A Partitioned Control Scheme for Mobile Robot Path Tracking

Dong Hun Shin, Sanjiv Singh, and Wenfan Shi

Robotics Institute
Carnegie Mellon University
Pittsburgh PA 15213

ABSTRACT

We have suggested a novel approach to autonomously navigate a full sized autonomous vehicle that separately treats vehicle control and obstacle detection. In this paper we discuss the vehicle control that has enabled our autonomous vehicle to travel at speeds upto 35 kmph. We point out the limitations of existing schemes that restrict their consideration to kinematic models and show that it is possible to obtain an increase in performance through the use of approximate dynamical models that capture first-order effects. Our approach combines such a modeling philosophy with accurate feedback in world coordinates from sensors that have only recently become available. Experimental results of our implementation on NavLab, a modified van at CMU, are presented.

1. INTRODUCTION

Recently, inertial sensors and satellite positioning systems have emerged as accurate devices with high data rates, providing position sensing data outdoors and are now widely used for aircraft and ship navigation. These sensors make it possible for autonomous vehicles to obtain rapid position feedback in world coordinates during outdoor operation. This is particularly useful in scenarios where it is not possible to distinguish explicit features such as road edges that can be tracked. We have proposed a paradigm for autonomous robot navigation that separately considers the issues of vehicle control and obstacle detection. In our scenario, an explicit path composed of a sequence of position tags is given to the vehicle. The vehicle travels along this path till it finds an obstacle. In the case an obstacle is found, the vehicle stops and replans its path around the obstacle. In this paper we discuss the issues of vehicle control that enable the vehicle to travel along the specified path at high speeds. Obstacle detection, the process that is responsible for bringing the vehicle to a halt upon detection of an obstacle is separately discussed in [1], while obstacle avoidance, the process of planning a path around the obstacle is discussed in [2].

Among existing path tracking approaches, some methods [3, 4] continuously generate paths that converge to the reference path from the deviated vehicle position; the generated paths are converted into steering angles and wheel velocities using simple vehicle kinematics. Other methods [5, 6] obtain steering angles by multiplying gains to vehicle heading and position errors. These gains are chosen by trial and error until satisfactory results are produced. In pursuing a dynamic model based approach, Muir [7] has developed a version of Newton-Euler recursive formulation of manipulator dynamics for the dynamic modeling of mobile robots. However, full-sized mobile robots—computer controlled cars, buses, and trucks—contain a number of subsystems, such as hydraulic power steering, geared transmission, combustion engines, and tires, which exhibit non-linear and time-varying behavior of different types. Hence, it is difficult to justify such a model-based scheme that treats the robot as a chain of rigid bodies as is commonplace in the treatment of manipulators. In addition, the non-holonomic constraints of wheeled vehicles make the equations of motion very complex, and it is very difficult to accurately model the interaction between the ground and the vehicle[8]. Dickmanns[9] has developed a visual feedback scheme to guide a vehicle on well-structured roads. He successfully modeled the behavior of the position and heading errors relative to the road edge, and then compensated for it through a gain-scheduling method.

We present a simpler control scheme that succeeds through the use of an approximate first order model of the complete dynamical behavior of the vehicle including latency in the actuators, computation time, and communication delays. The proposed scheme further separates the steering and velocity control of the vehicle through a judicious choice of a guide point, the point on the vehicle that is guided over the reference path. We are thus able to reduce the navigation problem to one of planning steering motions that will keep the vehicle on the specified path. Vehicle speed is independently decided by several factors like curvature of the path, and proximity to possible obstacles. We circumvent the explicit modeling of vehicle dynamics and the interaction with the ground by identifying first order effects that can be obtained through relatively simple experimentation. This model of the vehicle dynamics is incorporated in a feedfor-

Figure 1: NavLab, a Testbed for Autonomous Navigation

CH3051-0/91/0000-0338 $1.00

ward module which is combined with a feedback scheme to select steering motions at a subsecond interval. We have implemented this methodology on the NavLab [10], a testbed vehicle at Carnegie-Mellon University, and have been able to obtain autonomous navigation at speeds upto 35 kmph.

2. PROBLEM FORMULATION

2.1 Problem reduction

Given a list of postures $P_i(x_i, y_i, \theta_i)$ specified in world coordinates, we need a control strategy that will allow the vehicle to attain the P_i with bounded error. The following characteristics of conventionally steered (as opposed to skid steered and omni-directional) vehicle are noteworthy:

- In such mechanisms the relationship between controller coordinates (the steering angle, propulsion wheel angle) and controlled coordinates (vehicle position, heading) is described by non-exact differential equations. Thus the vehicle position error cannot be compensated only by the feedback in controller coordinates. For the robot to achieve specified positions and orientations, feedback must be in global coordinates.
- If the center of the rear axle in chosen as the guide point, the steering and propulsion control can be decoupled. This is because under this choice of guide point, the path curvature produced is purely dictated by the steering angle. The main advantage of choosing this convention is that vehicle control can be reduced to steering planning. The speed of the vehicle can be chosen based on other constraints [10].

We capitalize on the above characteristics by:

- Using nested control loops. Since feedback in world coordinates is available at low rates (10-20 Hz), we use it to compute reference inputs to actuators which are then servoed at a millisecond level. This puts an emphasis on real time command generation and relies on already mature servo technology.
- Formulating the problem as path tracking rather than trajectory tracking. Since we can decouple the steering and propulsion control, our method tracks paths rather than trajectories (time history of position); a simpler and more faster method of control.

2.2 Preview control and partitioned scheme

Typically a tracking problem is solved by first formulating it as a regulator problem and by changing the reference inputs as the desired output changes. Such a scheme uses feedback to compensate for errors. However, feedback control can only compensate for errors after the fact. In some case (as in ours) future desired states are known before the fact and this knowledge can be used to enhance performance. Hence, we have added a capability that is akin to what is referred to as preview control. The steering planner is now formulated in two parts:

- *feedforward compensator*, which uses a model of the vehicle dynamics and future path of the vehicle to make anticipating adjustments.
- *feedback compensator*, which provides closed-loop compensation for errors between the vehicle's actual path and its desired path;

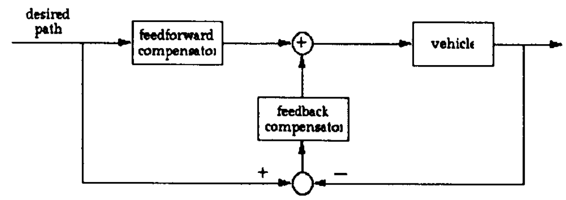

Figure 2: Partitioned Scheme Using Both Feedforward and Feedback Compensation Scheme

3. Proposed Method

3.1 Feedforward compensator

The behavior of a mobile robot can be anticipated by using an approximate dynamic model of the system. Since vehicle dynamics are too complicated to model and to compensate for the dynamics exactly, we have developed a scheme that compensates for the most dominant characteristic behavior of the system dynamics—latency. Since this phenomenon can be described by a time constant in a first-order lag system, the entire steering system is modeled as a first-order lag.

The effect of the feedforward compensation can be seen in Figures 3, 4, and 5. Figure 3 shows the response (solid lines) of the steering system to given reference commands (dashed lines). The reference commands correspond to a path composed of a straight line, a circular arc, and a straight line. The reference and resultant paths in this case are shown as dashed lines and long-dashed lines in Figure 5.

Feedforward compensation is accomplished by sending reference commands in advance. In this manner, the steering starts moving before the arc is reached. A useful amount of advance is the time constant of the first-order model. If the vehicle dynamics are modeled exactly by a first-order lag, then the steering reaches 63% of the desired value by the time the arc path transition is encountered. An intuitive argument can be made that such a scheme produces steering corrections that compensate for the inherent latency of the entire system. The solid line of Figure 5 shows the improvement produced

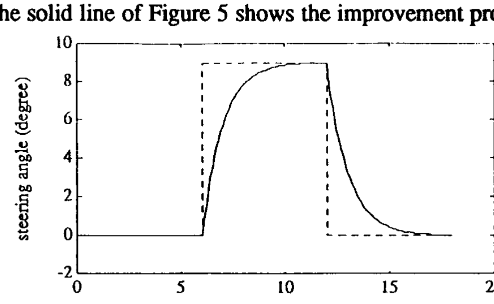

Figure 3: Response of the Steering Dynamics, in Response to a Path of Straight Lines and an Arc.

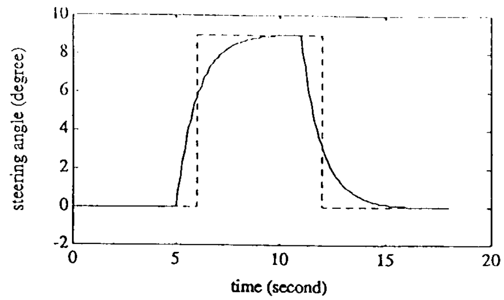

Figure 4: Compensated Steering Performance Modeled as a First Order Model

by feedforward compensation, compared to the open-loop response (long-dashed line of Figure 5).

3.2 Feedback compensator

The feedback compensator guides a vehicle by closed-loop compensation for instantaneous deviation from the prescribed path. At every control cycle the proposed scheme replans a simple, continuous path that converges to a desired path at some look-ahead distance. It also computes a steering angle corresponding to the part of the replanned path, which will be followed for the next time interval.

At any given moment, the vehicle will have errors in lateral position, heading, and curvature. If the vehicle is to be brought back onto the specified path within distance L (measured along the reference path), six boundary conditions can be stated corresponding to the initial errors (x, y, θ) and to zero errors after lookahead distance L. These boundary conditions can be sufficiently satisfied by a polynomial of fifth degree. A quintic polynomial can be constructed to describe the replanned path in error space as a function of s, the distance along the reference path upto L:

$$\varepsilon(s) = a_0 + a_1 s + a_2 s^2 + a_3 s^3 + a_4 s^4 + a_5 s^5 \qquad (1)$$

The expression for e(s) gives the error along the path and its second derivative is converted into the steering angle variation along the lookahead distance L:

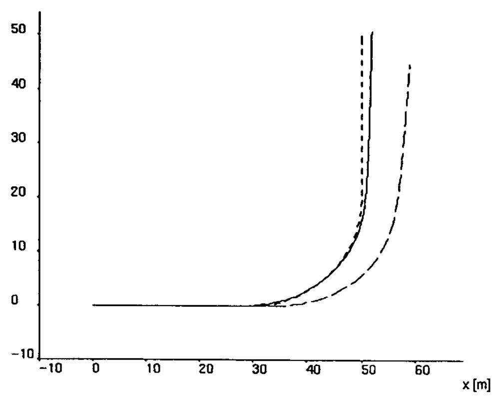

Figure 5: Open-loop Response to a Path of Straight Lines and an Arc

$$\phi_{feedback}(s) = \operatorname{atan}\left\{\frac{d^2\varepsilon(s)}{ds^2} l\right\} \qquad (2)$$

Feedback compensation is as simple as computing the 5 coefficients of the fifth order polynomial in Equation (1). Greater detail about the feedback compensation scheme can be found in [4].

3.3 Speed planning

As mentioned in Section 2.1, one of the benefits of choosing the guide point to be the center of the rear axle is that steering and propulsion can be controlled separately. We simply choose the minimum of the speed dictated by the following constraints: maximum speed limit in a particular area, distance to possible obstacles and destination, and maximum allowable lateral acceleration.

4. Results

4.1 Implementation

To verify and evaluate the performance of path tracking algorithm, the algorithm was applied to vehicle navigation, both in simulations and experiments. In computer simulations a four wheeled vehicle with kinematic and dynamic constraints is simulated. The steering and propulsion are characterized as first order lags with hard velocity and acceleration constraints and a pure time delay.

Our experiments were conducted on the NavLab, a modified truck (Figure 1) which is equipped with onboard computing and inertial sensors that provide explicit position information at 20 Hz.

4.2 Results

Four simulations show the performance of the schemes discussed:

- The vehicle is made to follow a path in open-loop fashion. In this case only the steering angle is regulated, while vehicle position errors are not compensated (Figure 6-A).
- Vehicle position errors are compensated using only the feedback scheme discussed in Section 3.2 (Figure 6-B).
- The only compensation made is a feedforward one, using the control scheme discussed in Section 3.1 (Figure 6-C).
- Both feedback and feedforward schemes are applied to compensate for the vehicle errors (Figure 6-D).

In the case of feedback only, the vehicle oscillates around the nominal path because the feedback gain makes the vehicle sensitive to position errors. For lower value of feedback gain (longer look-ahead distance), much less overshoot is noticed, but the feedback response is very sluggish. In the feedforward only case, the algorithm was able to compensate for the vehicle in a predictive fashion. However, such errors accumulate without feedback in the general case. In the last case, it can be seen that the combined feedback and feedforward schemes complement each other and the resultant path is qualitatively superior to those obtained by application of the other methods.

Experiments on the Navlab have shown results similar to those in simulations. Figures 7-A, B, C, and D show graphical results from real time navigation of the NavLab with variations of feedforward compensating time. The desired path consists of a 20 *m* straight line, a sudden jump by 5 *m* in lateral direction, and an 80 *m* straight line from the initial offset. The vehicle shows serious oscillation with only feedback compensation, as in Figure 7-A, which is similar to the simulation result of Figure 6-B. When the feedforward compensation time was 0.4 seconds (Figure 7-C) or 0.6 seconds (Figure 7-D), the experimental results showed the best tracking performance. These figures correspond to the inherent latency of 0.5 seconds in the control of NavLab. The look-ahead distance in the feedback compensator was fixed at 15 *m*. Evidently, the feedforward compensation has a dominant effect on the tracking performance, as the simulation and experimental results (Figure 6-D and Figure 7-C) show dramatic improvement from those without feedforward compensation.

Figure 8 presents the tracking performance of an arbitrarily recorded path that is approximately 500*m* long. Note that the resultant path is deviated from the reference path towards the end of the course. This is attributed to a heading drift in the inertial position sensor which got progressively worse unless calibrated periodically. The reference path in Figure 8 was found to have serious discontinuous jumps in heading and curvature, obtained when the reference path was recorded. Even with a physically unrealizable reference path, the vehicle was able to converge to the desired path.

5. Conclusion

A partitioned scheme for steering planning has been presented. It consists of a feedforward compensator and a feedback compensator. The former guides a robot along an intended path by producing, with the use of *a priori* knowledge of the future path, an anticipatory control; and the latter compensates for errors by finding a curve which will converge with the desired path while insuring a smooth reduction of all errors. The proposed method succeeds with a first order

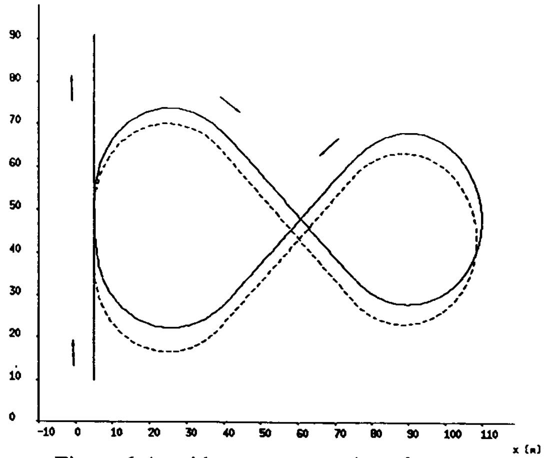

Figure 6-A: without compensation of errors

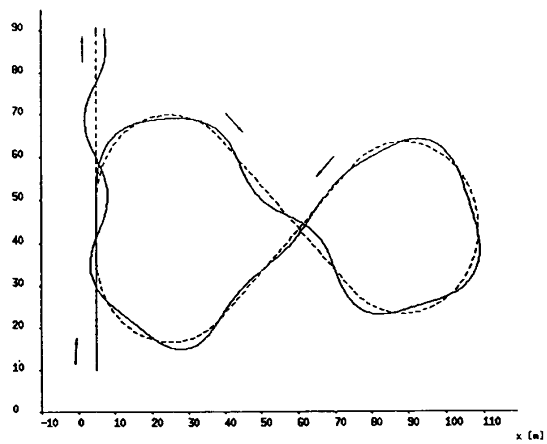

Figure 6-B: with only feedback compensation

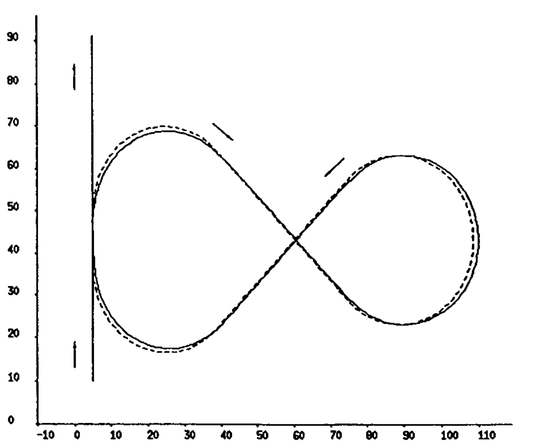

Figure 6-C: with only feedforward compensation

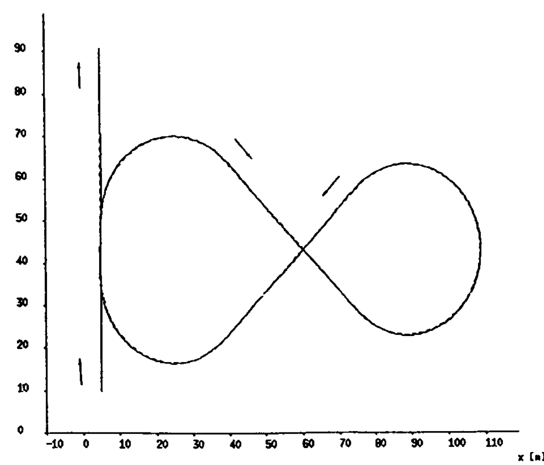

Figure 6-D: with both types of compensation

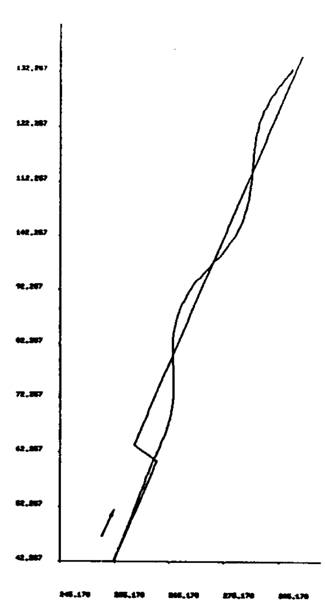

Figure 7-A: 0 second

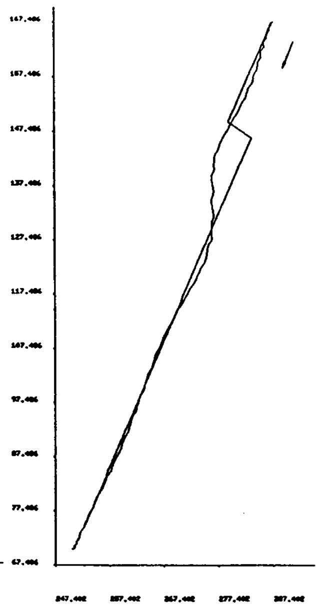

Figure 7-B: 0.2 second

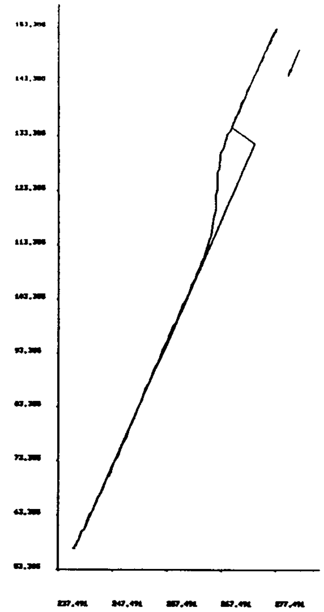

Figure 7-C: 0.4 second

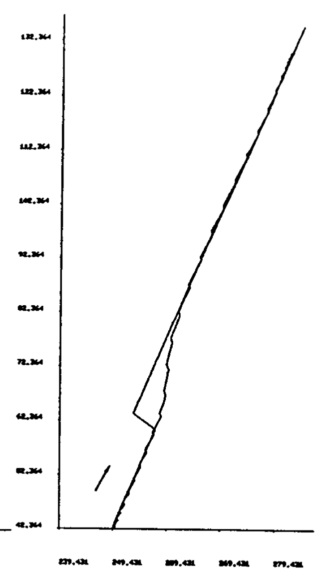

Figure 7-D: 0.6 second

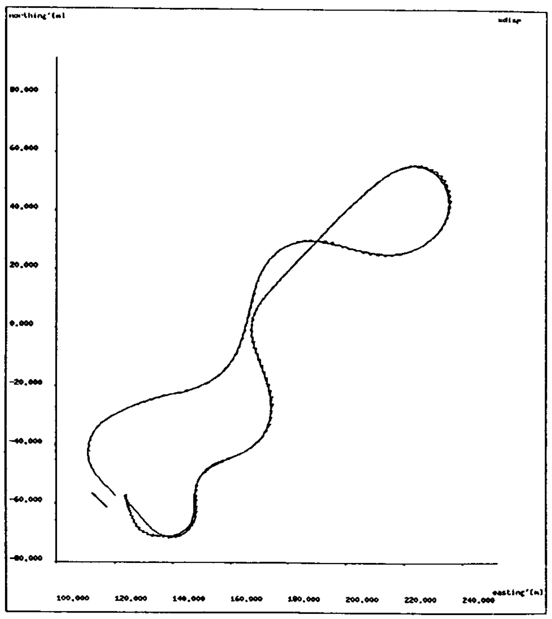

Figure 8: Real-time vehicle navigation

model by compensating for the total system latency. This circumvents the problem of complex physics in modeling and control and yet provides high performance through smooth error compensation and anticipatory control. These developments were then implemented on simulation and navigation hardware. Tracking performance was significantly improved with the addition of the feedforward scheme as vehicle speed increased.

References

1. S. Singh, P. Keller, "Obstacle Detection for High Speed Autonomous Navigation," In Proc. IEEE Conference on Robotics and Automation, April 1991, Sacramento.
2. D. Feng, S. Singh, B. H. Krogh, "Implementation of Dynamic Obstacle Avoidance on the CMU NavLab," In Proc. IEEE Conference on Systems Engineering, August 1990, Pittsburgh.
3. R. Wallace, et al., "First Results in Road Following," *Proceedings of IJCAI*, Los Angeles, August, 1985.
4. D. H. Shin and S. Singh, "Position Based Path Tracking for Wheeled Mobile Robots," *Proceedings IEEE/RSJ International Workshop on Intelligent Robots and Systems '89*, pp 386-391
5. Y. Kanayama, A. Nilipour and C. A. Lelm, "A Locomotion Control Method for Autonomous Vehicles," *IEEE Intl. Conference on Robotics and Automation*, Philadelphia, 1988, pp1315-1317.
6. W. L. Nelson and I. J. Cox, "Local Path Control for an Autonomous Vehicle," *IEEE Intl. Conference on Robotics and Automation*, Philadelphia, 1988, pp1504-1510.
7. P. F. Muir, *Modeling and Control of Wheeled Mobile Robots*, Ph.D. Thesis, Carnegie-Mellon University, 1988.
8. H. Dugoff, P. S. Fancher, and L. Segel, "An Analysis of Tire Traction Properties and Their Influence on Vehicle Dynamic Performance," *SAE paper* 700377, 1970.
9. E. D. Dickmanns, and A. Zapp, "Autonomous High Speed Road Vehicle Guidance by Computer Vision," *IFAC-Congress 1987*, Munich, PP232-237.

10 K. Dowling, et al., "NavLab: An Autonomous Navigation Testbed," Technical Report, Carnegie Mellon University, CMU-RI-TR-87-24.