

Adaptable robotics for disaster response and search & rescue: Integration of deformable smart car design and PI control

Yuang Lu

School of Advanced Technology, Xi'an Jiaotong-Liverpool University, Suzhou, Jiangsu, China, 215123

Yuang.Lu21@student.xjtlu.edu.cn

Abstract. In the context of disaster response and search and rescue operations, the need for adaptable and efficient robotic systems has become increasingly evident. This research paper addresses the evolving challenges in this domain by introducing a novel DIY deformable robot equipped with advanced PI (Proportional-Integral) control. The background of this study emerges from the growing urgency to enhance the capabilities of robotics in post-disaster scenarios, where navigation through complex terrains and the swift delivery of supplies are paramount. The primary objective of this paper is to develop and assess a versatile robotic platform using a multidisciplinary approach encompassing mechanical engineering, electronics, and control systems. The core of the investigation is a DIY deformable car consisting of two sensor-equipped containers and a linking module, making it well-suited for search and rescue missions. This paper contains the implementation of a PI control system to govern the robot's mobility and adaptability. This includes a detailed examination and demonstration of the PI control mechanism, encompassing its proportional and integral components. The actual results also indicate that the smart car controlled by the PI controller has better performance in both stability and speed.

Keywords: Deformable Robot, PI Control, Search And Rescue, Speed Control, Smart Car.

1. Introduction

In the rapidly evolving field of robotics and autonomous mobility, the intersection of deformable smart car design and PI (Proportional-Integral) control has become a focal point of research and innovation [1]. This convergence is driven by the growing recognition of the critical role these technologies play in disaster response and search and rescue operations.

Recent advancements in deformable smart cars offer a ray of hope in addressing the complex terrains and confined spaces frequently encountered in disaster-stricken areas. These vehicles possess the intrinsic capacity to adapt to challenging environments, making them indispensable tools for mission-critical tasks [2]. However, despite their potential, the practical implementation and optimization of deformable smart cars remain intricate and under-explored.

Simultaneously, PI control has emerged as an indispensable component in the enhancement of smart car behavior, particularly when precise navigation is paramount [3]. The application of PI control mechanisms empowers these vehicles to navigate terrains with unparalleled precision and adaptability

[1]. Nevertheless, a comprehensive inquiry into the dynamic interplay between deformable smart car design and PI control integration remains an area ripe for exploration.

This paper aims to delve deeply into this burgeoning domain, seeking to unlock the synergies and potentials hidden within deformable smart car design and PI control. Our research takes a dual-pronged approach: first, to discern the optimal design principles that enable smart cars to expertly traverse intricate terrains, and second, to scrutinize how the integration of PI control mechanisms can elevate their adaptability, responsiveness, and overall stability.

In pursuit of these objectives, our research leverages advanced tools such as SolidWorks for the creation of precise smart car models. These models enable a profound understanding of the intricate physical attributes of these vehicles. Further, simulation platforms like Falstad and Octave are employed to meticulously evaluate the impact of PI control on smart car behavior within controlled virtual environments [4].

The significance of this research transcends its theoretical scope. It promises to inform the future landscape of disaster response and search and rescue operations by empowering practitioners with the knowledge and tools required to optimize deformable smart car performance. Ultimately, this exploration is poised to save lives, minimize damage, and enhance the efficacy of robotic systems in the face of disaster.

2. The mechanical structure of the smart car

The robot consists of two containers and a connecting module equipped with a control circuit. According to Figure 1, an adjacent axis to the connecting module enables it to shift between a "U" shape and an "I" shape, adapting to various terrains and operational requirements. These containers, positioned between the wheels, possess the capacity to transport essential supplies and are outfitted with sensors to fulfill diverse functions [2].



Figure 1. Rendering of 3D modeling. A: Rendering of the Deformable Smart Car in “U” shape; B: Rendering of the Deformable Smart Car in “I” shape.

2.1. Hub Configuration

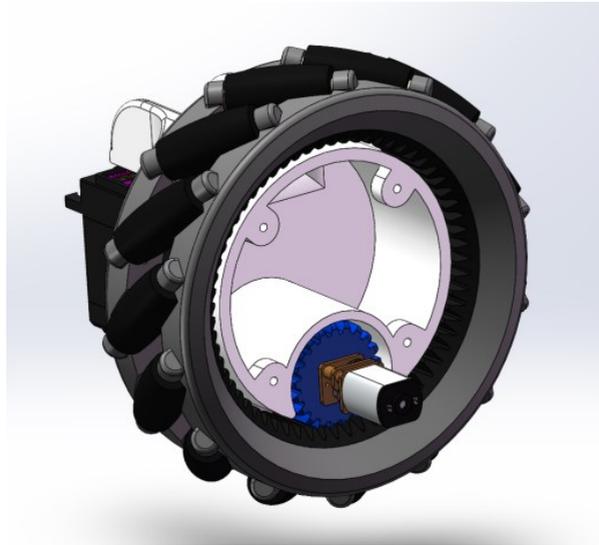


Figure 2. 3D modeling for hub configuration.

The primary rationale behind the hollow hub design of the Mecanum wheel is to enhance maneuverability and facilitate precise motion control, enabling a comprehensive range of movements encompassing forward, backward, translation, and rotation [5]. By diminishing the hub's material usage, the hub's inertial mass can be reduced, thus augmenting its responsiveness and maneuvering capabilities. This heightened responsiveness assumes critical importance when rapid alterations in direction or seamless movements are necessitated. Moreover, the hollow hub configuration simplifies the installation and maintenance of Mecanum wheels [5]. The presence of hollow hubs offers more room, facilitating the easier installation and routing of cables and connectors in and around the hub. Furthermore, the wheels are affixed with 45-degree bearings to attain all-encompassing fixation, enabling omnidirectional mobility.

2.2. Analysis of Shape Transformation for Pipe Navigation

The effective transformation of deformable car into an "I" shape to navigate through pipes marks a significant achievement in enhancing its adaptability to confined spaces. This transformation was made possible by the utilization of MG996R Servo Motors, which enabled the robot to extend its sides, effectively reducing its width and facilitating movement through narrow pipes. This maneuverability showcased the robot's versatility in complex environments.

Throughout the transformation process, the activation of searchlights within the pipes demonstrated the robot's commitment to safety and operational efficiency. Improved visibility within the pipes not only enhanced mission safety but also contributed to more efficient navigation, reducing the likelihood of encountering obstacles.

Upon exiting the pipe, the seamless retraction of the robot's sides to return to the "U" shape is notable. This transition was essential for regaining stability and optimizing operational efficiency after successfully completing the specific pipe navigation task.

The successful execution of shape transformation underscored the synchronized operation of the servo motors and Arduino board. This synchronization ensured precise and timely transformations, emphasizing the effectiveness of the control system in responding to dynamic environmental demands.

2.3. Analysis of Cylindrical Shape Utilization for Downhill Scenarios

A deformable car's capability to transform into an "I" shape and employ a cylindrical configuration for downhill scenarios represents a pivotal development in optimizing its energy efficiency and operational

speed. The robot's capacity to adapt its shape by rolling onto its curved bottom surface showcases an intelligent response to varying terrains.

The pivotal role of the control system in determining the necessity of a cylindrical shape based on the terrain demonstrates a high degree of autonomy and adaptability. This real-time decision-making process empowers the robot to maximize its efficiency by reducing energy loss and enhancing stability during downhill movements.

By embracing a cylindrical shape, deformable car effectively minimizes friction and energy consumption, resulting in enhanced operational efficiency during downhill navigation. This capability not only conserves energy but also contributes to improved operational speed, making the robot highly proficient in managing downhill scenarios.

Overall, the successful execution of shape transformation for pipe navigation and the intelligent utilization of a cylindrical configuration for downhill situations highlight deformable car's adaptability and the robustness of its control system, showcasing its potential for diverse applications in challenging environments.

3. Mathematical model of smart car

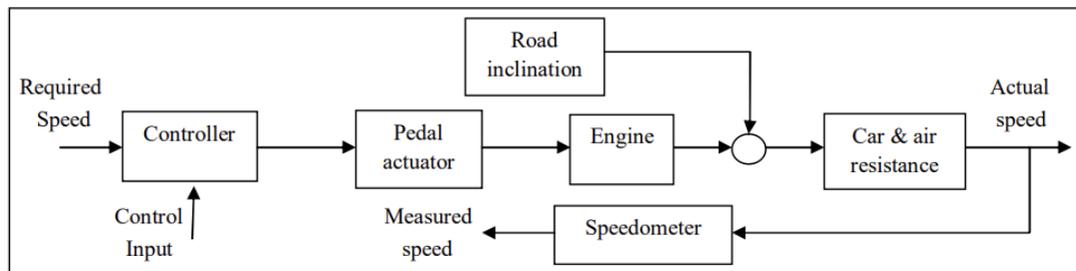


Figure 3. Flowchart for PID control of smart car [6].

Figure 3 illustrates the functional layout of a smart car, demonstrating the measurement of speed, road inclination, and air resistance in conjunction with the controller. Subsequently, the controller initiates appropriate control actions to achieve the desired output.

PI control plays a vital role in smart car navigation by dynamically adjusting the control signals sent to the car's motors or actuators. This adjustment is crucial for maintaining the car's desired trajectory and stability. The impact of PI control on smart cars can be summarized as follows:

In PI control, the proportional component, represented as 'P,' serves to calculate an error signal by measuring the disparity between the smart car's desired and actual positions [7]. Subsequently, it applies a proportional gain to this error, determining the magnitude of the control signal. In practical terms, the 'P' component plays a vital role in ensuring the smart car responds promptly to deviations from the intended path by steering in the appropriate direction. This function is instrumental in reducing oscillations and facilitating a smoother trajectory for the vehicle.

Within the PI control scheme, the integral component, denoted as 'I,' takes on the task of accumulating past error signals and applying corrective measures based on the cumulative error over time. This crucial element helps in rectifying steady-state errors that may arise due to factors such as wheel slippage or external disturbances. In the realm of smart car navigation, 'I' control becomes the linchpin for ensuring that the vehicle eventually reaches and sustains the desired position, effectively eliminating prolonged deviations from the intended path.

The standard PI control system configuration is shown in Figure 4 below.

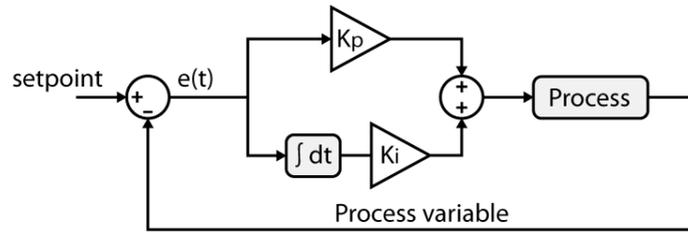


Figure 4. PID control system configuration [8].

According to Figure 4, the control signal is the sum of three terms. K_p stands for proportional term and K_i/s stands for integral term [4].

The complete PID control signal $u(t)$ is:

$$u(t) = K_p e(t) + K_i \int e(t) dt \quad (1)$$

where K_p is the proportional gain and K_i is the integral gain. Thus, the transfer function of the PI controller can be expressed by the formula:

$$\frac{U(s)}{E(s)} = K_p + \frac{K_i}{s} = \frac{K_p s + K_i}{s} \quad (2)$$

3.1. Simulate using Octave

The Octave Matlab code to simulate the above modified PI control system configuration (eq. (1) and eq. (2)) is presented in Appendix A.

The execution of the code generates one plot with three different value of K_i , shown in Figure 5 below.

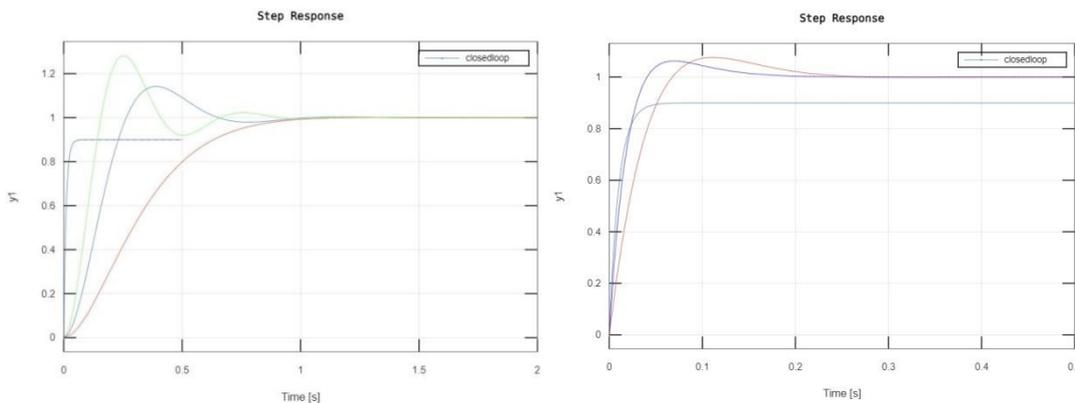


Figure 5. Step Response for PID Closed-loop System by Octave. A: Step Response for different value of K_p ; B: Step Response of PI control for different value of K_p and K_i .

From the Figure 5, it can be observed that, increasing the proportional coefficient K_p will generally speed up the response of the system and help reduce the static difference when there is a static difference [7]. However, an excessively large proportional coefficient will cause the system to have a large overshoot and produce oscillation, which will worsen the stability of the system.

K_i , as long as there is an error in the system, the integral effect will continue to accumulate, and the control quantity will be output to eliminate the error. Too strong an integral effect will increase the system overshoot and even cause the system to oscillate.

3.2. Simulate using Falstad

Falstad is used to verify the outcome from the Octave and reveal how PI control is built in the circuit. Figure 6 indicates the layout of circuit of PI control system.

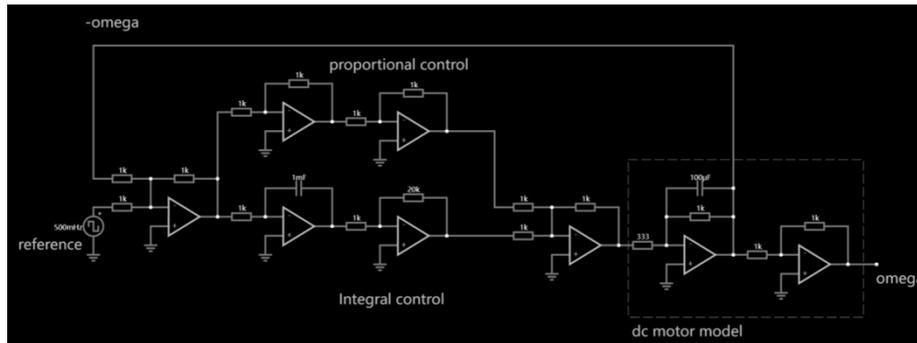


Figure 6. circuit of PI control system in Falstad.

In order to further explore the impact of K_i and K_p values on the PI system, this report selected different K_p and K_i values for simulation, and the results are shown in Figure 7 below.



Figure 7. Reference tracking for different value of K_p and K_i in Falstad. A: Reference tracking for $K_p=1$, $K_i=20$; B: Reference tracking for $K_p=2$, $K_i=20$; C: Reference tracking for $K_p=1$, $K_i=40$.

According to figure 7, certain properties can be derived. For Proportional Gain (K_p), Increasing the value of K_p makes the controller respond more aggressively to the current error. This means that the controller will react more quickly to deviations from the desired setpoint, resulting in a faster initial response. However, if K_p is set too high, it can lead to overshoot, where the system's output oscillates around the setpoint before settling. Overshoot can be undesirable in applications that require stability and precision. Conversely, decreasing K_p slows down the controller's response to error, which can reduce overshoot but may also result in longer settling times. Finding the optimal K_p value depends on

the specific system and control objectives. It involves balancing K_p to achieve a fast response without excessive overshoot.

For Integral Gain (K_i), it primarily addresses steady-state errors that persist over time. Increasing K_i strengthens the integral action, helping the controller eliminate long-term discrepancies between the setpoint and the actual process variable. However, setting K_i too high can lead to integrator windup, a situation where the integral term accumulates an excessive amount of error. This causes the controller to respond too aggressively when the error starts to decrease, potentially resulting in overshoot and instability. Reducing K_i weakens the integral action, making the controller less effective at eliminating steady-state errors, which may lead to larger steady-state errors in the system's response. Achieving the right balance for K_i is crucial. It should be adjusted to eliminate steady-state errors without causing excessive overshoot or instability.

K_p and K_i often have trade-offs. Increasing K_p for a faster initial response can sometimes lead to an increase in steady-state error, which may be compensated by increasing K_i . However, this should be done cautiously to avoid excessive oscillations or instability. The tuning process typically involves adjusting K_p first to achieve the desired response speed and then fine-tuning K_i to eliminate steady-state errors. This sequential tuning process allows for better control over the balance between responsiveness and precision.

4. Conclusion

This research, at its core, signifies a significant leap forward in the realm of adaptable robotic systems. The integration of deformable smart car design and PI control has unveiled unprecedented potential for enhancing the efficiency, adaptability, and precision of robotic systems. The smart car's capability to transform between "U" and "I" shapes, coupled with its adept navigation through narrow pipes, showcases its versatility and practical utility in complex environments.

Moreover, the intelligent utilization of cylindrical shapes during downhill scenarios highlights the robot's autonomous decision-making capacity and its prowess in conserving energy and improving stability. These features make it a formidable asset in search and rescue operations, disaster-stricken areas, and challenging terrains.

The mathematical model underpinning the smart car's control system, with PI control at its core, illuminates the pivotal role of this mechanism in maintaining trajectory, stability, and precise navigation. Proportional, integral, and derivative components synergize seamlessly, ensuring that the smart car adapts promptly and accurately to external factors.

In summary, the fusion of deformable smart car design, PI control, and shape transformation mechanisms represents a remarkable breakthrough in robotics. This research underscores not only their transformative impact but also their potential to redefine the possibilities of adaptable robotic technology. As we peer into the future, these innovations promise to usher in a new era of robotics, where adaptability, precision, and efficiency converge to address the most pressing challenges in disaster response and search and rescue operations.

References

- [1] Åström, K.J. and Hägglund, T., 2001. The future of PID control. *Control engineering practice*, 9(11), pp.1163-1175.
- [2] Zhou, Y., Xu, G., Qin, F., Xu, K., Wang, G., Ou, Y., Lin, G. and Zhang, Q., 2013, December. The prospect of smart cars: Intelligent structure and human-machine interaction. In *2013 IEEE International Conference on Robotics and Biomimetics (ROBIO)* (pp. 1899-1904). IEEE.
- [3] Araki, M., 2009. PID control. *Control Systems, Robotics and Automation: System Analysis and Control: Classical Approaches II*, pp.58-79.
- [4] Leros, A., Andreatos, A. and Zagorianos, A., 2010, July. Matlab-Octave science and engineering benchmarking and comparison. In *Proceedings of the 14th WSEAS international conference on Computers: part of the 14th WSEAS CSCC multiconference* (pp. 746-754).

- [5] Han, K.L., Choi, O.K., Kim, J., Kim, H. and Lee, J.S., 2009, August. Design and control of mobile robot with mecanum wheel. In 2009 ICCAS-SICE (pp. 2932-2937). IEEE.
- [6] Xiaojun, L. and Zhongbing, Y., 2017. Design of Smart Car Control System Based on Camera. Acta Electronica Malaysia, 1(1), pp.15-17.
- [7] Knospe, C., 2006. PID control. IEEE Control Systems Magazine, 26(1), pp.30-31.
- [8] Fernandez, G., Cherix, N., & Strobl, S. (2023, May 8). Pi controller implementation for current control. imperix. <https://imperix.com/doc/implementation/pi-controller>

Appendix A

The Octave-Matlab code to simulate the modified PI control system configuration (eq. (1) and eq. (2) is as follows:

```
clf
s = tf("s")
motor = 3 / (0.1 * s + 1)
cont = 3
closedloop = cont * motor / (1 + cont * motor)
step(closedloop)
hold on

cont = 1 + 20 / s
closedloop = cont * motor / (1 + cont * motor)
step(closedloop,"r")
hold on

cont = 2 + 40 / s
closedloop = cont * motor / (1 + cont * motor)
step(closedloop,"b")
hold on
```

Appendix B

```
clf
s = tf("s")
motor = 3 / (0.1 * s + 1)
cont = 3
closedloop = cont * motor / (1 + cont * motor)
step(closedloop)
hold on

cont = 3 / s
closedloop = cont * motor / (1 + cont * motor)
step(closedloop,"r")
hold on

cont = 1 / s
closedloop = cont * motor / (1 + cont * motor)
step(closedloop,"b")
hold on
```

```
cont = 6 / s  
closedloop = cont * motor / (1 + cont * motor)  
step(closedloop,"y")  
hold on
```