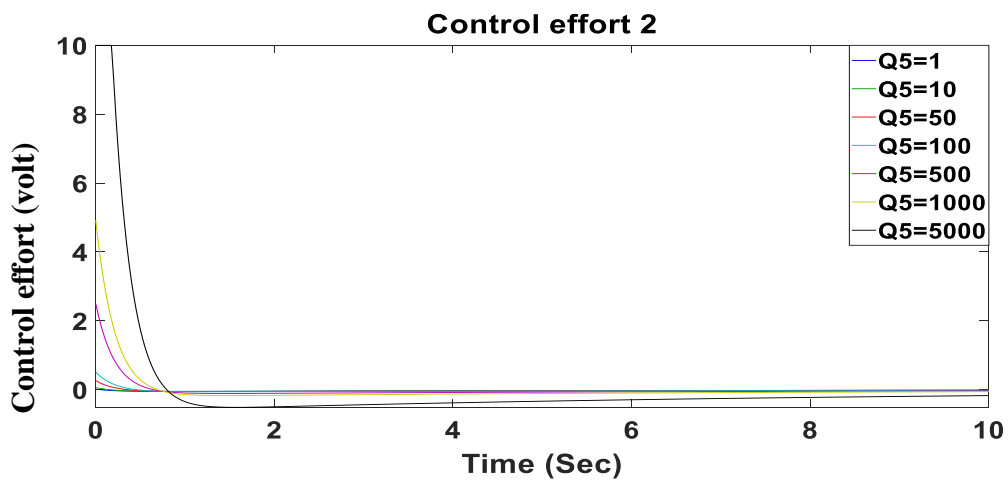
A.5. $Q5=1-5000$, $(Q1, Q2, Q3, Q4, Q6, R1, R2) = 1$.



(a)



(b)



(c)

Fig. 5.5.1 Range of Q5. (a) Relative Angular Velocity 2 (b) Control effort 1(c) Control effort 2.

A.6. Q6=1-10000, (Q1, Q2, Q3, Q4, Q5, R1, R2) =1.

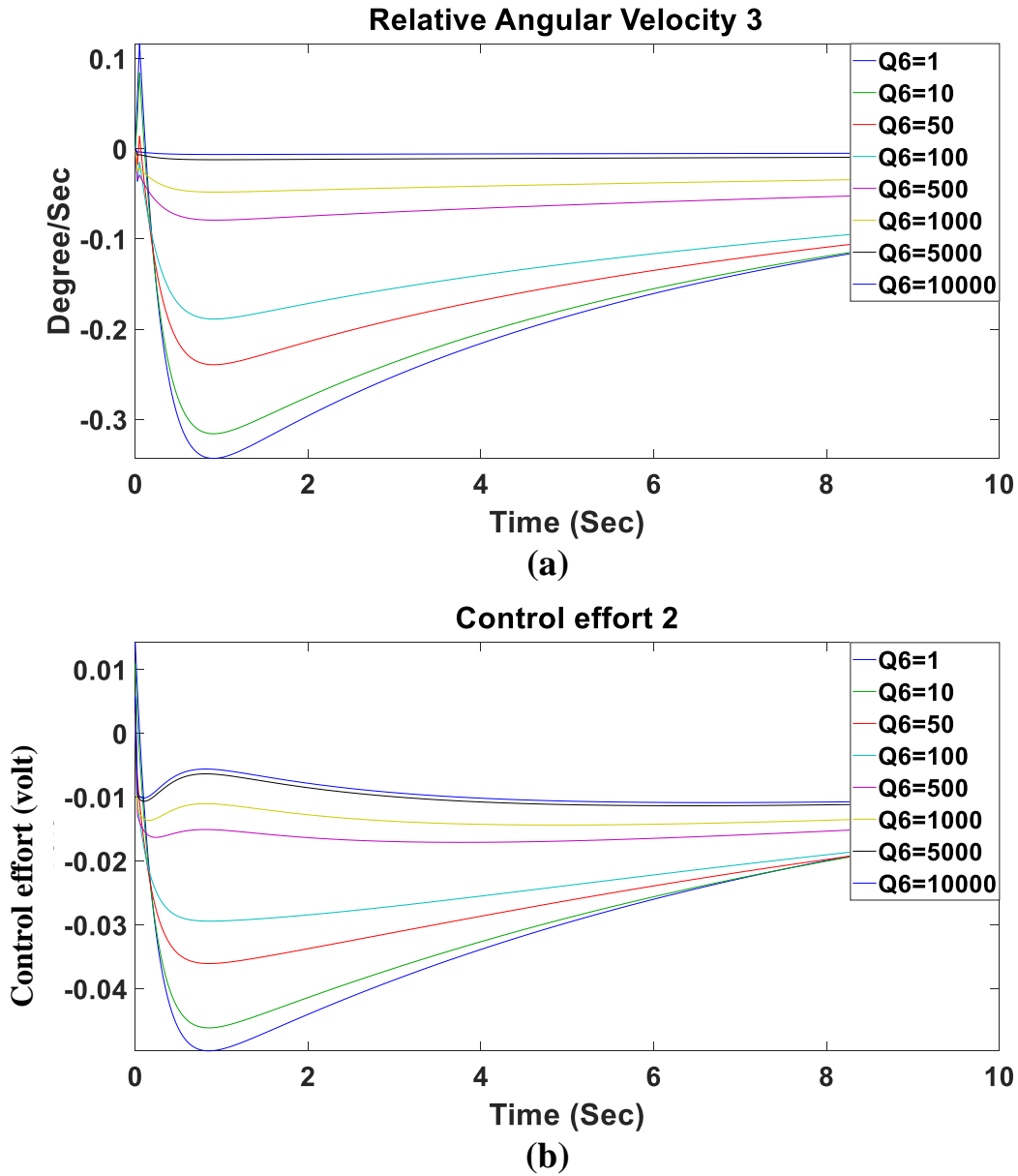
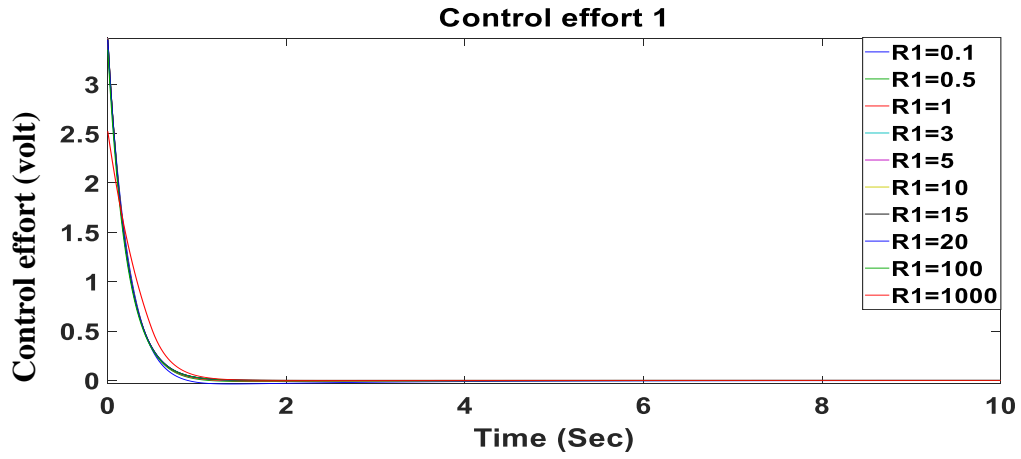
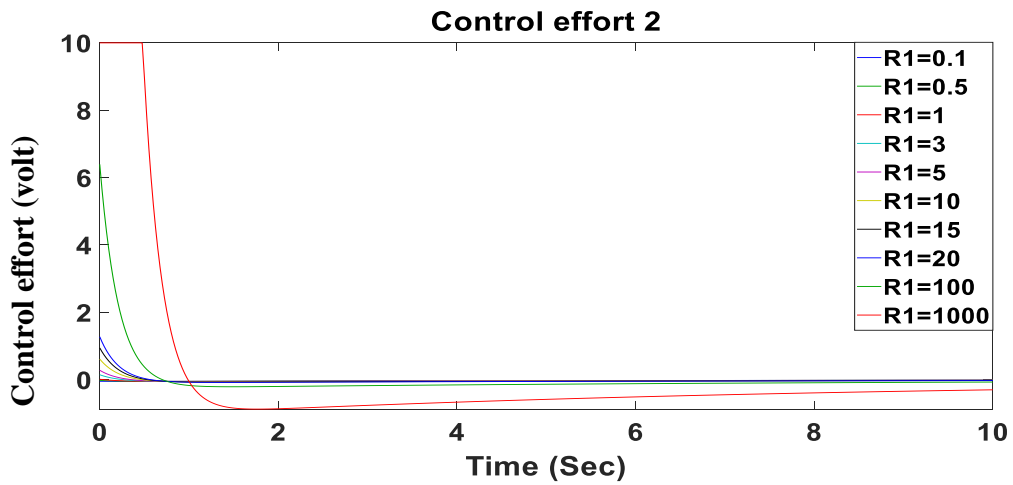


Fig. 5.6.1 Range of Q6. (a) Relative Angular Velocity 3 (b) Control effort 2.

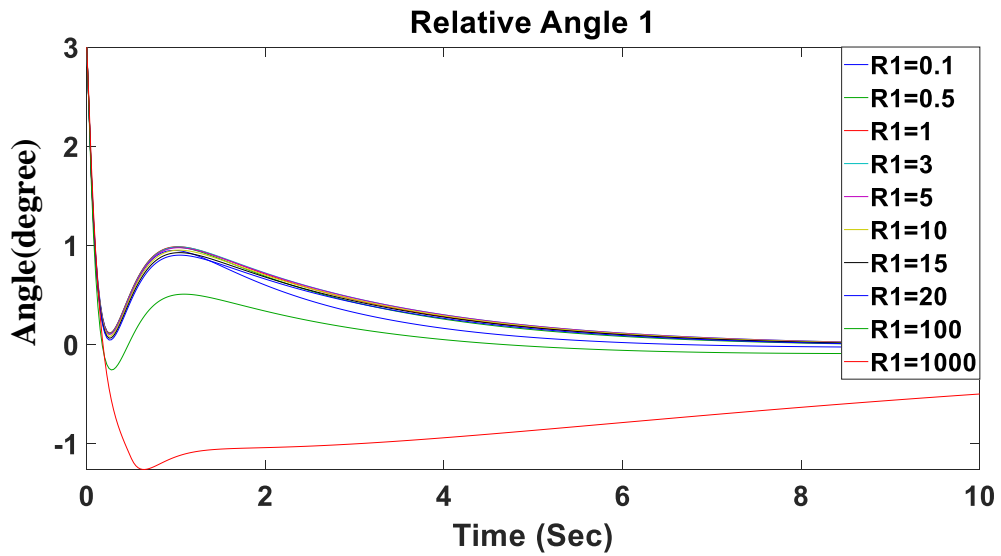
A.7. $R_1=1000$, $(Q_1, Q_2, Q_3, Q_4, Q_5, Q_6, R_2) = 1$.



(a)



(b)



(c)

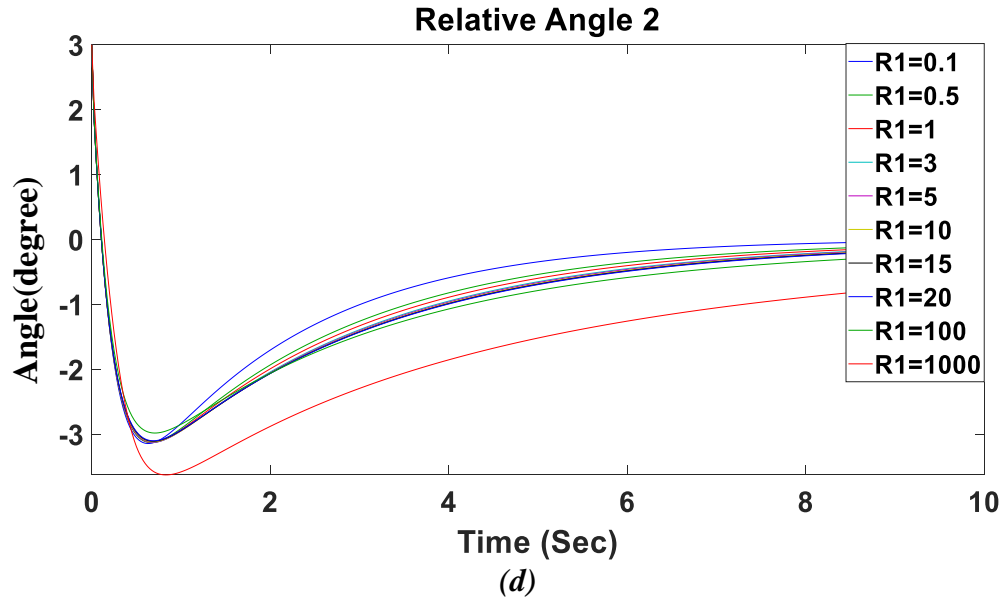
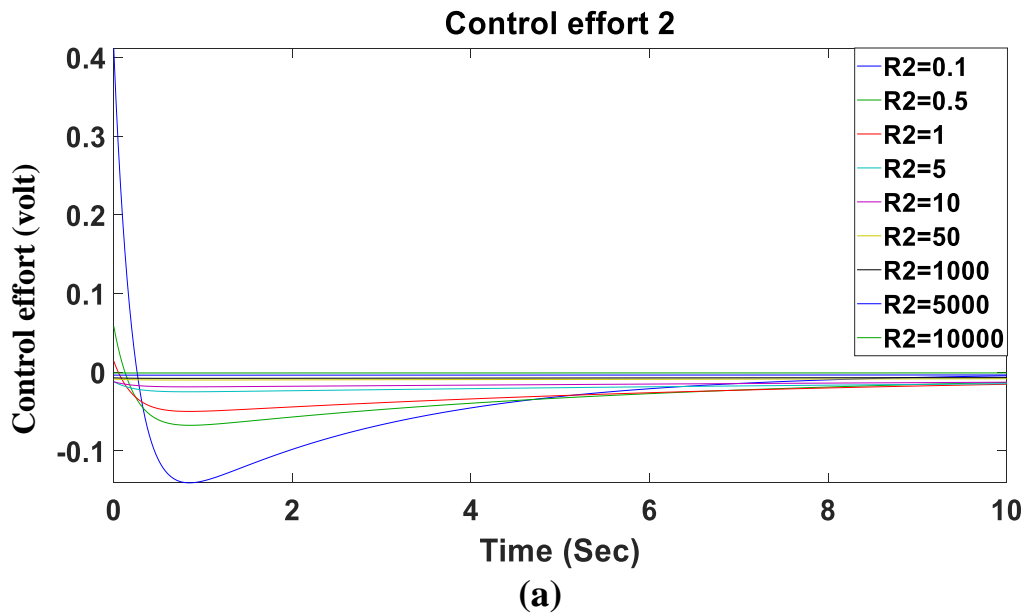
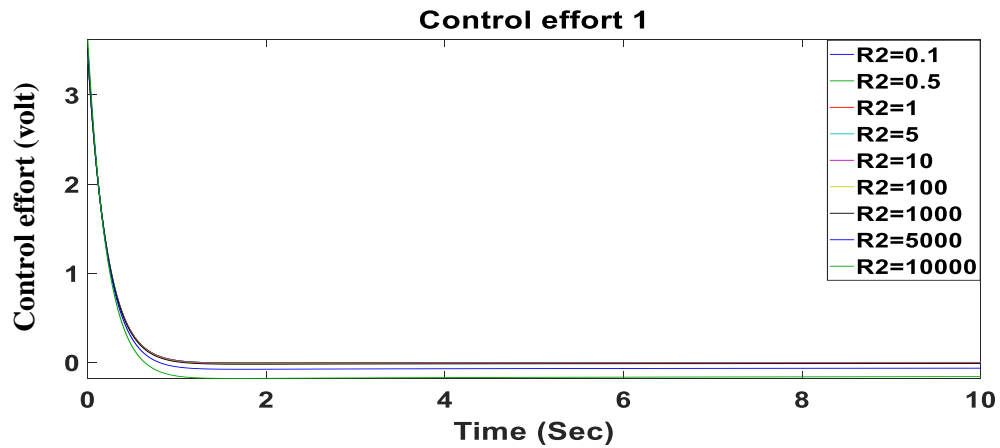


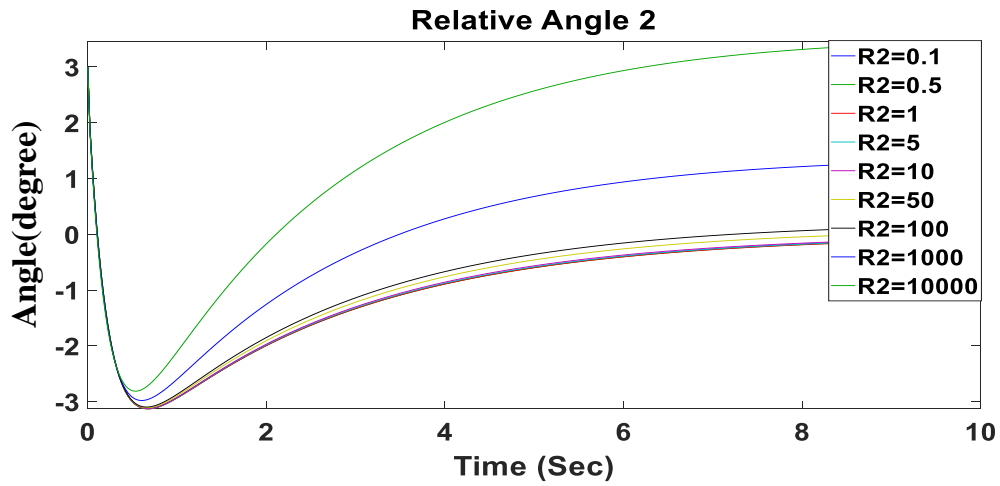
Fig. 5.7.1 Range of R1. (a) Control effort 1 (b) Control effort 2 (c) Relative angle 1 (d) Relative angle 2.

A.8. $R_2=10000$, $(Q_1, Q_2, Q_3, Q_4, Q_5, Q_6, R_1) = 1$.

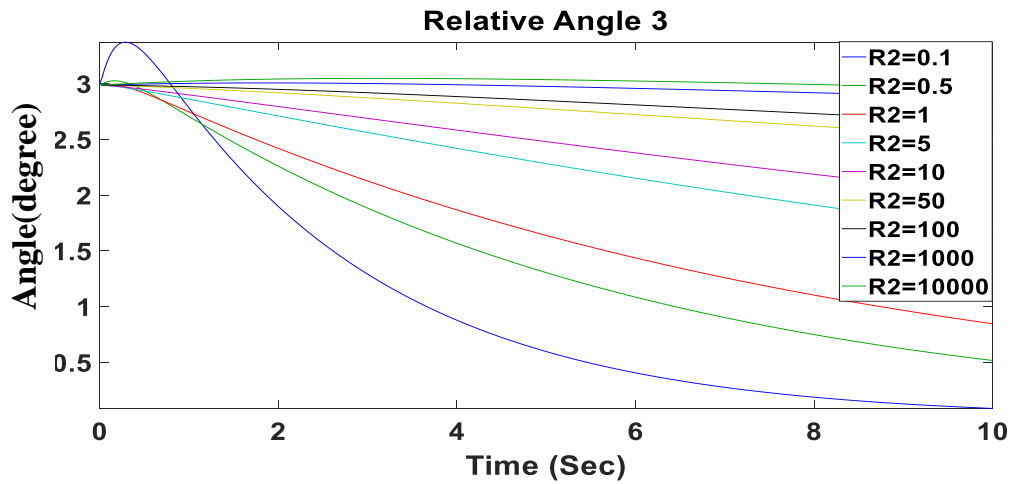




(b)



(c)



(d)

Fig. 5.8.1 Range of R_2 . (a) Control effort 2 (b) Control effort 1 (c) Relative angle 2 (d) Relative angle 3.

الخلاصة

تهدف هذه الدراسة إلى فهم التعقيد وموازنة التحكم في الوضع المستقيم لنظام الروبوت ثلاثي الوصلات. يعتبر لاعب الجمباز الآلي أحد الأنواع المهمة من الأنظمة ثلاثية الوصلات التي تحاكي الألعاب البهلوانية البشرية؛ وهو يتألف من ثلاثة مفاصل وثلاث وصلات (الذراع والجذع والساق، على التوالي) مدعومة بمحركين يعملان بالتيار المستمر.

تم اشتقاق نموذج رياضي للروبوت باستخدام معادلات لاغرانج. حيث أن النظام عبارة عن آلية غير خطية متعددة الارتباطات تتطلب نموذجًا رياضيًا معقدًا يأخذ في الاعتبار دقة المعلومات. إنه يقدم المزيد من التحديات في نمذجة لعبة الجمباز الروبوتية والتعامل مع مشاكل التحكم في الحركة. يتم استخدام صيغة لاغرانج ونموذج الشبكة العصبية الاصطناعية لنمذجة نظام Robogymnast غير الخطي.

أولاً، يتم استخدام وحدة التحكم في المنظم التريبيعي الخطي المنفصل (DLQR) لتحقيق التوازن بين الروبوت الجمباز في الوضع المستقيم. يعتمد بناء DLQR على اختيار مصفوفات الوزن. ثانياً، إيجاد القيم المثلى لمصفوفات الترجيح؛ يتم تطبيق تقنية تحسين السرب تسمى خوارزمية تحسين الحوت (WOA) لضبط مصفوفات الترجيح. بالإضافة إلى ذلك، يتم استخدام تقنية تحسين أخرى للعثور على القيم المثلى لمصفوفات الوزن، وتسمى هذه التقنية (AO) Aquila Optimization. تم تنفيذ تقييم أفضل تقنية. تحقق وحدة التحكم DLQR المستندة إلى WOA أفضل نتيجة وفقاً للاستجابة العابرة للزوايا النسبية ولكنها تستهلك جهداً أعلى من المحركين مقارنة بوحدة التحكم DLQR المستندة إلى AO. وصلت الوصلات الأولى والثانية والثالثة إلى حالة الاستقرار بعد 1.825 ثانية مع أدنى انحراف (-0.5° و 1.5° للوصلتين الأولى والثانية على التوالي، وعدم وجود انحراف للوصلة الثالثة). علاوة على ذلك فإن جهد التحكم للمحرك الأول يستهلك 7.12 فولت، والمحرك الثاني يستهلك 2 فولت لتحقيق الاستجابة المطلوبة، وهي أقل من الجهد المحدود (12 فولت).

ثالثاً، تم تصميم وحدة التحكم المنطقية المضيبة (FLC) لتحقيق الضبط عبر الإنترنت لتحقيق الاستقرار والتوازن في النظام. أظهرت نتيجة FLC أن النظام يستهلك وقتاً أطول للتسوية من DLQR المستند إلى WOA ليكون مستقرًا في الوضع المقلوب. لذلك، تم اقتراح وحدة تحكم هجينة تجمع بين FLC و DLQR المستندة إلى WOA لتحقيق الضبط عبر الإنترنت مع وقت استقرار أقل

للموضع الزاوي النسبي (1.5 ثانية) وانحراف مقبول للوصلات من نقطة التوازن المستقيمة (-1.15 درجة و-3.4 درجة) للوصلتين الأولى والثانية على التوالي، ولا يوجد انحراف للوصلة الثالثة). استهلك المحرك الأول 6.7 فولت من جهد التحكم، لكن المحرك الثاني استهلك -1.5 فولت فقط؛ كان هذا يعتبر جهدًا مرضيًا لجلب لاعب الجمباز الآلي إلى وضع مقلوب وتثبيتته في نقطة التوازن المستقيمة خلال مدة مناسبة.

وأخيراً، أظهرت المقارنة بين الطرق السابقة أن النظام الهجين يحقق الضبط المباشر مع استجابة مرضية لتثبيت الروبوت الجمباز عمودياً. توضح المقارنة مع الأبحاث السابقة أن FLC مع طريقة DLQR المستندة إلى WOA تحقق أفضل استجابة عابرة فيما يتعلق بالتجاوز وتسوية الوقت وجهد تحكم أقل من الطرق الأخرى.



جمهورية العراق
وزارة التعليم العالي و البحث العلمي
جامعة كربلاء
كلية الهندسة
قسم الهندسة الكهربائية والإلكترونية

تحكم ذكي للروبوت قليل النشاط بناءً على تقنيات التحسين

رسالة مقدمة الى مجلس كلية الهندسة / جامعة كربلاء وهي جزء من متطلبات نيل درجة الماجستير في
علوم الهندسة الكهربائية

من قبل:

آيه فليح حسن

باشراف :

أ.د. حيدر جليل كامل

أ.م.د. أحمد عبدالهادي أحمد