CRUISE CONTROL SYSTEM

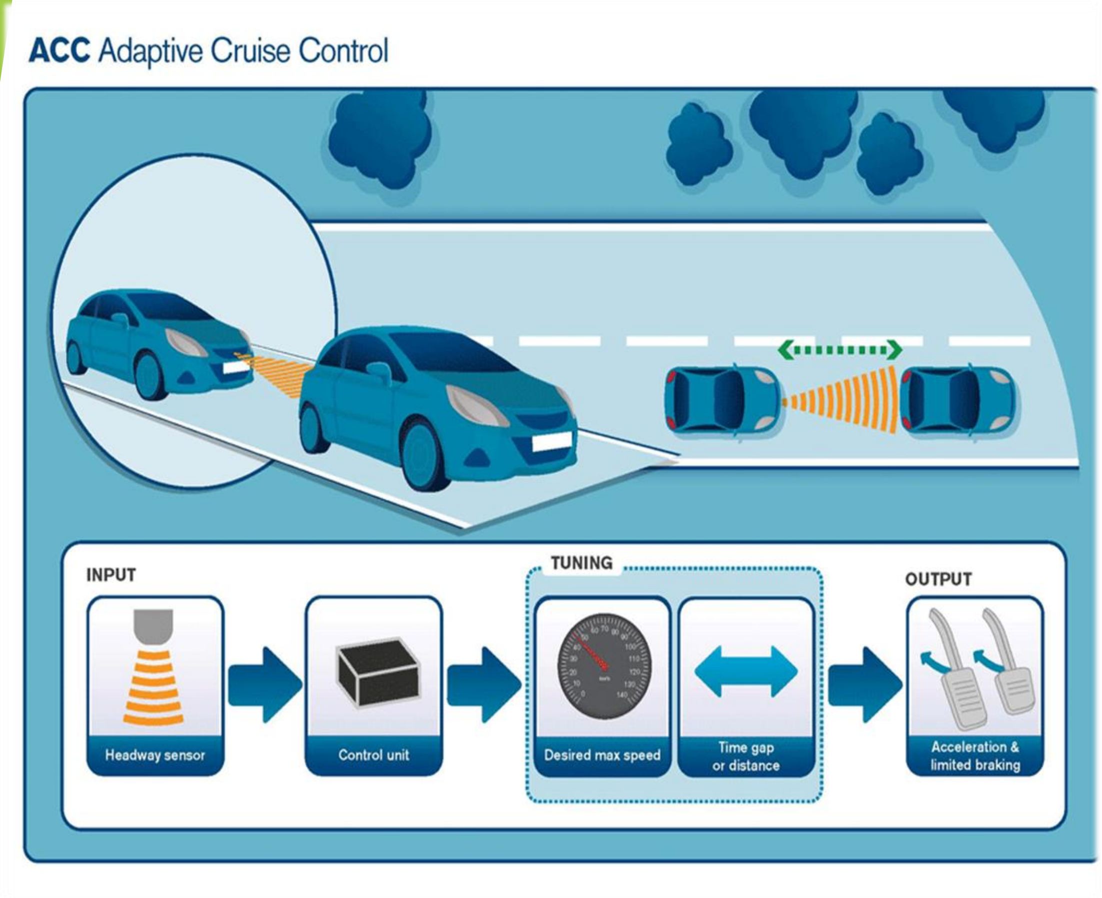
J.SAI SHANMUKHNATH

V.SHANTAN RAMI REDDY

CONTENTS:

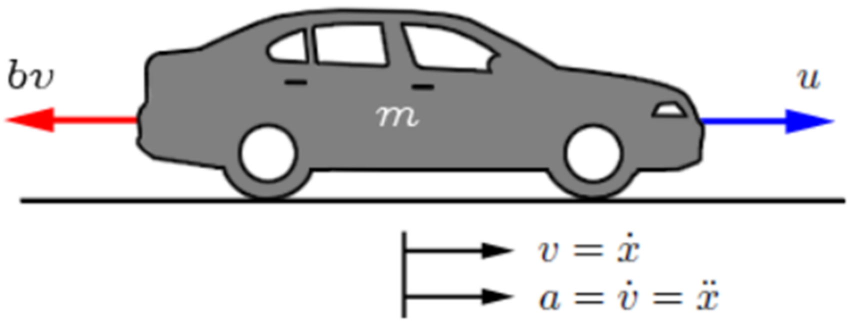
- INTRODUCTION
- METHODOLOGY
- TYPES OF CONTROLLERS
- CONCLUSION

INTRODUCTION:



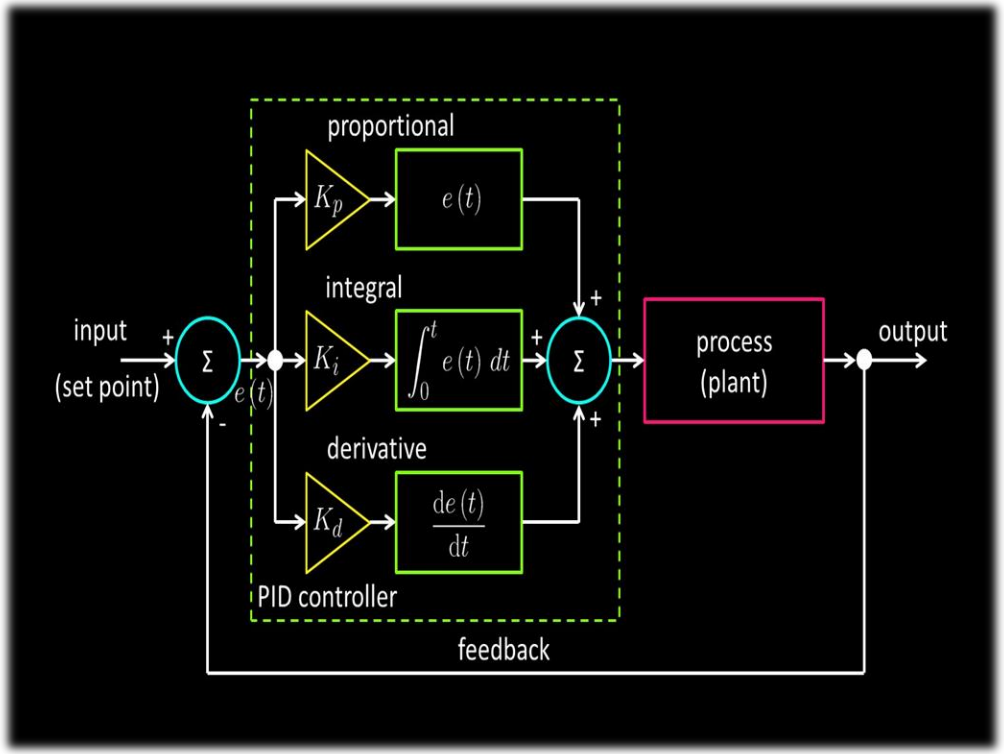
- Cruise Control System (CCS) helps maintain a constant vehicle speed without continuous driver input, making long drives more comfortable and fuel-efficient. It uses a closed-loop feedback system where speed sensors measure the current velocity, and a controller (often PID) adjusts the throttle based on the difference between actual and desired speeds.
- The system accounts for factors like vehicle mass, drag, engine force, and road slopes, often modeled in tools like MATLAB/Simulink. We have used MATLAB for this project. It aims for smooth and stable control, avoiding speed overshoot or delay. Components include speed sensors, throttle actuators, a microcontroller for control logic, and user controls. Safety features automatically deactivate CCS when the driver brakes or if a fault occurs.
- Modern CCS may include adaptive features, using radar or lidar to maintain safe distances—known as Adaptive Cruise Control (ACC). These advanced systems integrate mechanical, electrical, and software elements, relying on techniques like sensor fusion and adaptive algorithms for reliable performance in varying conditions.

METHODOLOGY:



- The mass of the car is represented by ‘m’, which is acted upon by the force, pull ‘u’ that is generated at the interface between the tire and the surface of the road. It is also assumed of the model that ‘u’ can be controlled while the dynamic of the power train is neglected. The oppositional pull, ‘v’ is assumed to be varied linearly with velocity of the car, and it acts opposite to the motion of the car.
- First-order mass-damper system.
- $m\dot{v} + bv = u$, $y = v$
- The state-space representation: $\dot{v} = (-b/m)\, v + (1/m)\, u$, $y = (1)(u)$
- The transfer function: $G(s) = \frac{1}{ms+b}$
- This shows how input throttle ‘u’ affects the output speed ‘v’ in the cruise control system.

CONTROLLER:



- The PID controller, developed in 1910 and popularized by the Ziegler–Nichols tuning method (Set Ki and Kd to 0) (Ultimate Gain Method) in 1942, remains the most widely used control method in industry due to its simplicity, effectiveness, and ease of use. Over 90% of industrial controllers still rely on PID at the basic level.
- A PID controller combines three actions:

a) Proportional (Kp): reacts to current error,

b) Integral (Ki): eliminates steady-state error by considering past errors.

c) Derivative (Kd): predicts future error for better transient response.

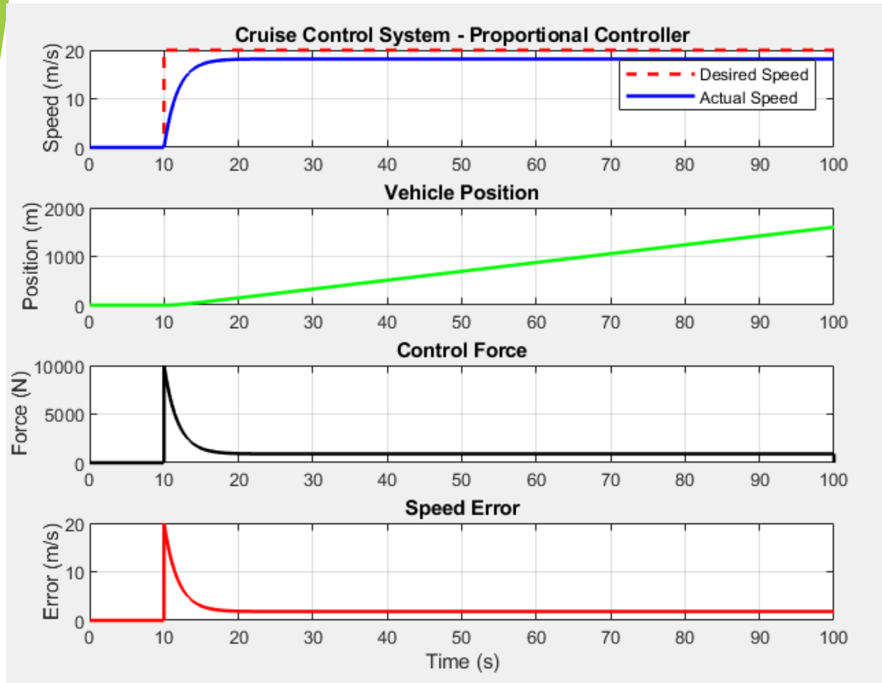
- Its transfer function can be expressed in:

a) Parallel form: $G(s) = Kp + Ki/s + Kd \cdot s$

b) Ideal form: $G(s) = Kp(1 + 1/Ti \cdot s + Td \cdot s)$

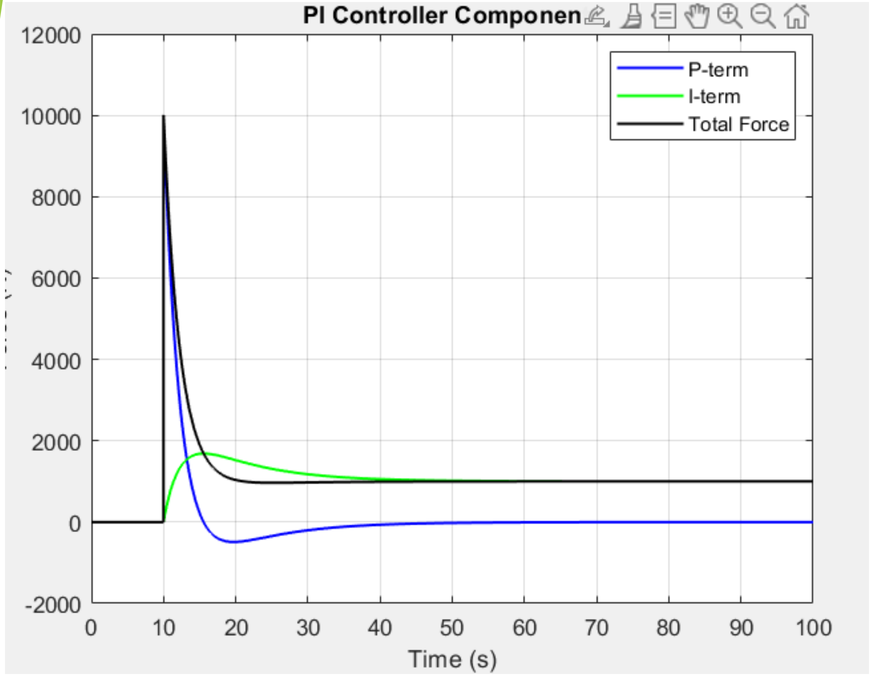
PID can be seen as a special case of phase lead-lag compensation, offering both transient and steady-state performance improvements.

P CONTROLLER:



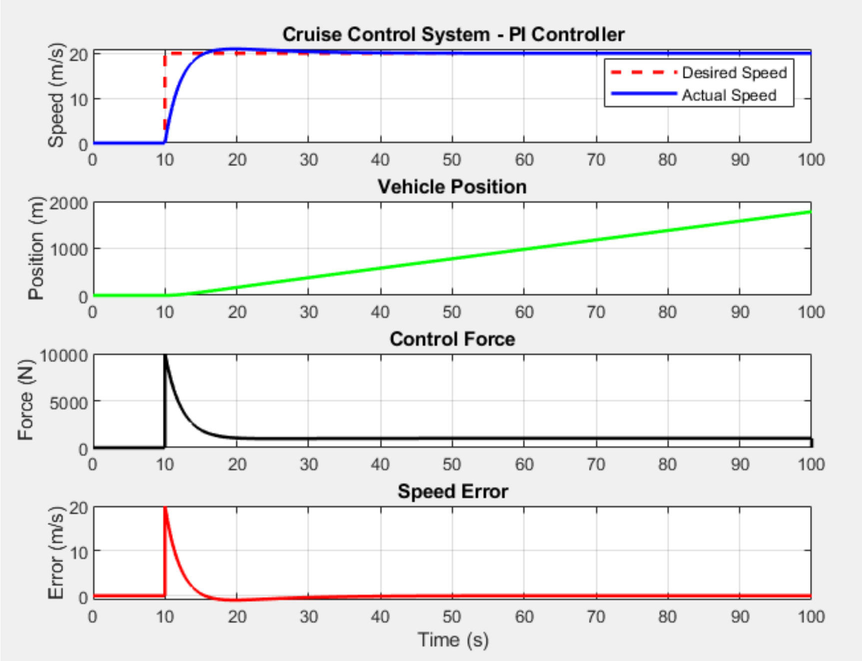
- This image shows a cruise control system using only a Proportional (P) controller over 100 seconds.
- Top graph: The vehicle accelerates towards the desired speed (20 m/s) but doesn't fully reach it, showing a steady-state error.
- Second graph: The vehicle’s position increases steadily as it moves forward.
- Third graph: The control force initially spikes (around 9,000 N) to accelerate the vehicle, then settles at a constant value.
- Bottom graph: The speed error starts high and decreases but never reaches zero, stabilizing at a small steady-state error.
- The key point: A P controller reduces error but cannot eliminate steady-state error, as it can't fully compensate for system forces like friction.

PI CONTROLLER COMPONENTS:



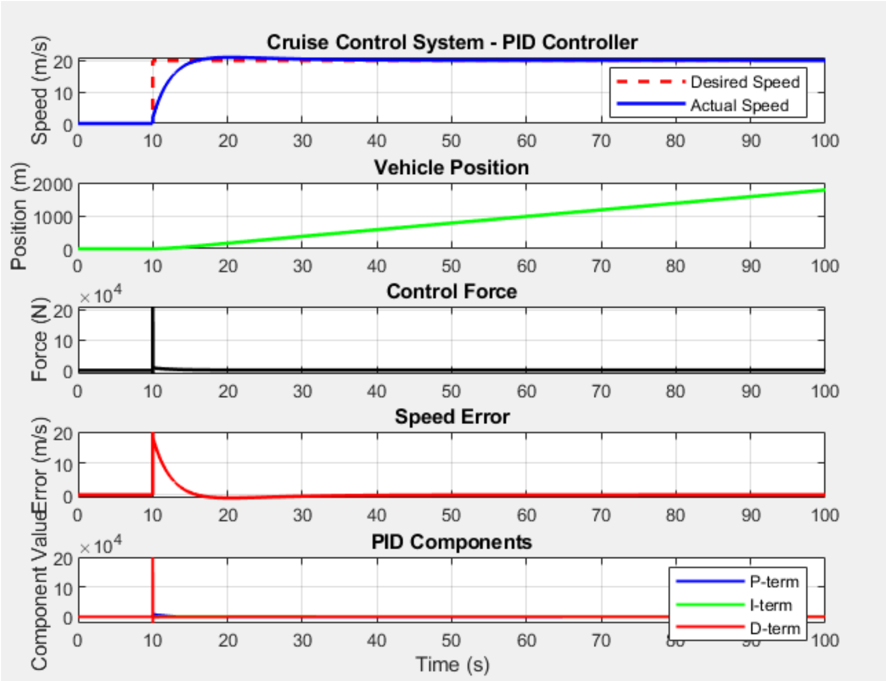
- This graph shows the breakdown of a PI controller for a cruise control system over 100 seconds.
- P-term (Blue line): Initially spikes to about 8,000 N at 10 seconds, then drops and briefly turns negative (15–40 seconds) to correct overshoot, stabilizing near zero later.
- I-term (Green line): Gradually rises from zero, reaching around 1,600 N to maintain the target speed by countering steady disturbances.
- Total Force (Black line): Peaks around 10,000 N during acceleration, then settles close to the I-term value (~1,000–1,200 N) at steady state.
- Overall, the P-term ensures quick response, while the I-term removes steady-state error, working together to stabilize the system.

PI CONTROLLER:



- This image shows a cruise control system using a PI (Proportional-Integral) controller over 100 seconds.
- Top graph: The vehicle accelerates from 0 to the desired 20 m/s, reaching it around 15 seconds with minimal overshoot.
- Second graph: The vehicle's position steadily increases, with a steeper slope after reaching target speed.
- Third graph: The control force peaks around 10,000 N initially, then settles around 2,000 N to maintain speed.
- Bottom graph: The speed error starts high but quickly drops to near zero as the vehicle matches the desired speed.
- Overall, the PI controller effectively brings the vehicle to the target speed, minimizing steady-state error and applying steady control.

PID CONTROLLER:



- This image shows a cruise control system using a PID controller over 100 seconds.
- Top graph: The vehicle accelerates from rest to 20 m/s by around 15 seconds with minimal overshoot.
- Second graph: The vehicle's position steadily increases as it maintains speed.
- Third graph: The control force spikes to about 20,000 N initially, then quickly drops to a steady level.
- Bottom graph: Shows the P, I, and D term contributions — P and D spike early to improve acceleration, while the I term grows to eliminate steady-state error.
- Overall, the PID controller achieves quick, stable speed tracking with improved transient and steady-state performance.

CONCLUSION:

- The cruise control system was successfully analyzed and simulated using different control strategies: Proportional (P), Proportional-Integral (PI), and Proportional-Integral-Derivative (PID) controllers. The Proportional controller provided a fast response but exhibited a steady-state error, failing to fully achieve the desired speed. Adding an Integral component (PI controller) eliminated the steady-state error, improving long-term accuracy and ensuring the vehicle maintained the target speed with minimal overshoot. However, the PI controller showed slight overshoot and longer settling time during transients.
- The PID controller further enhanced performance by introducing a Derivative term, which significantly improved the transient response, minimized overshoot, and enabled quicker settling without compromising stability
- The simulations demonstrated that the PID controller delivered the best overall performance for cruise control, providing both rapid error correction and steady-state accuracy. Therefore, for applications demanding both quick adaptation to changes and precise speed maintenance, a well-tuned PID controller is the most effective solution.
- The study highlights the importance of selecting appropriate control strategies and fine-tuning controller parameters to achieve reliable, smooth, and efficient cruise control operation under varying driving conditions.

REFERENCES:

1. https://www.researchgate.net/publication/353925596_CONTROL_DESIGN_AND_ANALYSIS_OF_CRUISE_CONTROL_SYSTEM
2. https://ieeexplore.ieee.org/document/10009608
3. https://ctms.engin.umich.edu/CTMS/index.php?example=CruiseControl&section=SystemModeling
4. https://ieeexplore.ieee.org/document/1453566
5. https://www.digikey.com/en/maker/projects/introduction-to-pid-controllers/763a6dca352b4f2ba00adde46445ddeb


THANK YOU