LQR and Genetic Algorithms: An Effective Duo for Assessing Control Expenditure and Performance in Dynamic Systems

1st Josefredo Gadelha da Silva
Centre for Ocean Energy Research
Department of Electronic Engineering
Maynooth University
josefredo.silva.2024@mumail.ie

2nd Marcio J. Lacerda
Centre for Communications Technology
School of Computing and Digital Media
London Metropolitan University
m.lacerda@londonmet.ac.uk

3rd Ariadne L. J. Bertolin
Centre for Ocean Energy Research
Department of Electronic Engineering
Maynooth University
ariadne.bertolin@mu.ie

4th Jander Santos
Centre for Ocean Energy Research
Department of Electronic Engineering
Maynooth University
jander.santos@mu.ie

5th Erivelton Nepomuceno
Centre for Ocean Energy Research
Department of Electronic Engineering
Maynooth University
erivelton.nepomuceno@mu.ie

***Abstract*—In this work, a novel methodology is introduced that employs genetic algorithms to determine the optimal weighting matrices for a linear quadratic regulator controller. A method is presented to construct a multi-objective fitness function that allows one to give prioritisation to energy consumption or other performance metrics, such as rise time, settling time, and steady-state error. To validate the effectiveness of the proposed approach, we conducted simulation studies based on a model of an inverted pendulum on a cart system. The results show a reduction of up to 30.36% in the energy of the controller and a reduction of 20.27% in its maximum value when choosing to prioritise the energy expenditure of the controller over other performance metrics, without significantly compromising the convergence of the system states. The results encompass an effective way of optimising energy expenditure in non-linear controller designs.**

***Index Terms*—Nonlinear Dynamical Systems, Green Algorithm, Sustainable Circuits and Systems, LQR, Inverted Pendulum.**

I. INTRODUCTION

The celebrated Linear Quadratic Regulator (LQR) formulates a state-feedback gain by optimising a performance metric, which incorporates both a weighted state and control input. It holds a pivotal role in various control techniques. When compared to traditional methods such as feedback control based on pole placement, it offers the advantage of computing the state feedback control gain matrix, thereby steering the system towards optimal performance [1]. The relative superiority of LQR over other techniques such as PID was also pointed out in literature [2]. This technique has been broadly explored and has also been integrated with contemporary control approaches like MPC [3], PID [4], and Robust Control [5].

The design of LQR requires the selection of two Hermitian positive-definite matrices, Q and R, which dictate the balance between error significance and energy expenditure [6]. The issue is that different choices for Q and R lead the system to different responses, suggesting that the response may not be truly optimal [7]. Since the 1980s, the selection of LQR weighting matrices has been investigated [8], [9] and several approaches have been suggested to address the problem.

Non-parametric optimisation [10], multi-objective evolution algorithm [11], analytical method [12], [13], stochastic method [14] and other iterative methods such as particle swarm optimisation [15] are just a few examples of approaches that have been used to optimally select the tuning parameters of the LQR. Recently, interest in using genetic algorithms (GA) for the selection of weighting matrices has grown [16]–[19]. Chen et al. [20] used GA to tune LQR control for a flying inverted pendulum system. Poodeh et al. [21] combined LQR with the optimal location of poles using GA to control and enhance a Buck converter response. In [22] the authors improved the controller for path tracking of an autonomous 4WS electric vehicle by using GA to weight LQR matrices, among other examples. This illustrates the versatility of both LQR and GA techniques, as well as their applicability to various nonlinear dynamical systems.

It is well understood that a trade-off exists between the choices of Q and R matrices. Specifically, one can either prioritise the system's performance in terms of response convergence speed at the expense of control input or the other way around. While this trade-off is widely recognised, the precise relationship between the selection of Q and R, the control input, and other performance metrics remains ill-defined. In fact, there is a lack of studies in the literature addressing these subtle aspects. This emerges as an important factor because controlling the system is not the only concern; it is also vital to ensure that the control input remains within acceptable limits. A key consideration is that by minimising the control input, one could prolong the lifespan of actuators. Even more crucially, this can ensure the functionality of systems subjected to energy saturation. It has also been observed that beyond energy saturation, scientific investigation has recently started to consider the relevance of 'energy-optimality', in a way to reduce energy consumption and improve system endurance [23]. This stresses the importance of choosing which performance metrics prioritise during controller design.

In this work, a novel methodology is introduced that employs GA to determine the optimal matrices Q and R. The distinctive aspect of this approach is the emphasis on analysing the relationship between the weighting matrices and control input in terms of maximum effort and energy expenditure, and other system performance metrics. A method is presented to construct a multi-objective fitness function that minimises energy consumption while prioritising performance metrics, such as rise time, settling time, and steady-state error. To validate the effectiveness of the proposed approach, we conducted simulation studies based on a model of an inverted pendulum on a cart system.

The approach adopted in this paper encompasses an effective method for optimising energy expenditure in non-linear controller designs and, consequently, a way to reduce the effort exerted on actuators by the control input, prolonging its lifespan. Furthermore, it enhances the method of weighting the LQR controller. By adopting the approach presented in this paper, one can not only balance the trade-off between energy expenditure and performance but also make better choices regarding which performance metric to prioritise.

II. BACKGROUND

A. Genetic Algorithm

A Genetic Algorithm, first introduced by John Holland [24] is an optimisation technique that draws inspiration from natural selection processes. Mimicking evolutionary principles, GA refines a set of potential solutions, over multiple generations to approach an optimal solution. It begins by formulating a fitness function $f(k)$, where k is a test solution to the problem that one aims to minimise or maximise.

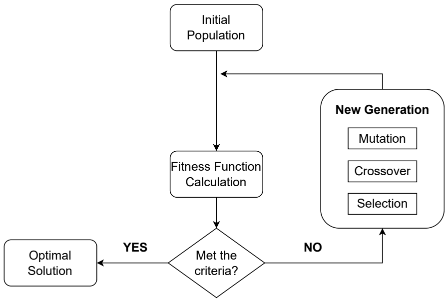

Fig. 1: General GA architecture according to [22].

Figure 1 summarises the GA optimisation process, which is run based on three typical genetic operations: i) *Selection*: refers to the process of favouring individuals to become parents based on the results of the fitness function. For the maximisation case, k_2 is more likely to be selected compared to k_1 if $f(k_2) > f(k_1)$; ii) *Crossover*: let k_1 and k_2 be potential solution of the optimisation problem, then $k_{12} = G(k_1,k_2)$, where G is a function that produces an offspring element, k_{12}, by operating on both parents, k_1 and k_2, selected in the previous step; iii) *Mutation*: introduces minor random variations in the offspring. The process continues by forming an entirely new generation and proceeds until the predefined criterion is met.

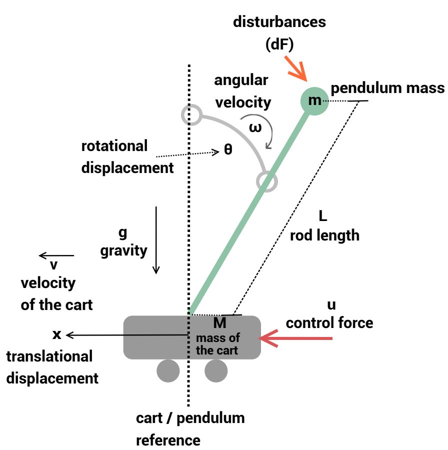

Fig. 2: Schematic: inverted pendulum on a cart system.

B. Inverted Pendulum on a Cart

The inherent complexity of stabilising an inverted pendulum on a cart presents significant challenges. Its natural instability and vulnerability to external disturbances have historically captivated the interest of control systems researchers.

Successfully achieving stability in this system can greatly influence numerous applications. Our proposed method comprehensively addresses the nonlinear dynamics of the inverted pendulum system, positioning it as a viable solution for practical implementations. A schematic of the system is depicted in Figure 2 and its dynamics are described by second-order nonlinear differential equations, as follows [25]:

$$\dot{x} = v, \tag{1}$$

$$\dot{v} = \frac{-m^2L^2g\cos(\theta)\sin(\theta) + mL^2(mL\omega^2\sin(\theta) - \delta v)}{mL^2(M + m(1 - \cos^2(\theta)))} + \frac{mL^2u}{mL^2(M + m(1 - \cos^2(\theta)))}, \tag{2}$$

$$\dot{\theta} = \omega, \tag{3}$$

$$\dot{\omega} = \frac{(m + M)mgL\sin(\theta) - mL\cos(\theta)(mL\omega^2\sin(\theta) - \delta v)}{mL^2(M + m(1 - \cos^2(\theta)))} + \frac{mL\cos(\theta)u}{mL^2(M + m(1 - \cos^2(\theta)))}, \tag{4}$$

where m is the pendulum mass, M is the mass of the cart, L is the length of a massless rod, x is the position of the cart, v is its velocity, θ is the angle of the pendulum, ω denotes the angular velocity, g is the gravitational acceleration, δ accounts for the frictional damping on the cart, and u is the control force applied to the cart.

C. Linear Quadratic Regulator

The primary goal of LQR is to determine a control law that minimises a quadratic cost function. This function is usually defined as the sum of the weighted quadratic differences between the system's state and a desired trajectory, and the weighted quadratic variations of the control inputs from their ideal values. We define J as the cost function to be minimised:

$$J = \int_0^\infty \left(x^TQx + u^TRu\right)\, dt, \tag{5}$$

where $x \in \mathbb{R}^n$ is the state vector, and $u \in \mathbb{R}^{n_u}$ is the control input vector. Both $Q \in \mathbb{R}^{n\times n}$ and $R \in \mathbb{R}^{n_u\times n_u}$ are the weighting matrices that relates the significance of state and control efforts within the optimisation process respectively [6]. Moreover we have $Q = Q^T \geq 0$ (positive semi-definite), and $R = R^T > 0$ (positive definite).

A state-space representation of the linearised version of the system yields

$$\dot{x} = Ax + Bu, \tag{6}$$

where $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times n_u}$ are obtained from the original nonlinear system. The state-feedback control law is defined by $u = -Kx$. The gain matrix $K \in \mathbb{R}^{n_u\times n}$ can be determined by solving the Riccati equation, which is defined as:

$$A^TP + AP - PBR^{-1}B^TP + Q = 0, \tag{7}$$

and the optimal gain is calculated as:

$$K = R^{-1}B^TP, \tag{8}$$

in which P is the solution matrix for the Riccati equation.

III. METHODOLOGY

The inverted pendulum setup was established as follows: $m = 0.1kg$, $M = 1kg$, $L = 0.5m$, and $g = 9.81m/s^2$. The GA considers Q as chromosomes and each one of its elements as alleles, the initial population initiates with 50 individuals, and a maximum of 500 generations is set. The optimal Q matrix is obtained by selecting, crossing over, and mutating each generation according to the criteria described below.

A. The Fitness Function

We define the controllability gramian as:

$$AW_c + W_cA^T + BB^T = 0, \tag{9}$$

where A and B are the system matrices. A good approximation for control energy can be achieved by defining a function $E(W_c)$ as:

$$E(W_c) = \text{trace}(W_c) = \sum_{i=1}^{n} \lambda_i, \tag{10}$$

Where λ_i are the eigenvalues of the controllability Gramian (W_c) and n is the order of the system. Let the fitness function derived from a weighted sum be defined as:

$$\text{F}(E,T_r,T_s,e) = G_1E(W_c) + G_2T_r + G_3T_s + G_4e, \tag{11}$$

where the additional terms T_r, T_s, and e represent the rise time, settling time, and steady-state error, respectively. We define T_r as the time when the system response is equal to or greater than 90% of the final value; T_s as the time at which the difference between the final value and zero is less than 2%; and e as the difference between the steady-state and zero. The idea of this fitness function is to allow the prioritisation of different parametric metrics by adjusting the weighting factors given by $G_1,\ldots,G_4$.

B. Selection Mechanism

The well-known roulette wheel is established as a method for selecting parents for the next generation of matrices. The idea behind this selection is to assign each member of the population a probability of being selected based on its fitness: the higher the fitness, the greater the chance of being selected. To determine which individual takes priority, the total fitness of the entire population is computed. This is followed by the calculation of the cumulative probability for each member. Members with greater fitness occupy larger sections of the "roulette wheel". When selecting an individual for updating, a random number between 0 and 1, referred to as the selector, is generated for each member in the population. The selector value then determines which member of the population is chosen.

C. Crossover Mechanism

The crossover rate is initially set to $C_r = 0.7$. Let Q_i denote the i^{th} parent matrices from the population. For each pair of parents (Q_i, Q_{i+1}) and (R_i, R_{i+1}), a random number, denoted by $\alpha \in [0, 1]$, is generated. If $\alpha < C_r$, a crossover operation is performed as:

$$Q_{\text{child1}} = \alpha Q_i + (1-\alpha)Q_{i+1}, \quad R_{\text{child1}} = \alpha R_i + (1-\alpha)R_{i+1},$$
$$Q_{\text{child2}} = (1-\alpha)Q_i + \alpha Q_{i+1}, \quad R_{\text{child2}} = (1-\alpha)R_i + \alpha R_{i+1}.$$

If $\alpha \geq C_r$, the children inherit the matrices directly from the parents:

$$Q_{\text{child1}} = Q_i, \qquad R_{\text{child1}} = R_i,$$
$$Q_{\text{child2}} = Q_{i+1}, \quad R_{\text{child2}} = R_{i+1}.$$

The offspring population maintains diversity while preserving the attributes of high-performing candidates.

D. Matrix Mutation Procedure

The mutation rate is initially set to $M_r = 0.5$. Let Q_i denote the i^{th} child matrix from the set of children matrices. For each Q_i, a decision to mutate is based on a probability governed by M_r. Specifically, if $rand() < M_r$, where $rand()$ denotes a generated random number, then the mutation is performed. The mutation is defined as:

$$Q_M = Q_i \times 0.1,$$

where Q_M is a scaled symmetric positive definite matrix with elements generated within the range $(0, 1000)$ with a scale factor of 0.1. The mutated child matrix is then given by:

$$Q_i^M = Q_i + Q_M$$

where Q_i^M is the mutated child matrix. This way, new regions in the solution space can be explored.

E. Constraints

The greater the value of R , the less the control effort. However, it is known that increasing the size of matrix R excessively may result in non-zero steady-state values. The proposed approach ensures that R is positive and $R \leq 1$. One could predefine the value of R to be the maximum, but allowing R to be defined by the GA provides a means to validate the algorithm.

IV. RESULTS

We utilised the software MATLAB, version R2023a, running on a computer equipped with a 12-core CPU CORE i9-10920XE, 64 GB of RAM, and a NVIDIA T400 4GB GPU with Windows 11 Enterprise, version 22H2 to conduct the simulations. All experiments were conducted for a linearised model of Equations (1) to (4). The routines used in this work are available at [26].

By allowing the GA to select the LQR parameters, the control operates effectively. Fig. **??** displays the convergence of the states when $G_1,\ldots,G_4 = 1$ were chosen to be equal. Fig. 4 highlights the effect of varying G_1 while keeping $G_2,\ldots,G_4 = 1$, equal to each other. One can observe that the control effort is minimised as G_1 increases. However, it is also evident that the system's sensitivity to the variance of this parameter is relatively low. This is tied to the criteria established for the selection, crossover, and mutation processes.

Let ε represent the energy of the control input signal. Then, the difference in energy of the control signal input with G_1 set to 1, 10, and 500, respectively, can be analysed by calculating the sum of the squared terms of each control signal. In this case, $\varepsilon_1 = 1220.00$, $\varepsilon_{10} = 992.6762$, and $\varepsilon_{500} = 776.3527$. Without significantly compromising the convergence of the states, one can reduce the control input energy by up to 36.36% when comparing ε_{500} with ε_1 and by 18.63% when comparing ε_{10} with ε_1. Furthermore, when assessing the maximum control input values, reductions of 9.78% and 20.27% were observed when comparing $G_1 = 1$ to $G_1 = 10$ and $G_1 = 500$, respectively.

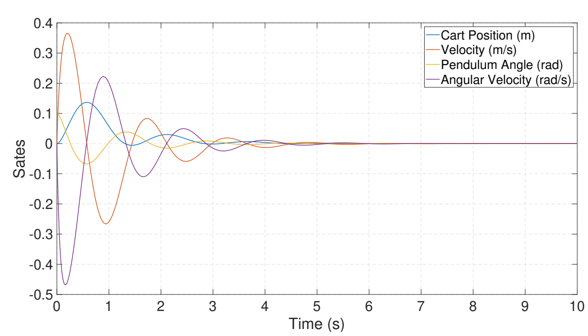

Fig. 3: Controlled states of the inverted pendulum on a cart.

On the other hand, Fig. 5 shows the effect of prioritising the system's settling time, keeping other weights equal to one and varying G_3 in three different values. The result shows that the outcome is

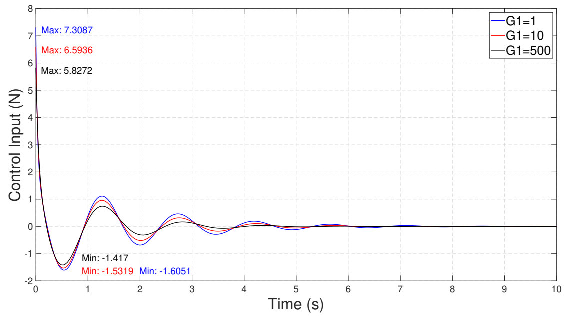

Fig. 4: Effects of varying the weight of the control expenditure parameter. The greater the weight G_1, the less the control expenditure.

effective. However, once again, it is understood that the system's sensitivity to change is relatively low.

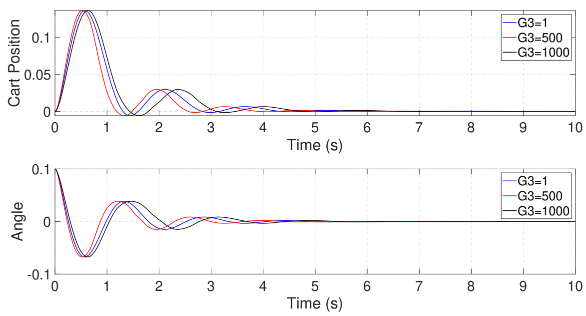

Fig. 5: Effects of varying the weight of the settling time parameter G_3. The greater the weight, the faster the response of the system.

Fig. 6 shows the evolution of the minimum value of the objective function for each generation, when $G_1, \ldots, G_4 = 1$. Gaps in the mentioned graph represent cases where the fitness function exceeded the imposed constraints. One can see that the method is also dependent on the number of generations. The greater the number of generations, the more refined the optimisation becomes. Thus, there is a trade-off between computational effort and the precision of this method.

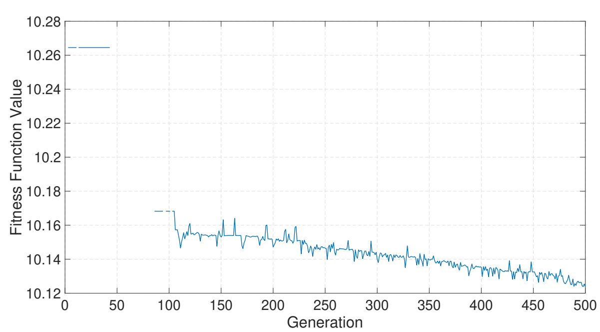

Fig. 6: Minimum Value Fitness function evolution.

A. Discussion

The methodology introduced in this study exemplifies the adaptability principle in control system design. The proposed fitness function, which facilitates the prioritisation of distinct performance metrics, represents a novel approach to designing cost functions. This method streamlines multi-objective functions, simplifying their complexities. Conventional control functions typically focus on just one or a few performance metrics, limiting the designer's flexibility in tackling the diverse dynamic issues prevalent in real-world systems. By integrating parameters such as rise time, settling time, steady-state error, and energy expenditure our cost function offers a thorough assessment of system performance. This is especially valuable in complex systems where the trade-offs between various performance metrics can lead to vastly different control outcomes.

The inclusion of weighting factors adds a layer of versatility to the control design. For example, in scenarios where rapid response is critical, a higher weight can be assigned to settling time to prioritise its minimisation. Conversely, in applications where the effort in the actuator is more crucial, weights can be shifted to emphasise the reduction of control energy expenditure. It is possible to strike a balance between achieving desired system performance and ensuring minimal control expenditure. The ability to fine-tune and minimise control effort through appropriate parameter selection is essential for sustainable and efficient system operation.

A substantial 36.36% reduction in the energy consumed by the controller plays a pivotal role in various contexts. This decrease not only enhances the energy efficiency of the control system but also contributes to resource savings and environmental sustainability. To illustrate its importance, one could consider battery-based systems or other systems under energy saturation. For instance, this energy savings can significantly boost the autonomy of electric vehicles and autonomous systems, becoming a critical advantage in an increasingly efficiency-conscious world focused on natural resource preservation.

This new way of tuning the controller paves the way for a more flexible, adaptable, and effective design methodology. It acknowledges the diverse aspects of system performance and furnishes the means to harmonise and enhance these elements in line with the specific requirements of each scenario.

V. CONCLUSIONS

In this study, we have successfully integrated GA with LQR control, demonstrating the efficacy of using evolutionary computation techniques in control system optimisation. Traditional LQR tuning methodologies, though effective, often employ a more rigid framework, which may not be sufficiently adaptable to a variety of complex system requirements. The adoption of GA, renowned for its global search capabilities, brings a novel dimension to this domain by efficiently navigating the vast parameter space to identify optimal LQR settings.

In this study, we showcased a versatile and adaptive methodology for tuning LQR parameters. There is potential in delving into advanced evolutionary algorithms or blending with other optimisation strategies to achieve even more nuanced control outcomes, as future work. Further refinement can be sought by redefining fitness functions and experimenting with alternative mutation processes. Instead of random variations in Q, a promising avenue might involve assessing the system's controllability matrix to pinpoint states that significantly influence the control of target outputs. This could provide a more structured and informed strategy for LQR tuning.

ACKNOWLEDGMENT

This publication has emanated from research conducted with the financial support of Science Foundation Ireland under Grant number 21/FFP-P/10065. Erivelton Nepomuceno was supported by Brazilian Research Agencies: CNPq/INERGE (Grant No. 465704/2014-0), CNPq (Grant No. 425509/2018-4) and FAPEMIG (Grant No. APQ-00870-17).

References

[1] E. Vinodh Kumar and J. Jerome, "Robust LQR controller design for stabilizing and trajectory tracking of inverted pendulum," *Procedia Engineering*, vol. 64, pp. 169–178, 2013.

[2] R. Banerjee, N. Dey, U. Mondal, and B. Hazra, "Stabilization of double link inverted pendulum using LQR," in *2018 International Conference on Current Trends towards Converging Technologies (ICCTCT)*, pp. 1–6, 2018.

[3] M. Balandat, W. Zhang, and A. Abate, "On infinite horizon switched LQR problems with state and control constraints," *Systems & Control Letters*, vol. 61, no. 4, pp. 464–471, 2012.

[4] S. Das, I. Pan, K. Halder, S. Das, and A. Gupta, "LQR based improved discrete PID controller design via optimum selection of weighting matrices using fractional order integral performance index," *Applied Mathematical Modelling*, vol. 37, no. 6, pp. 4253–4268, 2013.

[5] A. Sedehi and M. Mirjafari, "Robust optimal controller for buck converter under parametric uncertainty: An LQR approach," in *2022 13th Power Electronics, Drive Systems, and Technologies Conference (PEDSTC)*, pp. 101–106, 2022.

[6] K. Ogata, *Modern Control Engineering Fourth Edition*. Prentice-Hall, 2002.

[7] E. V. Kumar, J. Jerome, and K. Srikanth, "Algebraic approach for selecting the weighting matrices of linear quadratic regulator," in *2014 International Conference on Green Computing Communication and Electrical Engineering (ICGCCEE)*, pp. 1–6, 2014.

[8] M. Saif, "Optimal linear regulator pole-placement by weight selection," *International Journal of Control*, vol. 50, no. 1, pp. 399–414, 1989.

[9] Y. Yin and D. Li, "Selecting and simulation of weighting matrices for LQR control system with input delays," in *2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS)*, pp. 1054–1057, 2018.

[10] K. Miyamoto, J. She, D. Sato, and N. Yasuo, "Automatic determination of LQR weighting matrices for active structural control," *Engineering Structures*, vol. 174, pp. 308–321, 2018.

[11] Y. Li, J. Liu, and Y. Wang, "Design approach of weighting matrices for LQR based on multi-objective evolution algorithm," in *2008 International Conference on Information and Automation*, pp. 1188–1192, 2008.

[12] Ö. Oral, L. Çetin, and E. Uyar, "A novel method on selection of Q and R matrices in the theory of optimal control.," *International Journal of Systems Control*, vol. 1, no. 2, 2010.

[13] V. Kumar E and J. Jerome, "Algebraic Riccati equation based Q and R matrices selection algorithm for optimal LQR applied to tracking control of 3rd order magnetic levitation system," *Archives of Electrical Engineering*, vol. 65, no. 1, pp. 151–168, 2016.

[14] S. Trimpe, A. Millane, S. Doessegger, and R. D'Andrea, "A self-tuning LQR approach demonstrated on an inverted pendulum," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 11281–11287, 2014.

[15] M. K. Habib and S. A. Ayankoso, "Modeling and control of a double inverted pendulum using LQR with parameter optimization through GA and PSO," in *2020 21st International Conference on Research and Education in Mechatronics (REM)*, pp. 1–6, 2020.

[16] A. Ur Rehman, M. U. Khan, M. S. Shah, Q. Maqsood, M. F. Ullah, and S. Uba, "Aircraft pitch control based on genetic algorithm tuning with PID and LQR controller," in *2021 6th International Multi-Topic ICT Conference (IMTIC)*, pp. 1–6, 2021.

[17] S. Das, I. Pan, K. Halder, S. Das, and A. Gupta, "Optimum weight selection based on LQR formulation for the design of fractional order $\text{PI}^{\lambda}\text{D}^{\mu}$ controllers to handle a class of fractional order systems," in *2013 International Conference on Computer Communication and Informatics*, pp. 1–6, 2013.

[18] L. Guo, L. Wei, P. Ge, and Z. Qin, "Lane keeping controller based on LQR optimized by genetic algorithm," in *2022 6th CAA International Conference on Vehicular Control and Intelligence (CVCI)*, pp. 1–6, 2022.

[19] W. Yu, J. Li, J. Yuan, and X. Ji, "LQR controller design of active suspension based on genetic algorithm," in *2021 IEEE 5th Information Technology,Networking,Electronic and Automation Control Conference (ITNEC)*, vol. 5, pp. 1056–1060, 2021.

[20] H. Chen, Y. Yang, and J. Sun, "Improved genetic algorithm based optimal control for a flying inverted pendulum," in *2019 3rd International Conference on Electronic Information Technology and Computer Engineering (EITCE)*, pp. 1428–1432, 2019.

[21] M. B. Poodeh, S. Eshtehardiha, A. Kiyoumarsi, and M. Ataei, "Optimizing LQR and pole placement to control buck converter by genetic algorithm," in *2007 International Conference on Control, Automation and Systems*, pp. 2195–2200, 2007.

[22] A. Lu, Z. Lu, R. Li, and G. Tian, "Adaptive LQR path tracking control for 4WS electric vehicles based on genetic algorithm," in *2022 6th CAA International Conference on Vehicular Control and Intelligence (CVCI)*, pp. 1–6, 2022.

[23] N. Yang, D. Chang, M. Johnson-Roberson, and J. Sun, "Energy-Optimal Control for Autonomous Underwater Vehicles Using Economic Model Predictive Control," *IEEE Transactions on Control Systems Technology*, vol. 30, pp. 2377–2390, Nov. 2022.

[24] J. H. Holland, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. MIT press, 1992.

[25] S. L. Brunton and J. N. Kutz, *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*. Cambridge University Press, 2017.

[26] J. G. Da Silva, M. J. Lacerda, A. L. J. Bertolin, J. P. d. Santos, and E. Nepomuceno, "Lqr and genetic algorithms: An effective duo for assessing control expenditure and performance in dynamic systems," Jan 2024. Retrieved from osf.io/tvgu9.