# Robust Path Tracking of Autonomous Vehicle in Presence of Model Uncertainities via Model Based Linear Quadratic Gaussian (LQG) Control with Adaptive Q-Matrix

ZAEEMA WAJID [1,4], HASEEB UR REHMAN [1,2], NOMAN KHAN [3], IRFANA ALAM [2], DR ADIL ZULFIQAR[4], MUHAMMAD WALEED AFZAL[1,5], MUHAMMAD AMMAR AKRAM[*, 6,7], SYEDA LARAIB TARIQ [7,8]

[1] School of Computer Science and Software Engineering, Beaconhouse International College Faisalabad Pakistan.
[2] Department of Mechanical Engineering University of Engineering and Technology Taxila
[3] School of power engineering and engineering thermophysics Xian jiaotong university China
[4] National University of Computer and Emerging Sciences Faisalabad, Pakistan
[5] School of Computer Science and Electrical Engineering, NUST Islamabad
[6] Department of Mechanical Engineering, HITEC University Taxila
[7] U.S.-Pakistan Center for Advanced Studies in Energy (USPCAS-E), NUST Islamabad
[8] Mechanical Engineering Department, National University of Technology (NUTECH), Islamabad

***Abstract***—From few decades, the engineers are facing troubles and challenges in achieving the robust behavior of autonomous vehicles. The main task here is to achieve the autonomous vehicles stability, smooth tracking of path and fast response. In order to accomplish the mentioned task, we work on Linear Quadratic Gaussian (LQG). We implement the LQG with adaptive Q-Matrix in MATLAB and Simulink and see that it exhibits the smooth tracking, minimizes the overshoot and settling time. The LQG also rejects the noise or disturbance and tracks path smoothly even with the uncertainties. The performance of LQG, PID, Fuzzy, MPC and LQR controllers are compared with the results of LQG, and it is concluded that the LQG perform better in case of tracking and rejecting the disturbance.

***Keywords—LQG, LQR, Kalman Filter, PID, MPC***

## I. Introduction

One of the leading causes of death and permanent disability worldwide is traffic injuries. According to the World Health Organization (WHO), 20-50 million people suffer non-fatal injuries globally every year, and 1.24 million people die on the roads. This scenario tends to be safer due to the positive impact of smart car technology. In this case, autonomous vehicles emerge from advances in robotics, sensing, embedded systems, machine perception, navigation, and other related fields.

The impact of autonomous vehicles on current road traffic system dynamics is the subject of much speculation. Self-driving cars are believed to be able to expand on-demand mobility systems by allowing them to move autonomously between high-demand areas. This will help solve issues such as transportation, space usage, pollution and energy consumption. Autonomous vehicles are expected to bring significant benefits, which is why many stakeholders, primarily those in the automotive industry, are interested in seeing their development. Over the next five years, the auto industry will be racing to produce self-driving cars. Therefore, autonomous vehicles are expected to operate in current road transport systems (RTS) between 2020 and 2025. Concerns have been raised about the security of these systems.

In order to meet people's growing demand for convenient and safe vehicles, autonomous driving technology is developing rapidly. Some basic autonomous driving technologies, such as advanced driver assistance systems (ADAS), GPS waypoints tracking, Model Predictive Control, Linear Quadratic Control, PID, Adaptive PID, Fuzzy logic and many more. But, all of these have their own limitations, difficulties, complexity and tuning concerns.

Proportional Integral derivative (PID) controller is used extensively because of its simplicity, independence of process, cost effectiveness, provides good stability if tuned properly and have good response to disturbances. Apart from the advantages the PID controller has some limitations which cannot be oversee. The PID controller cannot dynamically compensate for unknown disturbances. In addition, there are conflicting parameter configurations between overshoot control and response speed. PID controllers are known to respond poorly to slow perturbations or ramp-type inputs. However, PID also show bad performance in case of controlling the integrating control and time delays [1,2].

Fuzzy Logic Controller is cheaper than other controllers, it has better performance capability and is more robust than PID. Fuzzy logic allows autonomous vehicle decision making with estimated values under incomplete and uncertain information. Previously, computer-based autonomous driving models integrated engine concepts through the use of fuzzy logic. Notably, the self-driving car can identify possible collision scenarios, calculate and evaluate the degree of risk involved in execution at runtime, and mitigate these risks by slowing or stopping the self-driving car. The use of fuzzy logic is not without its limitations. Because fuzzy logic is not always precise and produces subjective results, it may not be widely accepted. Fuzzy refers to something that is imprecise or unclear, so extensive hardware testing is required to verify and validate systems based on fuzzy knowledge. Defining membership functions and precise, fuzzy rules is challenging [3-6].

The quadratic cost function is the basis of linear quadratic controllers (LQR), which are widely used in motion and process control. LQC is used to control the yaw angle and lateral position stability of autonomous vehicles during various control operations. The main benefit of LQR is that complete status feedback produces an ideal input signal. However, LQR has disadvantages in systems where actuator limitations must be taken into account, such as when limits are controlled or manipulating variables is challenging [7,8].

One of the most widely used optimal control strategies for constrained linear or nonlinear systems is the model predictive controller (MPC). MPC optimizes the performance of process control by predicting future values using mathematical dynamic process models. MPC is a promising candidate for controlling systems. It achieves the dynamics of system model and predict the future system progress. It then, get the optimal solution by selecting the first element as current time control law. MPC is a more valuable tool than LQR when multivariable and constrained systems are involved. To achieve the goal of two tracking outputs with a single control signal, the controller must make trade-offs. Therefore, MPC is more helpful than LQR when applied to constrained multivariable systems. Apart from its advantages over LQR the MPC still has an important limitation is its computational burden, which can be a problem when the optimization is carried out online in real time. To consider uncertainties, make the problem very difficult to resolve. Usually it implies a high computational cost. It's necessary to have an accurate dynamic model. It also does not grantee the stability [9-11].

The above-mentioned [controllers have some limitations and they do not guarantee smooth path tracking and stability for the safety of a person.

In section II we will discuss the literature review. The section III will contain the methodology of work and section IV will be of MATLAB Simulation of work. Section V will be of results and discussions. At the end section VI will be of conclusion.

## II. Literature Review

Rabii Fessi et al. [12] presented the LQG controller for a quad rotor UAV based on particle swarm optimization. He used an established dynamic model and design the LQG controller for the stabilization of position and dynamics of quad rotor. The results were good in case of settling and rise time but the overshooting problem was at max compare to other controllers.

Djurović, Ž. M. et al. [13] proposed a sequential LQG approach for nonlinear tracking problems. The suggested method is demonstrated through the aircraft application. The major issue with the method is of its sensitivity to perturbation of nominal trajectory and this is due to its nonlinear property which is hard to tackle.

David,Stephen, Marc et al. [14] analyzed the LQG-Obstacle for collision avoidance. The simulation of work results in safe and smooth flight but the need of adding the obstacles geometry explicitly and selecting the linearization point in advance in case of car like system is challenging.

Alexander and Jordan et al. [15] has shown the comparison of LQG and MPC for steering control systems. The results of their work concluded that the MPC is a high maintenance and computational controller but LQG is compact and less computationally intensive, shows better stability.

Lupian, L. F et al. [16] proposed an LQG control design for position and trajectory tracking of ball-board systems in their paper. [16] MATLAB simulation results prove the adaptability, robustness and good control performance of the LQG controller.

Tornero, J. et al. [17] studied that the stabilization of wheeled inverted pendulum using LQG optimal controller is presented. This shows a good stability performance but the experiment is held on the flat surface which means an ideal environment free of disturbances which is not the case we see in real life.

Zexi Liu et al. [18] showed that the MultiMate LQG controller for path tracking. According to this approach it is possible to control and estimate states when sensor noise is injected. But for that it is must for the sensor to obey the real-time constraints.

Tan, J. [19] presented the design and simulation of LQG controller for mobile cart. The system presented in both ways open-loop and close-loop. The system is unstable when run in open loop and the stability could be achieved through variance in close-loop status.

Jiazheng and Weijie et al. [20] proposed the tracking of square reference signals using model based LQG robust Controller. The study shows the improved results in position tacking and noise rejection. In spite of those benefits the presence of discontinuity in control law introduces the chattering and can be removed by $H\infty$ but the implementation of $H\infty$ is not easy.

Van den Berg et al. [21] studied LQG-MP path planning for robots with motion uncertainty is presented. The work results in precise path planning but the authors did not use a priori distributions and covariance in experiment between the states that led the system to unable to indicate the new distributions which dropdown the performance below threshold.

The work done by Jur van den Berg et al. [22] showed that the LQG based planning sensing and control of steerable needles. This approach minimizes the probability of intersecting the needle into the bone. In this paper the LQR does not bound the control inputs which is a drawback when the needle curvature exceeds the attainable limit.

Stephen D.Patek et al. [23] proposed the LQG control of a two-wheeled mobile pendulum system. The presented system ensures the stabilization of pendulum and fix the unstable equilibrium. The LQG strategies give the satisfied results.

Maidul Islam et al. [24] presented the LQG based close loop control of type 1 diabetes. The work is used to investigate the blood glucose regulation problem because it continuously monitors the glucose level and inject insulin by measuring the LQR gain because it prevents the deviation from target. But this automated system is risky because any kind of failure led the person to risk his/her life.

Patra, A. K et al. [25] work presented the comparison of performance of LQR and LQG on quad rotor. LQR does not offer estimation and integral part so it is unable to track noise and disturbance but its response is fast. LQG has the high controllability.

Dixit, S. et al. [26] studied the Back stepping LQG controller for the stabilization and trajectory tracking. The authors perform simulation and produce results in presence of disturbance, so it turned out that BLQGC outperforms PID, H∞ and LQR in case of robustness and disturbance rejection.

After detailed literature review, it is observed that the latest technique and controller which is Linear Quadratic Gaussian Control (LQG) with adaptive Q-Matrix that can grantee human safety and comfort by handling the noise and errors and sure the stability of autonomous vehicle at various speed by doing the automated tuning which is eliminated factor in previous work done.

## III. METHDOLOGY

We have designed the LQG controller for zone tracking and for that we use LQG model and is formed with the combination of LQR and Kalman observer. The model gives better performance in case of stability, error elimination and path tracking. It takes the dynamics matrices of plant or vehicle and give results after performing optimization. In optimization the constraints of LQR and Kalman observer continuously update until they approached at fully optimized solution. The task is achieved in several steps:

- *LQG Model.*
- *Select the Dynamics of Path and Plant or Vehicle.*
- *Design LQR.*
- *Design Kalman Observer.*
- *Check response with different Q values.*
- *Check Plant or Vehicle response without LQG controller.*
- *Check Plant response with LQG controller in the presence of uncertainties.*
- *Check the fully optimized response of Plant or Vehicle.*

After getting results by following these steps we compare the performance of LQG with the results of other research works on different controllers and get to know that it gives us good response compare to other controllers.

### A. Linear Quadratic Gaussian Model

The architecture of LQG for path tracking is divided into two parts one part is of plant and other is of LQG controller. The right side of model is plant which consists of vehicle dynamics in form of matrices and are also subjected to outer disturbances, the sensor data which is subjected to noise and a path generator. The left side is simply a LQG controller containing two blocks one is Kalman observer and the other is LQR as shown is Fig 1. The information of current vehicle dynamics and path is sent to the kalman observer which estimates the best solution to the problem in the presence of disturbances and uncertainties and LQR make the ride more stable and comfortable. Then the best updated values or data is sent to vehicle dynamics and again the same loop happens several times until we get the desired response or path is fully tracked [27].

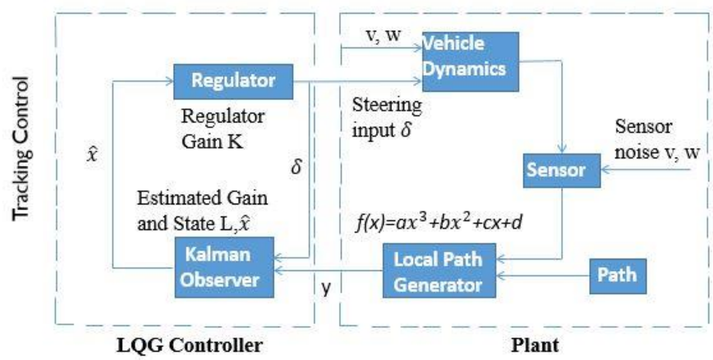


*Figure.1: Linear Quadratic Gaussian Control architecture.*

### B. Plant and Path Dynamics

After selecting the model, we need to know the plant and path dynamics on which our proposed controller works and find the best solution to track the path faster and smoother. The discrete path tracking model is given as Eq. (1) and the state space of model is given is Eq. (2).

$$X_t = AX_{t-1} + Bu_{t-1} + v \quad (1)$$
$$Y_t = CX_t + w$$

where A, B and C are the system matrices consisting of the moment of inertia I, the competitor's stiffness C and the speed change V. The control input is represented by u, the output state variable is represented by Y, and the input state variable is X. These are Gaussian noise w and v. The matrix given is shown in Eq. (2).

$$A=\begin{bmatrix}1 & dt & 0 & 0\\ 0 & 1-\frac{2dt(C_{af}+C_{ar})}{mV_x} & \frac{2dt(C_{af}+C_{ar})}{m} & \frac{2dt(-C_{af}\,I_f+C_{ar}\,I_r)}{mV_x}\\ 0 & 0 & 1 & dt\\ 0 & \frac{2dt(C_{af}\,I_f-C_{ar}\,I_r)}{I_zV_x} & \frac{2dt(C_{af}\,I_f-C_{ar}\,I_r)}{I_z} & 1-\frac{2dt(C_{af}\,I_f^2-C_{ar}\,I_r^2)}{I_zV_x}\end{bmatrix}$$

$$B=\begin{bmatrix}0\\ \frac{2dt(C_{af})}{m}\\ 0\\ \frac{2dt(C_{af}\,I_f)}{I_z}\end{bmatrix} \qquad C=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix} \qquad (2)$$

Eq. (1) and Eq. (2) are path tracking dynamics and Table. 1 is showing the test Plant or Vehicle parameters.

*TABLE 1: Plant or Vehicle Parameters.*

| **Parameters** | **Values** |
|---|---|
| Speed $V_x$ | $5\ m/s$ |
| Mass $m$ | $1490kg$ |
| Moment of Inertia $I_z$ | $2600kgm^2$ |
| Center of Gravity Point $I_f$,$I_r$ | $1.1m$ , $1.6m$ |
| Cornering Stiffness $C_{af}$,$C_{ar}$ | $Both$ $53000N/rad$ |

### C. Linear Quadratic Regulator

The model based LQG controller does not need tuning and can be applied as it is even in case of continuous variations. The LQR and Kalman observer solve the tracking and noise problems.

The linear system is shown in Eq. (1) is the basis of LQR from where we could know the LQR control input and can be expressed as Eq. (3) [27].

$$u_{t-1} = -K_{LQR}X_{t-1} \qquad (3)$$

There are two main gains Q is the state gain and R is the control gain that are need to be set properly from the sake of reduced error or cost function as shown is Eq. (4). In the proposed work we use the adaptive Q so that it can update itself at every iteration.

$$J = \sum_{t=0}^{\infty} X_t^T Q X_t + u_t^T R u_t \qquad (4)$$

The regulator gain $K_{LQR}$ can be found through Eq. (5) using Discrete Algebraic Riccati Equation (DARE) and that is the most important gain in designing LQR.

$$K_{LQR}=(B^T\ P_t\,\mathrm{B}+\mathrm{R})^{-1}+B^T P_t\,\mathrm{A} \qquad (5)$$

### D.Kalman Observer

State observer estimates the true state of system on which it gives the optimized solution. State observer combines the system's behavior and external measurements to achieve optimal estimation. Kalman observer is the righteous observer for linear systems that uses the state space model of system and sensor data to estimate the system's state.

The linear system is shown in Eq. (1) can be perceived as a Kalman observer in Eq. (6) [27].

$$\hat{X}_t=A\hat{X}_{t-1}+Bu_{t-1}+L\ (y_t\text{-}\hat{y}_t)$$
$$\hat{Y}_t=\mathrm{C}\hat{X}_{t-1} \qquad (6)$$

The $\hat{X}_t$ and $\hat{Y}_t$ are the estimated state and output means the Kalman observer update the plant or vehicle dynamics. The L is the Kalman observer gain matrix. That L can be found using the DARE and calculated through Eq. (7).

$$L = \sum_t C^T(\sum_t CC^T + W)^{-1} \qquad (7)$$

The gain L played the most significant role in finding the optimal solution as the gain K of LQR is most important in reducing the error.

## IV. Matlab Simulation

The LQG model is shown in Fig 2.

The model contains several blocks:

- Reference step block generates a step input and fed the input to Feed forward gain which is used to reject the persistent disturbances. Applying feedforward control can significantly improve the performance of the control system.
- The sum of LQR gain and Feed forward gain is fed to the vehicle dynamics block which contains the system matrices A B C and D and we can also incorporate the values of v and w in this block.
- The output constraints block shows the output of plant dynamics on the desired path.
- Scope block outputs the plant response with and without controller action.
- Sensor noise block introduces the noise to the plant and the data of plant with and without noise is first combined with mux and then sent to Kalman state estimator block.
- The Kalman state estimator block works by leveraging a dynamic model of the system, multiple continuous measurements (e.g. from sensors), and known control inputs to create an estimate of the amount of change in the system that is better than the estimate produced using a single measurement. system.
- In parallel the integral action is used to achieve steady state, to remove the deviations and noise effect.
- The outputs of Kalman state estimator and integral action are fed to the mux and then send to LQR gain block.
- Linear quadratic regulator (LQR) gain blocks are a well-known technology that provide optimally controlled feedback gain to support closed-loop stability and high-performance system designs.

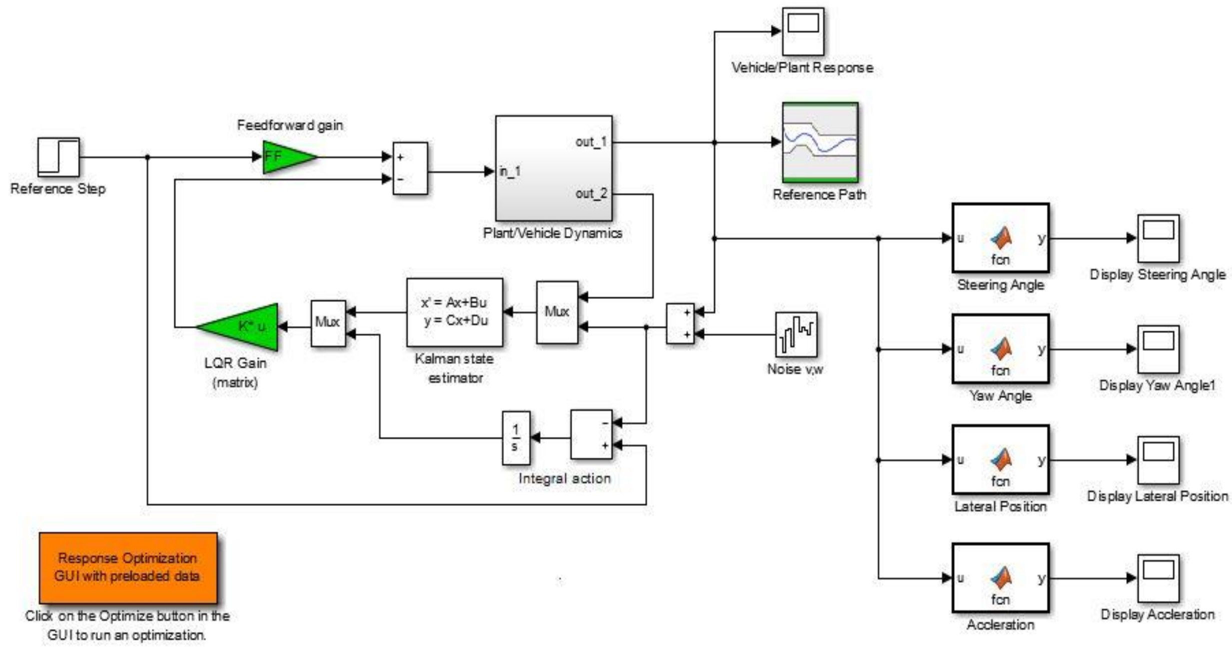


*Figure.2: Linear Quadratic Gaussian (LQG) Control Model.*

- The updated values of LQR gain and Feed forward gain is first summed up and then sent to plant dynamics block and the states of system are updated accordingly.
- Response optimization GUI block does the process in Iterative manner to find the optimal response.
- In this process the functions of steering angle, yaw angle, lateral position and acceleration updated themselves accordingly and show the results of dynamics to follow the desired path.

## IV. Results and Discussions

From LQG model the results are divided into four sections. At first the impact of Q-matrix is shown. Then the response of model when there is no control action, in between the iterations and when it is fully optimized. It can also be seen that how the different parameters of vehicle will according to those responses.

### A. Response of Q-matrix

The Q-matrix is used to eliminate the manual tuning of parameters at every iteration. So first the value of R is set to 1 which is sign of considering the dynamics of the system to be accurate and also set the value of Q to 1 and see the impact of it on position of the vehicle. The response with Q=1 is shown in Figure 3.

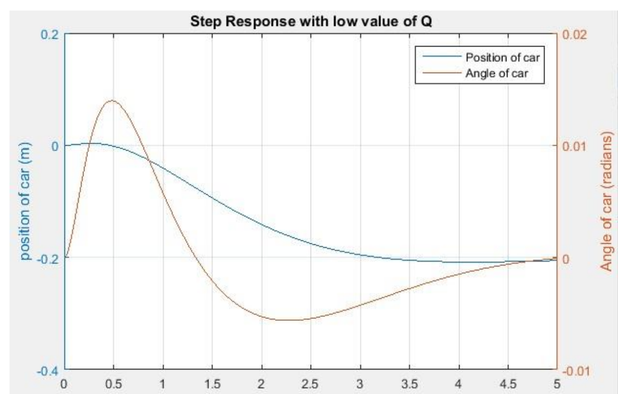


*Figure 3: Response of LQR with Q=1.*

In Figure 3 the red curve is the angle of the car and blue is the position of the car. The graph shows that both the position and angle curves are overshooting fine for Q=1. But both angle and position are not getting to their steady state. They are have a disturbed settling time. So the desire response is not achieved. To eliminate the error from low value of Q it is need to implement LQR control with High value of Q where it gives us the satisfied response. The response with high value of Q is shown in Figure 4.

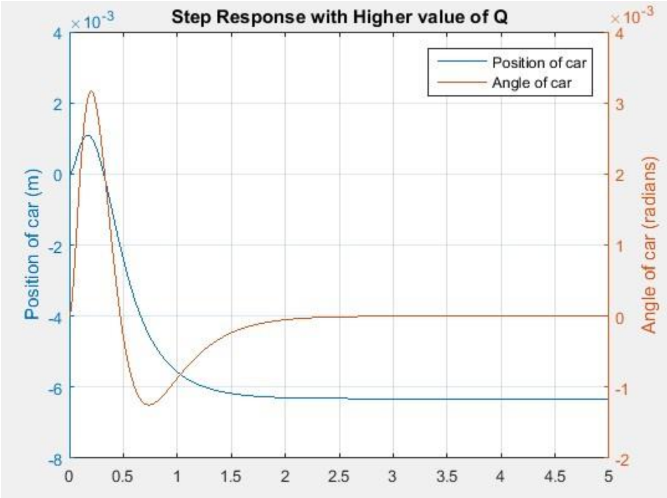


*Figure 4: Response of LQR with Q>>1.*

Figure 4. shows that with high value of Q the both angle and position first overshoot and then achieve the steady state. The overshoot value gets reduced and the setting time error is also compensated by the Q>>1. This improves the response of the system. The tracking error also gets smaller with the help of higher Q value. But still we have to bound the value to Q not to exceed the limit and become a drawback for the system instead of benefit, so we use value from 1 to 100 and in iterations it continuously updates until the optimal response achieved.

*B. Path tracking with and without Control Action and no uncertainties*

The response of the overall model without the activation of the controller LQG and without adding uncertainty is shown in Figure 5.

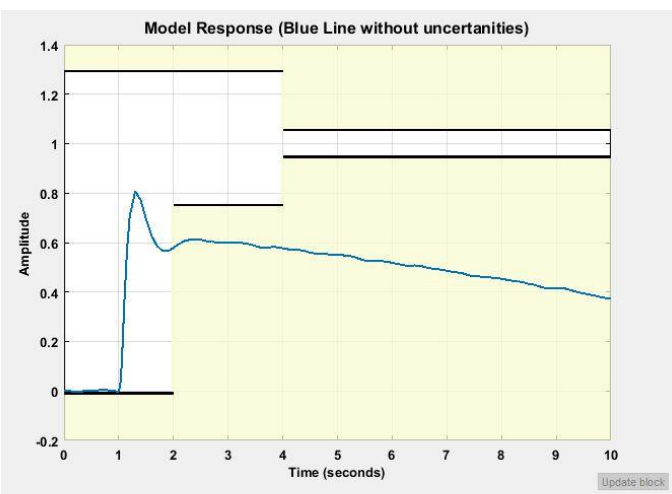


*Figure 5: Response of Plant/Vehicle with no Controller and uncertainties.*

Figure 5 shows the reference path that is needed to be followed. The is showing with the black lines and they are fixed at some bounds means first upper bound is set at 1.3 amplitude and 3s and the second upper bound is set at zero amplitude and 2s, the thirst lower bound is in the middle to narrow down the reference path and get smooth tracking so its magnitude is 0.75 unit and the settling time is 4s the final bounds are starting from 4s from where we need to have a steady response of our model with the amplitude equal to 1 as a reference step that we gave as a step reference input. That reference step should be followed to get the response between the required paths. The reference step is shown in Figure 6 and its amplitude is 1 unit and then extend by feed forward to have a good range. Without the controller action the response of model is same as plant because of course we are implementing the model on the plant. It first overshoots little bit because of the step reference but there was no controller present and optimization action to force the model to follow the path. So, that's why it is decaying and not coming back on path and give the disturbed response that is not required.

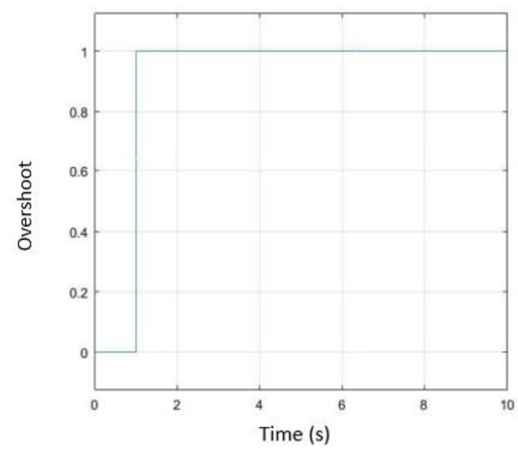


*Figure 6: Reference Step.*

Then apply the controller action when there are no uncertainties and noise. The model follows the exact reference step response as shown in Figure 7.

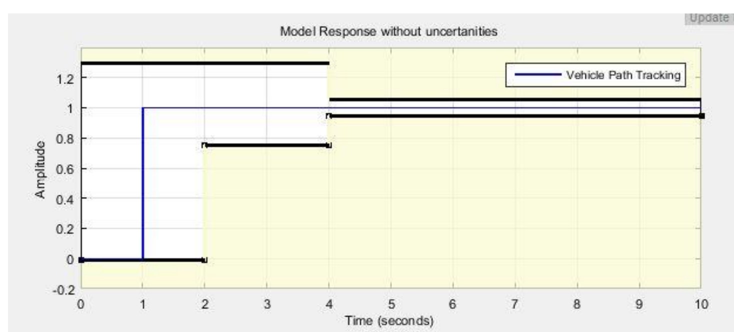


*Figure 7: Path Tracking with Controller and no uncertainties.*

### C. Path tracking with and without Control Action and with uncertainties

#### i. At Start of Optimization

The response of model without the activation of LQG controller or with no sufficient values provided to the system to follow the path. Along with that the two uncertainties are added a little up from the original plant response and little down to the original pant response and we add them to the model response as replica of plant response and see that they are also not recognizing the path because of the elimination of controller LQG action. The response is shown in Figure 8. I which the dark blue line is the model response without uncertainty and the black dashed lines are showing the model uncertainties. The solid black lines are showing the path that the plant of vehicle should follow when implement the LQG controller. The noise is added to the plant dynamics also and see the response of LQG controller in presence of uncertainties and noise. The noise added shown in Figure 9.

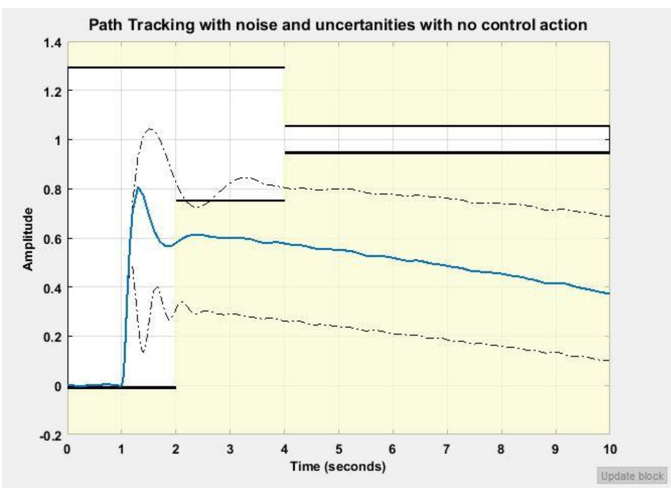


*Figure 8: Path Tracking without Controller and with uncertainties.*

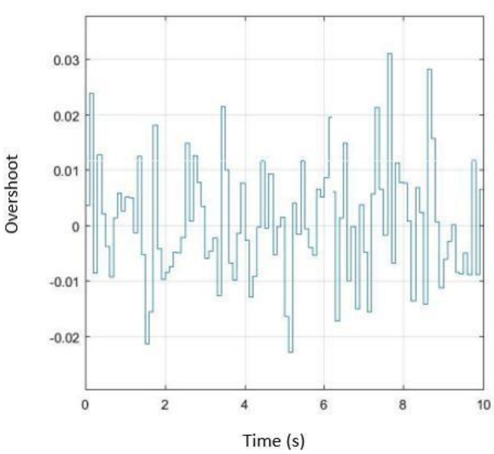


*Figure 9: Noise v and w.*

#### ii. In the Middle of Optimization

After seeing the initial unsettled responses of plant and model, LQG controller was implemented and run an optimizer for getting the desired response means to get the path on track. The plant response in the middle of iterations with the activation of LQG controller is shown in Figure 10.

The optimizer run iteratively to continually update the values of LQR gain and Kalman estimator parameters. To observe the behavior of model we take a look at the response of model in the middle of iterations and see that it is not fully optimized to give the desired result and does not follow the path completely instead it is deviating from the path and still try to get in the path with further iterations. The iterations are shown in Table 2.

*TABLE 2: Optimization progress table in the middle of iterations.*

Optimization Progress Report

| Iteration | F-count | Output Constraint (Upper) (<=0) | Output Constraint (Lower) (>=0) |
|---|---|---|---|
| 0 | 13 | -0.1922 | -0.8926 |
| 1 | 26 | -0.0592 | -0.4918 |
| 2 | 40 | 0.1625 | -0.3928 |
| 3 | 68 | 0.1519 | -0.3909 |
| 4 | 81 | 0.0188 | -0.2274 |
| 5 | 94 | 0.2201 | -0.0732 |
| 6 | 108 | 0.1839 | -0.0153 |

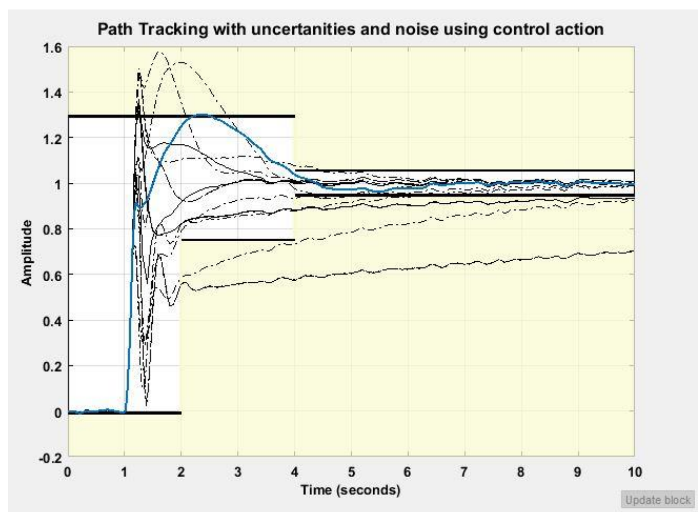


*Figure 10: Path Tracking with Controller and with uncertainties in the middle of iterations.*

#### iii. After Full optimization

The optimizer runs iteratively and stops when the vehicle or plant follows the path exactly as required and reaches the optimized value of LQR gain and Kalman estimator parameters and get the fully stale response within desired range. In the middle of iterations, the model does not follow the path completely and possess fluctuating behavior at start and then try to cop up with this but the response does not satisfy the requirement at that point but when fully optimized it looks great and followed the path perfectly as shown in Figure 11. The required response was recorded for 14 iterations. The number of iterations at with the response get stables is shown in Table 3.

*TABLE 3: Optimization progress table after full iterations.*

Optimization Progress Report

| Iteration | F-count | Output Constraint (Upper) | Output Constraint (Lower) |
|---|---|---|---|
| 0 | 13 | -0.1922 | -0.8926 |
| 1 | 26 | -0.0592 | -0.4918 |
| 2 | 40 | 0.1625 | -0.3928 |
| 3 | 68 | 0.1519 | -0.3909 |
| 4 | 81 | 0.0188 | -0.2274 |
| 5 | 94 | 0.2201 | -0.0732 |
| 6 | 108 | 0.1839 | -0.0153 |
| 7 | 124 | 0.0853 | -0.0174 |
| 8 | 137 | 0.0219 | -0.0517 |
| 9 | 153 | 0.0275 | -0.0485 |
| 10 | 168 | 0.0696 | -0.0234 |
| 11 | 181 | 0.0124 | -0.0086 |
| 12 | 194 | -0.0013 | -8.0874e-04 |
| 13 | 207 | 1.7295e-04 | 2.6843e-06 |
| 14 | 220 | 7.0205e-06 | -7.3331e-08 |

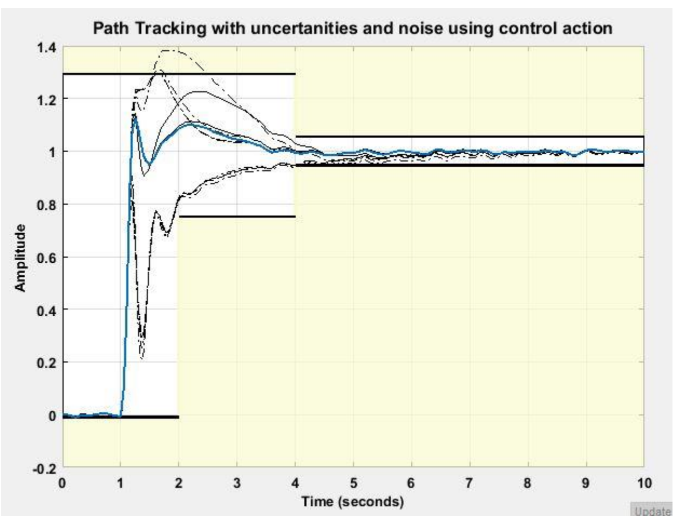


*Figure 11: Path Tracking with Controller and with uncertainties Fully optimized*

At start the model try to fit in the path and rises the overshoot the same is with uncertainties. Because of the noise the model gets the harder time to fit in that's why at start the model gets out of the bound in the presence of noise but then the values and parameters of LQG gets refined and get the exact response, the model gets settles down at 4s and the overshoot first is more than 1 unit and then it gets in line and lower than 1 unit.

It is also seen that when path was narrowed down, the path is still followed, and the vehicle response is not coming out of the way even in the presence of uncertainties as shown in Figure 12.

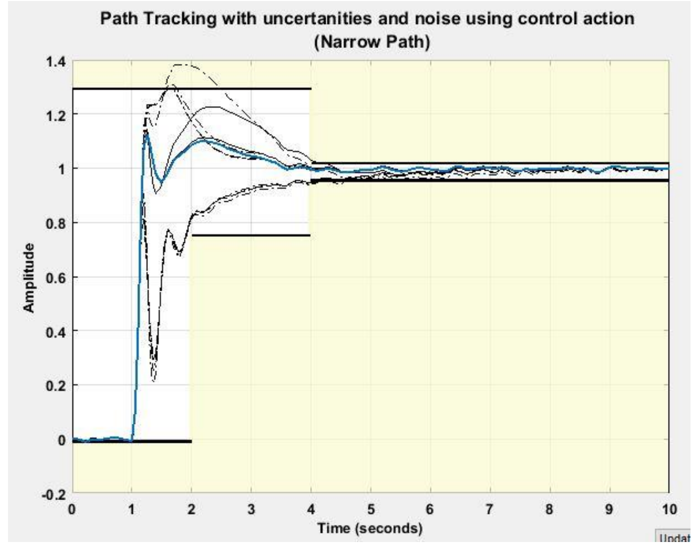


*Figure 12: Response of Model with Narrow path.*

The change of steering angle, yaw angle, lateral position and acceleration are shown is table 4**.**

*TABLE 4: Changing Parameters of Vehicle.*

| Output Parameters | At the Start | In the Middle | At the End |
|---|---|---|---|
| Steering Angle (degree) | $-30^o$ | $17^o$ | $5^o$ |
| Yaw Angle (degree) | $41^o$ | $12^o$ | $0^o$ |
| Lateral Position (meter) | $-28.45$ | $-4.07$ | 0 |
| Acceleration (meter/$s^2$) | 8.3125 | 4.15 | 0.2 |

## V. CONCLUSION

In control systems, the stability has the essential part to play and always hard to achieve especially when there is a noise or any kind of disturbance. The proposed controller LQG is implemented in this work to get that stability in less time and with less overshoot. We have implemented the LQR along with the Kalman estimator to get the benefits of both. The tracking dynamics and vehicle dynamics are used to design the LQR and Kalman estimator. The response is observed in three stages: 1) Response on initial stage. 2) Response in the middle of optimization. 3) Response at the end of optimization. The final response gave us perfect path tracking on the desired path. We also add noise to our system and see that even in the presence of noise the LQG controller has gave perfect tracking performance and reject disturbance seamlessly.

LQG has given better stability with less settling time, less overshoot, improved path tracking and achieved steady state in the required time. So, it is established that the LQG is best among other controllers when it comes to stability.

After seeing the good response of LQG controller on one path, in future it is suggested to implement the LQG controller for multiple paths tracking and to implement the LQR-MPC to compare the performance of both to see which performs better in terms of stability, when subjected to noise.

## References

[1] Farag, W. (2019). Complex Trajectory Tracking Using PID Control for Autonomousn Driving. *International Journal of Intelligent Transportation Systems Research. doi:10.1007/s13177-019-00204- 2*

[2] Zaeema, Fizza,. PID Tuning with reference tracking and plant uncertainty along with disturbance rejection, *Systems Science & Control Engineering An open Access Journal Taylor and Fransis*,9(1),160-160.

[3] Nguyen, A.-T., Rath, J., Guerra, T.-M., Palhares, R., & Zhang, H. (2020). Robust Set-Invariance Based Fuzzy Output Tracking Control for Vehicle Autonomous Driving Under Uncertain Lateral Forces and Steering Constraints. *IEEE Transactions on Intelligent Transportation Systems, 1–12. doi:10.1109/tits.2020.3021292*

[4] Hafiz Halin, Wan Khairunizam, Path tracking Simulation of Buggy Car by Fuzzy information of Steering Wheel, *SympoSIMM* 2019, LNME, pp. 57–66, 2020.

[5] Carlos Andre,Fabio,Marconi, Hierarchical Fuzzy Control, *INTECH Open source*, 2012.

[6] Muhammad Bilal Kadri, Disturbance rejection using Fuzzy model free adaptive control (FMFAC) with adaptive conditional defuzzification Threshold, *Journal of the Franklin Institute*.(2014),http://dx.doi.org/10.1016/j.jfranklin.2014.02.003

[7] Yakub, F., & Mori, Y. (2015). Comparative study of autonomous path-following vehicle control via model predictive control and linear quadratic control. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering, 229(12), 1695–1714.* doi:10.1177/0954407014566031.

[8] Piotr Lichota, Franciszek Dul, System Identification and LQR Controller Design with incomplete state observation for Aircraft Trajectory tracking, *Energies*, 2020, 13, 5354; doi:10.3390/en13205354.

[9] Cui, Q., Ding, R., Zhou, B., & Wu, X. (2017). Path-tracking of an autonomous vehicle via model predictive control and nonlinear filtering. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering,232(9),1237–1252.* doi:10.1177/0954407017728199.

[10] Yakub, F., Abu, A., Sarip, S., & Mori, Y. (2016). Study of Model Predictive Control for Path-Following Autonomous Ground Vehicle Control under Crosswind Effect. *Journal of Control Science and Engineering, 2016, 1–18.* doi:10.1155/2016/6752671.

[11] D.Limon, Teodoro Almao,. Robust control of the distributed solar collector field ACUREX using MPC for tracking, *The international Federation of Automatic Control*,2018, DPI2007-66718-C04-01.

[12] Fessi, R., & Bouallègue, S. (2019). LQG controller design for a quadrotor UAV based on particle swarm optimisation. *International Journal of Automation and Control*, 13(5), 569. doi:10.1504/ijaac.2019.101910.

[13] Djurović, Ž. M., & Kovačević, B. D.(2008). *A sequential LQG approach to nonlinear tracking problem. International Journal of Systems Science, 39(4), 371–382.* doi:10.1080/00207720701779989

[14] David,Stephen, Marc,. LQG-Obstacles: Feedback Control with Collision Avoidance for MobileRobots with Motion and Sensing Uncertainties, *2012 IEEE International Conference on Robotics and Automation.* 978-1-4673-1405-3/12.

[15] Alexander and Jordan, Comparison of model predictive control (MPC) and Linear Quadratic Gaussian (LQG) Algorithm for Electrohyraulic Coantrol System, *25th Scientific Conference on Power Engineering and Power Machines (PEPM'2020)*, Volume 207, 2020.

[16] Lupian, L. F. & Avila, R. (2008). Stabilization of a Wheeled Inverted Pendulum by a Continuous-Time Infinite-Horizon LQG Optimal Controller. *2008 IEEE Latin American Robotic Symposium*. doi:10.1109/lars.2008.33.

[17] Tornero, J., Piza, R., Albertos, P., & Salt, J. (n.d.). Multirate LQG controller applied to self- location and path-tracking in mobile robots. *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the Next Millennium(Cat.No.01CH37180).* doi:10.1109/iros.2001.976239.

[18] Zexi Liu, Design and Simulation of a LQG Optimal Controller for a Mobile Cart, *Department of Electrical & Computer Engineering, Temple University 1801 N. Broad Street, Philadelphia, PA 19122.*

[19] Tan, J., Sun, W., & Yeow, J. (2018). Tracking of square reference signals using internal model- based LQG robust controller for positioning of a micro-electro-mechanical systems micro mirror. *Micro & Nano Letters, 13(5), 704–708.* doi:10.1049/mnl.2017.0681.

[20] Jur van den Berg, Pieter Abeel, LQG-MP: optimized Path Planning for Robots with Motion Uncertainty and Imperfect Tate Information. *NIH Award 1R01EB-006435-01A1*.

[21] Van den Berg, J., Patil, S., Alterovitz, R., Abbeel, P., & Goldberg, K. (2010). LQG-Based Planning, Sensing, and Control of Steerable Needles. *Algorithmic Foundations of Robotics IX, 373–389.* doi:10.1007/978-3-642-17452-0_22.

[22] Ervin Burkus, Peter Odry, LQG Control of a Two-Wheeled Mobile Pendulum System, *NTELLI 2015:The Fourth International Conference on Intelligent Systems and Applications*, BN: 978-1-61208-437-4.

[23] Stephen D.Patek, Mare D, Linear Quadratic Gaussian-Based Close loop Control of Type 1 Diabetes, *Journal of Diabetes Science and Technology* Volume 1, Issue 6, November 2007.

[24] Maidul Islam, Mohamad Okasha, Performance evaluation of linear quadratic regulator and linear quadratic gaussian controllers on quadrotor platform, *International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-7, Issue-6S, March 2019.*

[25] Patra, A. K. (2020). Backstepping LQG controller design for stabilizing and trajectory tracking of vehicle suspension system. *SN Applied Sciences, 2(2).* doi:10.1007/s42452-020-1945-7.

[26] Dixit, S., Fallah, S., Montanaro, U., Dianati, M., Stevens, A., Mccullough, F., & Mouzakitis, A. (2018). Trajectory planning and tracking for autonomous overtaking: *State-of-the-art and future prospects. Annual Reviews in Control, 45,76–86.* doi:10.1016/j.arcontrol.2018.02.001.

[27] Lee, K., Jeon, S., Kim, H., & Kum, D. (2019). Optimal Path Tracking Control of Autonomous Vehicle: Adaptive Full-State Linear Quadratic Gaussian (LQG) Control. IEEE Access,7, 109120–109133. doi:10.1109/access.2019.2933895.

## About Authors:

**First Author/ Corresponding Author** – Zaeema Wajid, MSc Electrical Engineering, Lecturer at School of Computer Science and Software Engineering, Beaconhouse International College Faisalabad Pakistan.

**Second Author** – Haseeb Ur Rehman, MSc Mechanical Engineering, Lecturer at School of Computer Science and Software Engineering, Beaconhouse International College Faisalabad Pakistan.

**Third Author** – Noman Khan, PhD Scholar at School of power engineering and engineering thermophysics Xian jiaotong university China.

**Forth Author** – Irfana Alam, BSc Mechanical Engineering Student at Department of Mechanical Engineering University of Engineering and Technology Taxila.

**Fifth Author** – Dr Adil Zulfiqar Assistant Professor at National University of Computer and Emerging Sciences Faisalabad, Pakistan.

**Sixth Author** – Muhammad Waleed Afzal, Lecturer at School of Computer Science and Software Engineering, Beaconhouse International College Faisalabad Pakistan.

**Seven Author/Corresponding Author** – Ammar Akram, MSc Mechanical Engineering, PhD Scholar at USPCASE NUST, Islamabad. Lecturer at Department of Mechanical Engineering, HITEC University Taxila.

**Eighth Author** – Syeda Laraib Tariq, MSc Mechanical Engineering, PhD Scholar at USPCASE NUST, Islamabad. Lecturer at Department of Mechanical Engineering, National University of Technology (NUTECH), Islamabad.