A Joystick Driving Control Algorithm with a Longitudinal Collision Avoidance Scheme for an Electric Vehicle

Mooncheol Won*

Department of Mecharonics, Chungnam National University Daejeon, 305-764, Korea

In this paper, we develop a joystick manual driving algorithm for an electric vehicle called Cycab. Cycab is developed as a public transportation vehicle, which can be driven either by a manual joystick or an automated driving mode. The vehicle uses six motors for driving four wheels, and front/rear steerings. Cycab utilizes one industrial PC with a real time Linux kernel and four Motorola MPC555 micro controllers, and a CAN network for the communication among the five processors. The developed algorithm consists of two automatic vehicle speed control algorithms for normal and emergency situations that override the driver's joystick command and an open loop torque distribution algorithm for the traction motors. In this study, the algorithm is developed using SynDEx, which is a system level CAD software dedicated to rapid prototyping and optimizing the implementation of real-time embedded applications on distributed architectures. The experimental results verify the usefulness of the two automatic vehicle control algorithms.

Key Words : Collision Avoidana, Joystick Control, Electric Vehicle, CAN, Cycab

1. Introduction

The use of private automobiles, especially in city centers, had led to severe problems of congestion, pollution, safety and general degradation of the quality of life. The CyberCars project (Parent, 2000) in Europe is aimed at developing a new mobility, with an alternative solution to the private passenger car, having the same flexibility and much less nuisances. The Cycab, which is a small electric vehicle developed by the Robosoft company in France, can be one of the public transportation means for new public transportation systems. The Cycab can be driven by either a joystick manual mode or an automated mode such as platooning. Figure 1 shows the outlook of Cycab.

* **E-mail** : mcwon@cnu.ac.kr
TEL : +82-42-821-6875; **FAX** : +82-42-823-4919
Department of Mecharonics, Chungnam National University Daejeon, 305-764, Korea. (Manuscript **Received** April 26, 2002; **Revised** July 7, 2003)

In a joystick driven vehicles, the longitudinal joystick command from drivers can be interpreted as either vehicle speed command or motor torque command. In this study, the joystick command is interpreted as a motor torque command. This torque interpretation can be more natural than speed command interpretation, since the torque interpretation is in accordance with the acceleration pedal command of internal combustion engine vehicles, which means more familiarity to drivers.

Also, from the controller design point of view, the torque interpretation is easier than the other, since we can directly relate the joystick input to the motor PWM duration. As an exception to the torque interpretation, the zero longitudinal joystick command, which occurs when the driver releases the joystick, is interpreted as slowing down and stop command rather than a zero torque command. This feature makes the driving be easier when the driver tries to stop on a slope, since Cycab has only an emergency on/off type electromagnetic brake. Without the feature, the driver must control the joystick to issue the

Fig. 1 The outlook of cycab

torque command that cancels the gravity force if the driver maintains zero speed on an hill.

This study develops control algorithms for traction motors that achieve the following three features.

(1) A slowing down and stop algorithm using feedback on uphill and downhill for the zero joystick command in normal driving situations. (Normal Stop Control : NSC)

(2) An emergency slowing down and stop algorithm for possible collision situations. (Emergency Collision Avoidance Control : ECAC)

(3) An open loop torque distribution algorithm that can compensate the back electromotive force (EMF) voltage drop and realize power matching in the left and the right wheel motors. (Motor Torque Distribution : MTD)

The first and the second algorithms are automatic closed loop control algorithms. In the first algorithm, the wheel speed information is utilized to achieve a predetermined linearly decaying velocity profile. The ECAC is useful for safe manual drivings when an obstacle comes into a road abruptly. For the distance measurements between obstacles and Cycab, an ultrasonic sensor system is developed and interfaced with Cycab. The distance information and the wheel speed information are used as feedback information to control vehicle speed. The three algorithms are combined together and implemented to Cycab. Experimental results verify the usefulness of the NSC and ECAC algorithms.

This paper is organized as follows. Section 2 describes the Cycab hardware and the real time control software. Section 3 explains the developed control algorithm. Section 4 shows the experimental results for NSC and ECAC algorithms, and summary and conclusions follow.

2. Cycab Hardware and Real-time Control Software

2.1 Cycab Hardware

The detailed Cycab specification is shown in (Lisowski and Parent, 1997), In this study, we explain basic components related to vehicle control.

The specific Cycab used in this study utilizes six D.C motors, a joystick, and 48 V battery power for moving. Four motors are used for traction and soft braking of each four wheel independently, and two motors are used for steering control of front wheels and rear wheels. Four on/off type electromagnetic brakes are used as emergency brakes on traction motors. Each wheel of the front or the rear is connected by a typical four bar mechanical link, which means their steering angles are not independent. One industrial PC with a real time Linux kernel and four Motorolla MPC555 micro-processors are used to control those six motors and to get the joystick command. For the necessary data exchange between the processors, a CAN (controller area network) (Lawrenz, 1997) is used.

The baud rate of CAN used is 125 kbps, and the message parameters include joystick commands, wheel speeds, and steering angle, etc. There are also four encoders for measuring the wheel speeds and two absolute encoders for front and rear steering angle measurements. Figure 2 shows the schematics of the hardware of the Cycab used in this study.

The roles of MPC555 are the acquisition of the sensor information including vehicle speed, steering angle, and joystick command, and generation of the motor PWM durations for motor amplifiers. The control algorithm is distributed to four MPC555s and the PC using the communications on the CAN bus. It is SynDEx, which decides the optimized distribution of computing operation on the processors of the architecture in order to minimize the total execution duration. Other roles of PC include downloading the

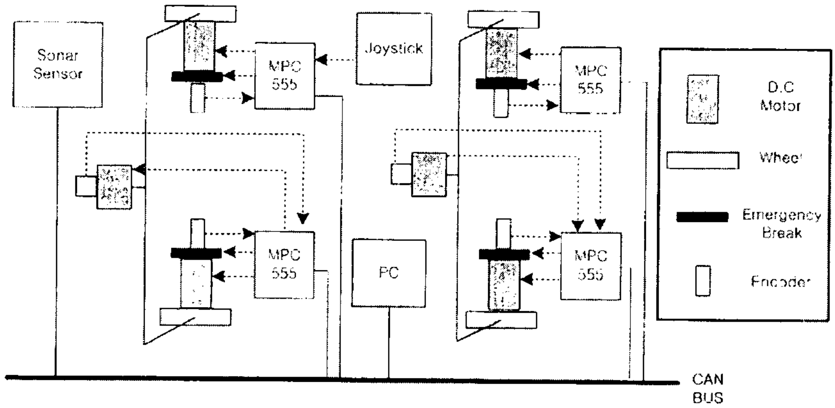


Fig. 2 Hardware schematics of Cycab

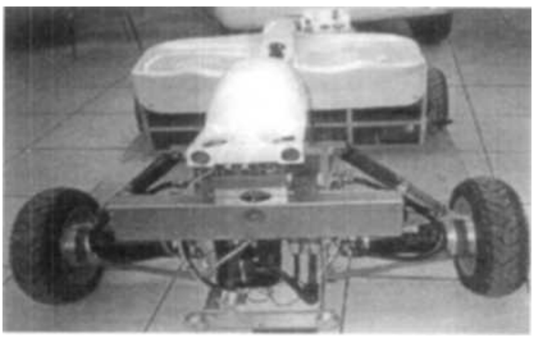


Fig. 3 Ultrasonic sensor system mounted on Cycab

executives for MPC555 through CAN and offering an ergonomic GUI that allows to monitor the control related variables including sensor values.

2.2 Ultrasonic sensor system

Two ultrasonic sensors are used to measure the distance between Cycab and obstacles. Two Polaroid 6500 series sonar ranging modules are used with a PIC16F877 micro-processor to control the ultrasonic sensor activation. Figure 3 shows the ultrasonic sensor system mounted on Cycab. The micro-processor also sends the measured distance to PC with the aid of a CAN controller. The reliable sensing range of the ultrasonic sensor is about 40 cm to 9 m. Since the sensor's sensing frequency increases with reduced distance from an obstacle, the sensor system sends the distance information at arbitrary time instance. A real time task that handles the ultrasonic sensor data reception is created in the PC program, which is independent to the periodic control tasks scheduled by SynDEx.

2.3 Real time control software

The control software has been developed using SynDEx developed by INRIA in France, and the RTAI(Real Time Application Interface) kernel. A detailed description of SynDEx can be found in website (http://www-rocq.inria.fr/syndex/). SynDEx is a system level CAD software, supporting the "Algorithm Architecture Adequation" (AAA) methodology (Grandpierre et al., 1999), for rapid prototyping and optimizing the implementation of real-time embedded applications on multi-component architectures.

3. Developed Algorithms

A good motor control algorithm is necessary to achieve a good interpretation of user's driving intention from the joystick, good driving quality, and emergency velocity reducing and stop capability to avoid possible collisions with obstacles. Three separate algorithms are developed and integrated for a joystick maneuver. They are normal stop control on a hill (NSC), emergency collision avoidance control (ECAC), and motor torque distribution (MTD). NSC automatically slows down and stops Cycab on uphill/downhill in normal driving situations, i.e, situations without collision possibility. ECAC slows down and stops Cycab to avoid collision with obstacles in emergency. MTD tries to compensate motor back EMF, and to match powers of the left wheels and the right wheels. The flow chart for the integrated algorithm is shown in Figure 4.

In NSC and ECAC algorithms, computed longitudinal commands are generated that replace the user joystick command. The MTD algorithm modifies the human joystick command, and also serves for an inner loop controller in the two automatic modes. Figure 5 shows the schematics of the control algorithm. In Figure 5, the manual joystick control by human is represented inside the dotted box. In the two automatic control cases

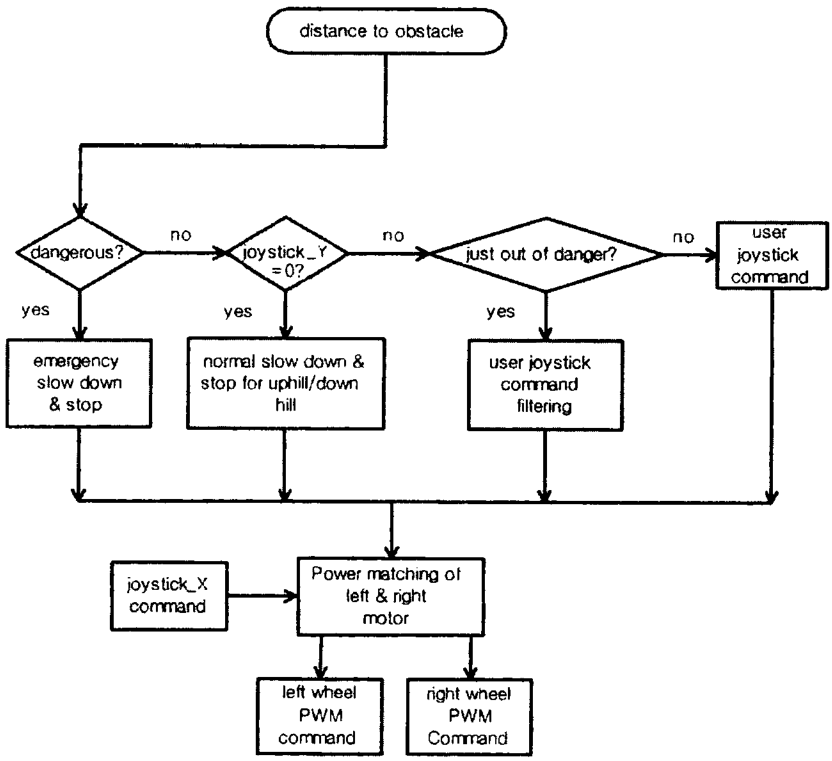


Fig. 4 Flow chart of integrated algorithm

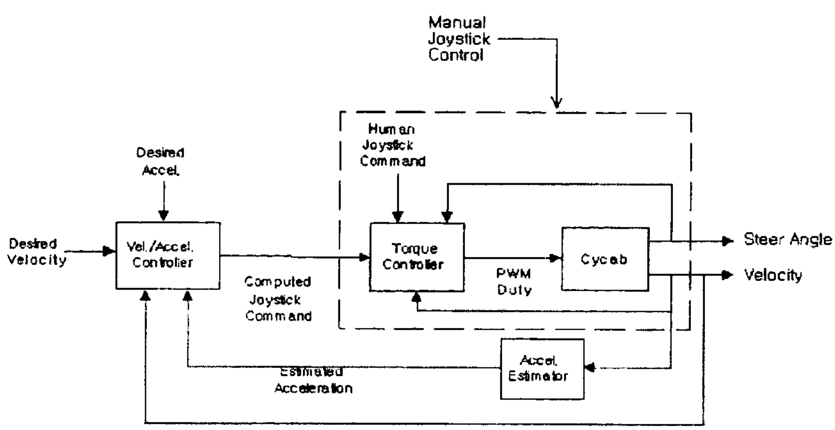


Fig. 5 Control algorithm schematics

(NSC and ECAC), the desired acceleration and velocity can be generated from the specific slow down and stop strategies and the desired distances from obstacles. In these cases, the human joystick command is ignored, and the computed fictitious joystick command generated by the outer velocity/acceleration controller is used in the control loop. We can think these cases as the computer controller giving the joystick command in place of a human driver.

In the following, the three algorithms are presented in detail.

3.1 Normal stop control on a hill (NSC)

Cycab does not have a brake system similar to a hydraulic brake for commercial cars, which can apply gradual braking force. It has only emergency on/off electro-magnetic brake system, which gives a full brake torque when switched on, and gives no brake torque when switched off. Therefore, to reduce the vehicle speed gradually, the vehicle should use the negative torque from the motors. When the driver put the joystick longitudinal command to zero (zero torque command) on a moderate uphill or a downhill, the vehicle gradually reduces speed and finally stops because of the rolling resistance (Wong, 1993). However, on a relatively steep hill, the vehicle does not stop, and slides down because of the large gravity force. Therefore, drivers must control the joystick with a great care to maintain zero speed on such a hill, which is a great burden. To reduce this effort, a servo stop algorithm is developed. This algorithm also includes a velocity reducing feature to realize a smooth stop. As soon as the driver puts the joystick longitudinal command to zero during driving in any speed and on any slope, then the algorithm starts to make the vehicle speed reduced linearly and finally makes the vehicle stop even on a very steep hill.

For this automatic speed reducing control, an integral (I) type control algorithm is utilized along with a state feed back structure inside the integral, i.e., the computed joystick command, J_c is given as

$$J_c = \int_0^t \{ K_v(v_d - V) + K_a(a_d - a)\} \, dt \qquad (1)$$

where K_v and K_a are positive state feedback gains, v_d is the desired velocity, v is the vehicle velocity, and a_d is the desired acceleration, and a is the acceleration.

For digital implementations, the control law is interpreted as

$$J_c(k) = J_c(k-1) - \{ K_v(v_d(k) - V(k)) + K_a(a_d(k) - a(k))\}\Delta T \qquad (2)$$

where $\Delta T (=100 \text{ msec})$ is the control loop time. This algorithm resembles the control strategy of human, since human add or subtract their control command from the previous command. The terms inside the integral is a state feed back form. With only the speed error feedback, the speed error shows considerable overshoots and oscillations for necessary large values of velocity feedback gains. For the acceleration feedback, the vehicle acceleration is estimated by differentiating and applying a simple first order filter.

The desired speed history is chosen as a linear line with a negative slope, which means a constant deceleration. When the speed is less than 0.16 m/sec., another integral type stop algorithm is used.

$$J_c(k) = J_c(k-1) - K \, sign(v(k)) \qquad (3)$$

Here, K is a positive constant gain.

3.2 Emergency collision avoidance control (ECAC)

This algorithm is developed to avoid possible collisions with another vehicles or pedestrians during the joystick maneuver caused by abrupt interruptions of obstacles or user's lack of concentration on maneuver. In the following, a hazardous region of maneuver is defined by the vehicle speed and the distance from the obstacles.

First, suppose a simple situation that a non-moving obstacle is located D meters in front of Cycab which has forward speed V_0 m/sec. To make Cycab stop d_0 meter in front of the obstacle with a constant deceleration maneuver, the constant deceleration value is calculated as $-\frac{1}{2}V_0^2/(D-d_0)$ m/sec^2. We can generalize this value of deceleration at any instant of time and

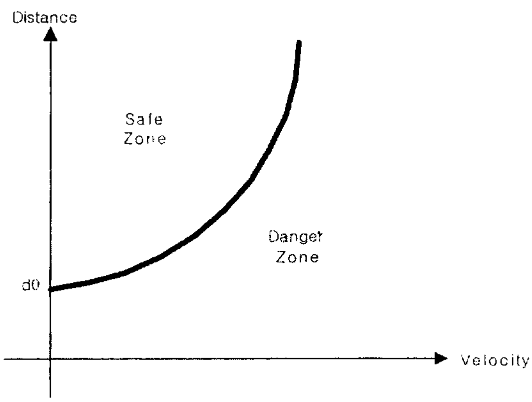


Fig. 6 Hazardous region definition

for moving obstacles, i.e., the desired deceleration is

$$a_d(t) = -\frac{1}{2} V_0^2(t) / (D(t) - d_0) \, \text{m/sec}^2 \quad (4)$$

In this study, the hazardous region is defined as

$$\frac{1}{2} V_0^2 / (D(t) - d_0) > 0.5 \, \text{m/sec}^2 \quad (5)$$

This criterion is based on the fact that people usually decelerate the passenger car with a deceleration less than 0.05 g, when they only release the accelerator pedal and do not use the brake. Graphically, the region can be represented as Figure 6.

Here, again the velocity reducing and stopping algorithm in section 3.1 is utilized to realize the desired deceleration value. The only difference is in the desired velocity and acceleration. The desired velocity necessary for state feedback structure is obtained from the numerical integration of the desired deceleration in equation (4).

$$v_d(k) = v_d(k-1) + a_d(k) \Delta T \quad (6)$$

Here, the desired acceleration can be time varying rather than a constant in section 3.1. Whenever, the hazardous region is escaped by sufficient velocity reduction from computer control, the driver's joystick command is reactivated. However, a first order filter with a large time constant is applied for the driver's joystick command until the filtered joystick command becomes close to the unfiltered joystick command. The reason for applying the slow low pass filter is to improve the riding quality. Without the filter, considerable jerks are experienced, which is induced from the abrupt exchange of command between computer and driver commands near the hazardous border. Even in the hazardous region, if the user joystick command is more negative than the automatically calculated command, the user command overrides the computer calculated command, which gives higher priority to human drivers in controlling dangerous situations.

3.3 Motor torque distribution algorithm (MTD)

This algorithm is developed to decide relevant motor amplifier PWM (Pulse Width Modulation) durations from the joystick command. The command is interpreted as a motor torque command in this implementation. In D.C motors, there is a back EMF voltage drop, which is linearly proportional to motor speed, i.e.,

$$V = RI + K_b \omega \quad (7)$$

$$T = K_t I \quad (8)$$

where V is the applied voltage, R is motor resistance, I is motor current, ω is motor speed, K_b is back EMF constant, K_t is motor torque constant, and T is motor developing torque. In equation (7), the motor inductance effect is neglected. The PWM pulse width is proportional to the applied voltage in average. Therefore, if we construct the PWM command using the joystick command and the back EMF drop voltage, then the PWM command will represent the joystick command interpreted as a torque command. In this study, the PWM pulse width command of each wheel is decided as

$$PW_l = \alpha J_y S_l + \beta K_b \omega_l \quad (9)$$

$$PW_r = \alpha J_y S_r + \beta K_b \omega_r \quad (10)$$

where α and β are unit conversion constants, J_y is the longitudinal joystick command, S_l and S_r are scaling factors for power matching, and and ω_l are ω_r the wheel speed of the left and right wheels. The second term in each equation compensates the Back EMF voltage drop using the motor speed measurements. The first terms correspond to the desired torque, since the developed torque is proportional to the motor current.

The use of the scaling factors is to have the power matching on the left and the right motors. When the vehicle turns a corner, the left wheel speed and the right wheel speed are different. Inspired by the differential gear mechanism of commercial automobiles, the left wheel torque and right wheel toque commands are decided differently to match the power of the left and right wheels, i.e, to satisfy $T_l\omega_l = T_r\omega_r$, where T_l and T_r are left wheel torque and right wheel torque respectively. For that purpose, S_l and S_r are decided by

$$S_l/S_r = \omega_r/\omega_l \qquad (11)$$

$$(S_l+S_r)/2 = 1 \qquad (12)$$

The speed ratio, ω_r/ω_l is a function of the steering angle. A predetermined function from experiments and the steering angle command (lateral joystick commands, J_x) are used to get approximated speed ratio. To check whether the algorithm realizes the same power, we need to put torque sensor on each wheel. However, the torque sensors were not available, the algorithm could not be verified experimentally.

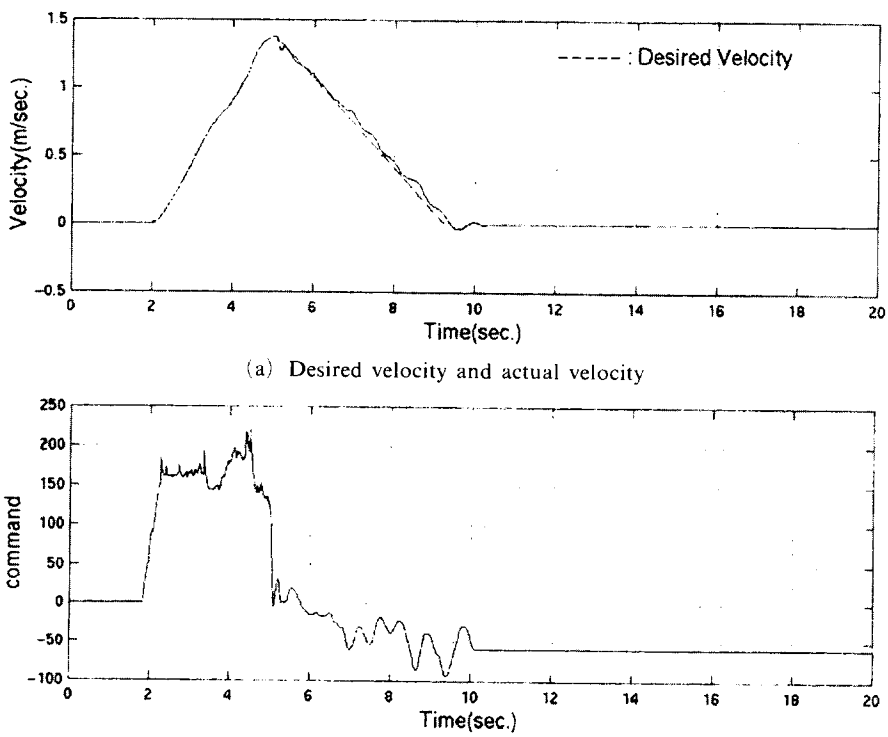
The joystick commands obtained from potentio-meters are quite noisy. Therefore, the joystick input signals are filtered by a digital first order filters, and a neutral region is defined to have noise immunity for zero mean random noise.

4. Experimental Results

This section presents two kinds of experimental results. One (NSC) is automatic stop on uphill/downhill in normal situations, and the other (ECAC) is emergency collision prevention stop using the ultrasonic sensor feedback. All experiments are tested on straight roads.

4.1 Normal stop control on a hill (NSC)

Two automatic stop experiments have been car-



(a) Desired velocity and actual velocity

(b) Human joystick or computed command

Fig. 7 Automatic slow down and stop on a downhill ried out. One is the case when Cycab is running down a hill, and the other is the case when Cycab is climbing up.

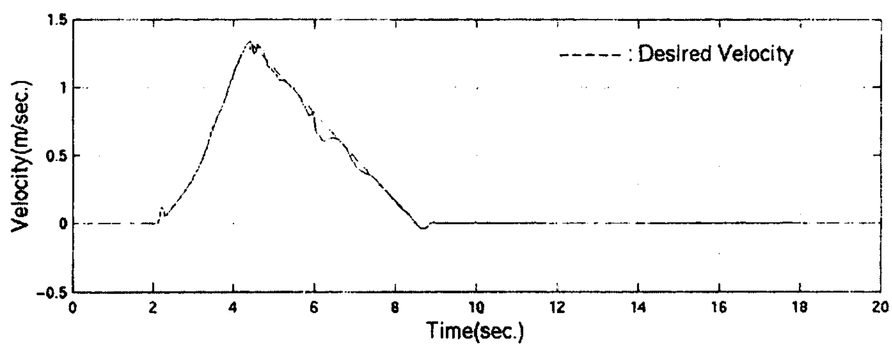
Figure 7(a) shows the actual and desired velocity for the down hill case. Up to 5.0 second, the driver applied positive joystick command to increase the speed. At time around 5.0 sec., the driver releases the joystick, and the automatic slow down/stop control algorithm is activated. We can see that the desired velocity is well tracked with the error less than about 0.2 m/sec. The desired velocity (dotted line in Figure 7(a)) exists only when the control algorithm equation (8) is active, because velocity tracking is tried only in control algorithm (8). The control algorithm (8) is active when the driver released the joystick and the vehicle speed is greater than 0.16 m/sec.. The same is true for Figures 8, 9, 10. The actual velocity of vehicle is calculated by averaging the four wheel speed information. Figure 7(b) shows the user joystick command before 5.0 sec., and computed joystick command after 5.0 sec. The negative computed command does not change after 10 second, and is used to generate negative torque in the motors.

Figure 8(a), and (b) show similar results for the uphill case. In this case, a positive joystick command is maintained to make the vehicle stop on the up hill.

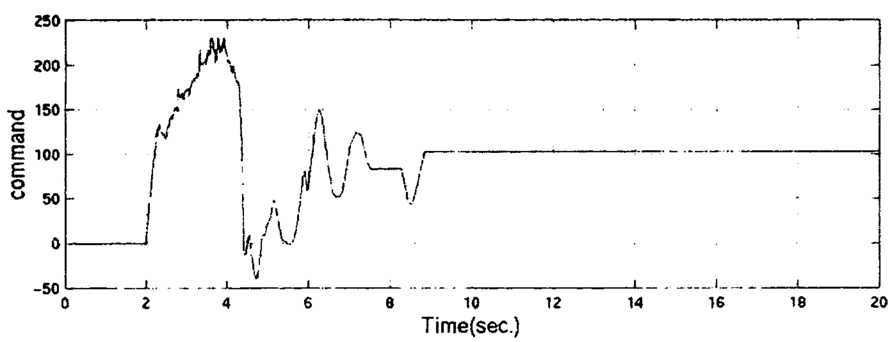
4.2 Emergency collision avoidance control (ECAC)

Two experiments have been carried out. One is the case when a pedestrian is in the middle of the road with Cycab approaching to the pedestrian. The other is the case when a pedestrian is crossing the road in front of Cycab. In both cases, the driver commanded the maximum joystick value to realize dangerous situations.

Figure 9 shows the experimental results of the first case. In (a), the measured distance between Cycab and the pedestrian is shown. Since the detection limit of the ultra-sonic sensor is up to 10 m, and reliable range is up to 9 m, any data

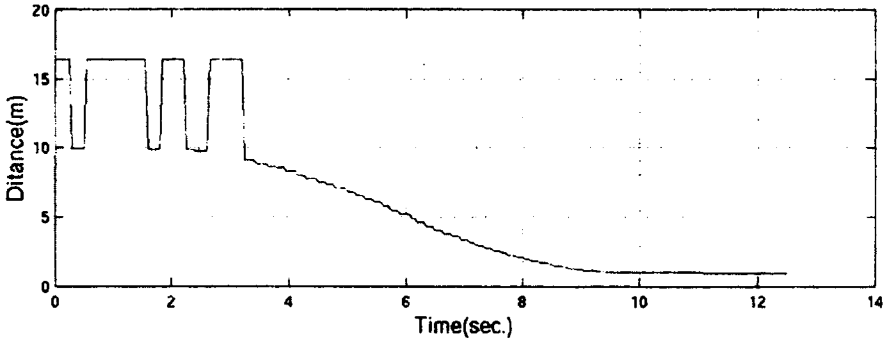


(a) Desired velocity and actual velocity

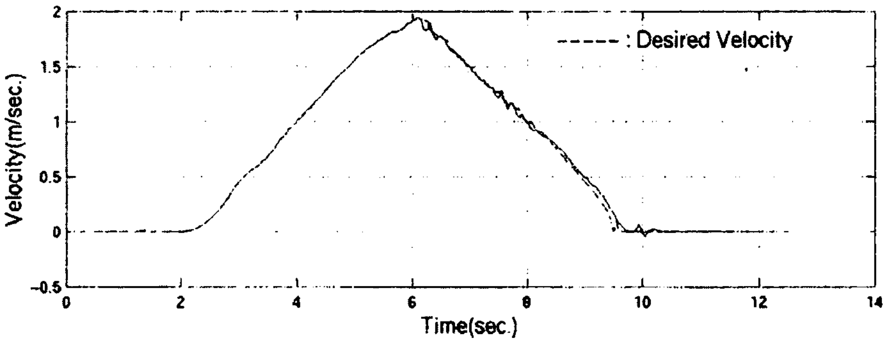


(b) Human joystick or computed command

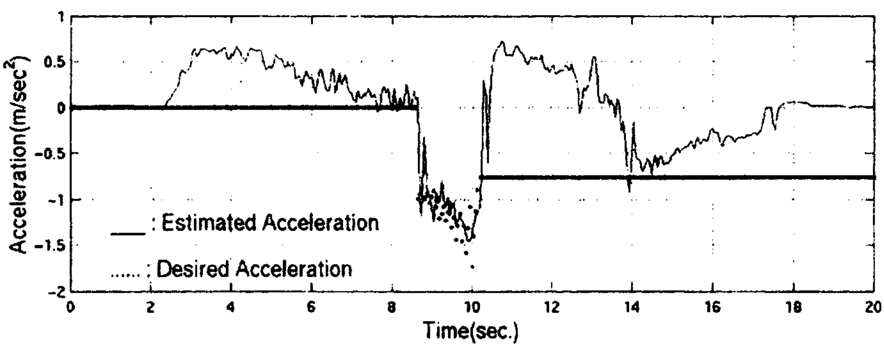
Fig. 8 Automatic slow down and stop on an uphill



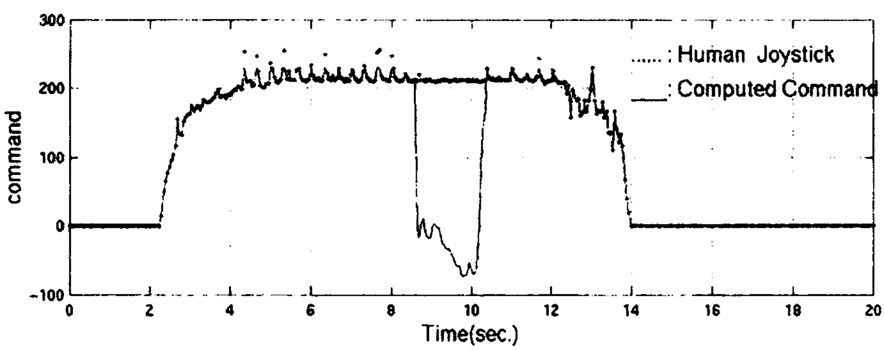
(a) Distance to obstacle measured from sonar sensor



(b) Desired velocity and actual velocity

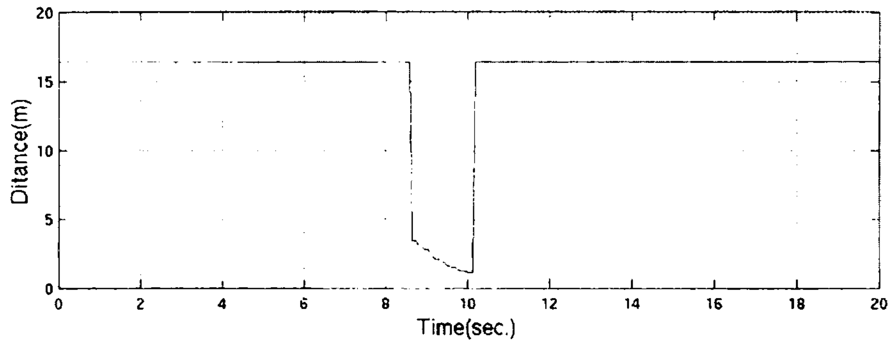


(c) Desired acceleration and estimated acceleration

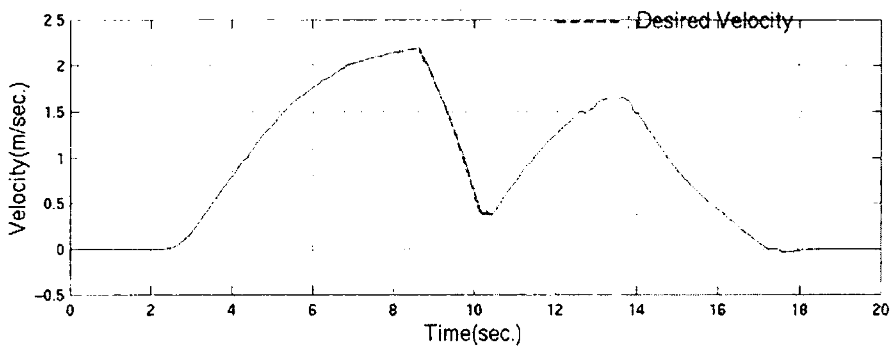


(d) Human joystick or computed command

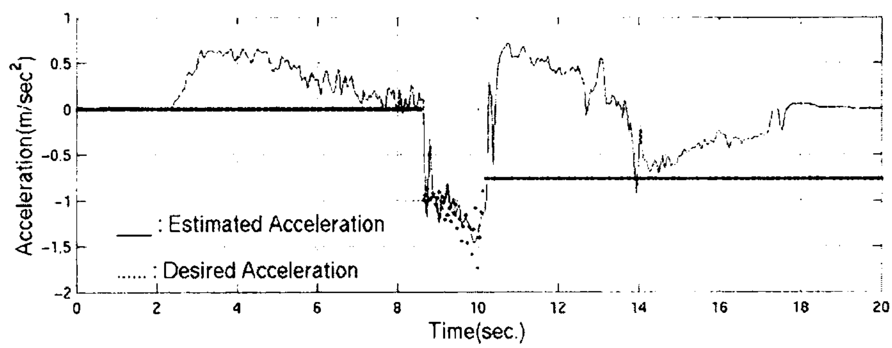
Fig. 9 Collision avoidance stop for pedestrian standing in the middle of road



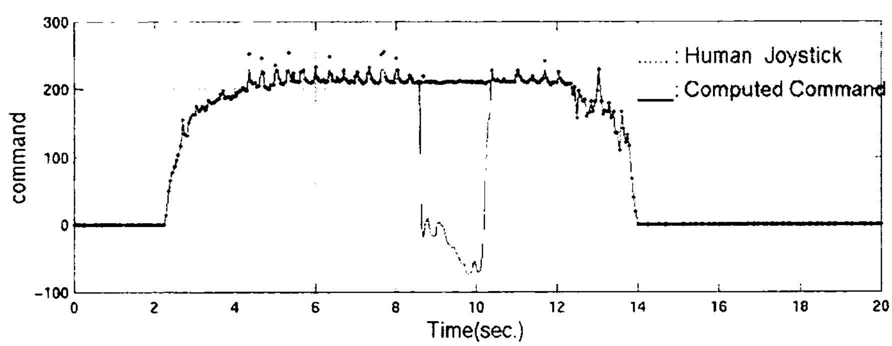
(a) Distance to obstacle measured from sonar sensor



(b) Desired velocity and actual velocity



(c) Desired acceleration and estimated acceleration



(d) Human joystick or computed command

Fig. 10 Collision avoidance stop for a pedestrian crossing larger than 9 m should be ignored. In (b), the desired velocity and actual velocity are shown. The ultrasonic sensor continuously detects the pedestrian when Cycab approaches within about 9.0 m. The control algorithm, however, is activated at 6.0 second, since the danger condition is not met before 6.0 second. After, 6.0 second, the speed reduced linearly following the desired speed history within the error bound of 0.2 m/sec., even though the human joystick command is maximum positive value as shown in (d). Finally, at 10 second, Cycab stops in front of the pedestrian with the spacing of 1.1 m. The desired spacing is 1.0 m. Several experimental results showed that the final spacing errors are less than 0.2 m, when the vehicle power is enough for emergency stopping. In (c), the desired and estimated accelerations are shown. Between 6.0 and 8.2 second, the acceleration shows some chattering around the desired deceleration, which is due to the exchange of controlled command and the human command along the danger and safe region border. The estimated acceleration can be more spiky than the actual one because the acceleration is estimated from the differentiation and low pass filtering of the possibly noisy vehicle speed signal. The actual riding quality is acceptable, and drivers do not feel any uncomfortable jerks. Between 8.2 to 10 second, the operation is inside the danger region, and the command is not chattering.

Figure 10 shows the pedestrian crossing case results. In this case, the sensor detects the crossing pedestrian at 8.5 second. At this initial detection instant, the distance to obstacle is 3.2 m. From that time on, the vehicle slows down its speed quickly until 10.3 second. After 10.3 second, the pedestrian finishes crossing, and the sensor does not detect any obstacle, and the vehicle velocity is increasing because of the maximum positive human command in (d). In (c), we can see the controller makes the actual acceleration follow the desired one.

5. Summary and Conclusions

In this study, two automatic stop control algorithms for an electric vehicle driven by a joystick have been developed and implemented. The developed algorithms are as follows.

(1) An automatic slowing down and stop algorithm on hills using vehicle speed feedback (NSC)

(2) An emergency slowing down and stop algorithm for possible collision situations utilizing ultrasonic sensors (ECAC)

In addition to those, a motor torque distribution algorithm has been developed, which can compensate the motor back EMF drop and can realize power matching in the left and the right wheel motors.

The feedback speed control laws in algorithms (1) and (2) are developed without using longitudinal vehicle dynamic equations. Instead, a simple integral type control with velocity and acceleration feedback structure is used. For the acceleration feedback, vehicle acceleration is estimated from the wheel speed measurements.

Experimental results show the usefulness of the developed algorithms (1) and (2). Experimental results of NSC show that the controller makes the vehicle speed follow the linearly decaying desired speed trajectories within error bound of 0.2 m/sec. Also, NSC makes the vehicle stop without any noticeable jerk or oscillatory movements on hills.

Experimental results of ECAC show that the controller also makes the vehicle stop in front of the obstacle with less than 20 cm error of spacing between the vehicle and obstacles, when the vehicle power is sufficient in emergency stopping. Also, the desire vehicle speed is well tracked as in NSC. During emergency stopping, the riding quality is acceptable, and drivers do not feel uncomfortable jerks. In this study, only longitudinal movement of vehicle is controlled. In future study, lateral movement control algorithms for collision avoidance can be developed and integrated into ECAC.

Acknowledgment

The author expresses sincere thanks to Dr. Michel Parent in INRIA, France and Dr. Thierry

Grandpierre for their helpful advices. Also, the author expresses thanks to Gauthier Fleutot for the ultrasonic sensor setup.

References

Grandpierre, T., Lavarenne, C. and Sorel, Y., 1999, "Optimized Rapid Prototyping for Real-Time Embedded Heterogeneous Mutiprocessors," CODES'99 7th International Workshop on Hardware/Software Co-Design, Rome.

Lawrenz, W., 1997 *CAN System Engineering*, Springe.

Lisowski, L. and Parent, M., 1997 "Specifications of a Small Electric Vehicle : Modular and Distributed Hardware Approach," IEEE/RSJ Int. Conf. On Intelligent Robots and Systems, Grenoble, France

Parent, M., 2000, "Cybernetic Cars for a New Transportation System in the Cities," Proposal for a Fifth European Framework Programme of Information Society Technologies.

Wong, J. Y., 1993, *Theory of Ground vehicles.*, John Wiley & Sons, inc.