FRACTIONAL CONTROLLER FOR GUIDANCE OF AUTONOMOUS GROUND VEHICLES

A. Rodríguez-Castaño [1]
A. Ollero [1]
B.M. Vinagre [2]
Y.Q. Chen [3]

[1] *University of Seville, Seville – Spain*
[2] *University of Extremadura, Badajoz – Spain*
[3] *Utah State University, CSOIS, Logan – U.S.A.*

Abstract: This paper investigates the use of Fractional Order Control (FOC) in path tracking problems. A fractional controller has been simulated to analyze the performance for changes in navigation speed. This controller is compared to a typical Pure-Pursuit technique. Experiments with the ROMEO 4R autonomous vehicle have been carried out using the fractional controller.

Keywords: autonomous vehicles, position control, controllers implementation.

1. INTRODUCTION

Autonomous guidance of ground vehicles has been a research and development topic in the last 20 years. A lot of methods have been proposed to accomplish this, like *pure-pursuit*, fuzzy control, LQG/LQR techniques and generalized predictive control (Ollero, 2001; Elkaim, *et al.*, 1997).

In this paper, a method, called the ε-controller and proposed by Davidson and Bahl (2001), is used. This controller computes the normal distance from the vehicle to the desired path, ε, and generates a desired velocity vector, so the vehicle can follow the path. A modified version of this controller is used in this paper: only the direction of the vector is used, while the navigation speed remains constant.

Suarez, *et al.*, (2003) have pointed out the use of FOC controllers to improve the ε-controller performance. FOC has been introduced in the last decades for managing a variety of control problems. First experiences in path-tracking applied to XY cutting tables and mobile robotics can be found in (Orsoni, *et al.*, 2001; Orsoni, *et al.*, 2002).

In this paper we present preliminary simulation and experimentation results of the use of FOC in the path-tracking problem and the application to the ROMEO 4R autonomous vehicle.

This paper is organized as follows. Section 2 briefly describes the dynamic model of the vehicle. Section 3 reviews the modified ε-controller. In section 4 the fractional order controller (FOC) is presented. Simulation and experimental results are discussed in detail in sections 5 and 6 respectively. Finally, in section 7 some conclusions and guidelines for future works are stated.

2. VEHICLE DYNAMIC MODEL

The vehicle used in our work is the ROMEO 4R autonomous vehicle (see Figures 6 and 7), with Ackerman steering system (Bahl, 2002). The vehicle model is a typical "bicycle" kinematic model (see Figure 1), adding a first order dynamic for the steering subsystem and a first order dynamic for the traction subsystem as can be seen in (1):

$$\begin{aligned}
\dot{x} &= -v\sin\theta \\
\dot{y} &= v\cos\theta \\
\dot{\theta} &= v\gamma \\
\dot{\gamma} &= -\frac{1}{T_\gamma}(\gamma - \gamma_d) \\
\dot{v} &= -\frac{1}{T_v}(v - v_d) \\
\gamma &= \frac{1}{R}
\end{aligned} \tag{1}$$

Parameters for the ROMEO 4R used in our simulations are T_γ=1 second and T_v=1.5 seconds.

The coordinate system used can be seen in Figure 1, where θ is positive anti-clockwise and γ is positive when the vehicle turn left. (X_G,Y_G) is a global coordinate system (GCS), (X_B,Y_B) is a body-fixed coordinate system (BFCS) attached to the vehicle, and (x,y) are the coordinates of the BFCS in the global coordinate system.

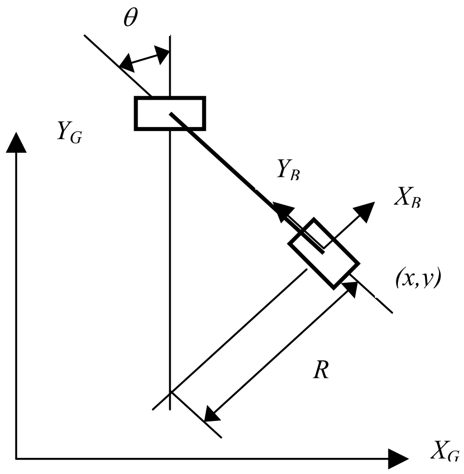

Fig. 1. Coordinate systems and kinematic model.

3. PATH-TRACKING ALGORITHM

The path tracking problem was accomplished by using the scalar ε-controller (Davidson and Bahl, 2001). Basically, it is a regulator that operates on the vehicle normal deviation ε from the desired path. The ε-controller (C_ε) generates a desired velocity vector $\mathbf{V_I}^*$ which depends on the lateral deviation from the path. This, $\mathbf{V_I}^*$, vector has two components: one tangent to the path (V_t) and other normal to the path (V_n). If the vehicle is near the path, the ε-controller increases V_t and decreases V_n, allowing the vehicle to get closer to its maximum speed. However, when the vehicle is far from the desired path, V_n is increased and V_t is decreased, reducing the speed of the vehicle.

The *MakeSetPoint* (MSP) algorithm converts the desired velocity vector $\mathbf{V_I}^*$ into steering angle setpoint. This is accomplished by rotating the velocity vector, given in a Global Coordinate System (GCS), into the vehicle-fixed coordinate system. Then, the steering angle setpoint is obtained from the vehicle geometry. A detailed description of the algorithm can be found in (Davidson and Bahl, 2001: Bahl, 2002).

The ε-controller used in this paper is a modified version of the one explained above (Davidson and Bahl, 2001). It should be noted that when the vector $\mathbf{V_I}^*$ is rotated into the vehicle coordinate system (BFCS) the y component of this rotation can be close to zero ($\mathbf{V_I}^*$ perpendicular to Y_B axis). This y component is chosen as the navigation speed, so the vehicle could stop. To avoid this problem, only the direction of $\mathbf{V_I}^*$ is used. Then, there is no change in speed, the vehicle moves at constant speed V_d.

4. C_ε - CONTROLLER REGULATION SCHEME

The control law of the ε-controller turns the two-dimensional path tracking problem into a scalar regulation. The controller tuned and tested in this paper is the following:

$$u(t) = K_P\,\varepsilon(t) + K_I\, I^{\alpha}\varepsilon(t) \tag{2}$$

being

$$I^{\alpha}\varepsilon(t) \equiv \frac{1}{\Gamma(\alpha)}\int_0^t (t-\tau)^{\alpha-1}\varepsilon(\tau)d\tau,\ t>0, \alpha\in\Re^+ \tag{3}$$

We have chosen α=0.5 and hence we have obtained a fractional controller. This value have been selected by iterative trial and error simulation. This controller has been implemented digitally, with a sample period T=0.1 seconds. The fractional integral ($1/s^{\alpha}$) term has been discretized using a finite dimensional IIR filter model. The continuous fractional expansion (CFE) of the Tustin transformation (see Equation 4) has been used.

$$s = \frac{2}{T}\frac{1-z^{-1}}{1+z^{-1}} \tag{4}$$

For our case (see Vinagre, *et al.*, 2000), with α=0.5 the aproximation for the fractional part of the integrator is:

$$\frac{1}{s^{0.5}} \approx \frac{0.22z^5 + 0.11z^4 - 0.22z^3 - 0.08z^2 + 0.04z^1 + 0.007}{z^5 - 0.5z^4 - z^3 + 0.375z^2 + 0.1875z^1 - 0.031} \tag{5}$$

Parameters K_P and K_I are 0.05 and 0.03 respectively.

5. SIMULATIONS

5.1 Simulation scenario.

The simulations show how the vehicle approach a straight line. The vehicle starts at position x=4 meters, y=0 meters, θ=0 radians, and tries to follow the Y_G axis (see Figure 2).

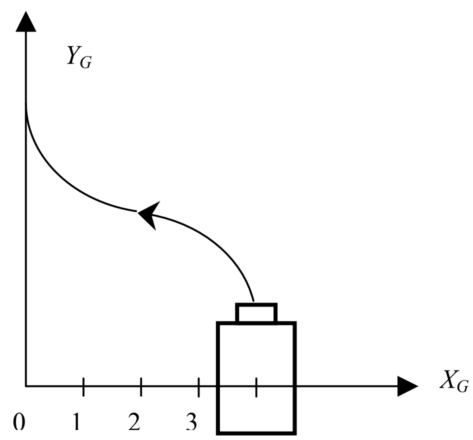

Fig. 2. Reference path.

System response for different constant speed (V=1, 2, 3 m/s) of the vehicle have been studied, comparing how the vehicle approach the path.

In the following, the proposed controller is compared with a typical Pure-Pursuit (PP) technique (Ollero, 2001). This controller has been tuned by simulation for 1 m/s speed, selecting a constant Lookahead parameter (L=6 meters).

5.2 Simulink model.

The Simulink model shown in Figure 3 simulates the control arquitecture described in Section 3. In this case (see Figure 3), the normal deviation ε to the path is the x coordinate of the vehicle. The velocity vector has two components: tangent and normal component to the path. The tangent component is parallel to the Y_G axis, and the normal component is parallel to the X_G axis. As described, the MSP block converts the velocity components to a curvature (steering angle) setpoint, while the navigation speed (V_d) remains constant. Both commands are applied to the ROMEO 4R model described in Section 2. Then, the PI controller uses the lateral deviation to generate a normal velocity component. Finally, the C_ε block uses the normal velocity component to provide the desired velocity vector to the MSP block. This velocity vector has the two components V_x and V_y.

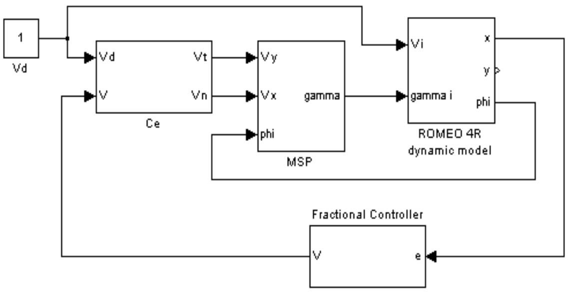

Fig. 3. Simulink model used.

5.3 Simulation results.

As shown in Figure 4, the performance of the PP controller may vary significantly for different speeds. On the other hand, the fractional controller shows a more robust performance for changes in speed (see Figure 5), and there are three parameters (K_P, K_I, α) which provide more flexibility to adapt to different navigation conditions. As can be seen (Figure 5), even though the controller has been tuned for V_d=1 m/s, when the speed is changed to 2 and 3 m/s the vehicle approach to the path quite properly. The system response is slower, but it does not show oscillations. These oscillations appear when the Pure-Pursuit technique is applied (see Figure 4).

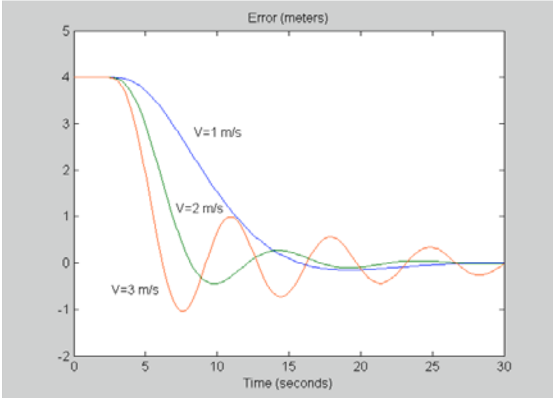

Fig. 4. Pure-Pursuit technique system response.

6. EXPERIMENTS

The fractional controller presented in the above sections has been implemented in the autonomous vehicle ROMEO4R (see Figure 6). ROMEO4R is an electrical golf cart like vehicle with Ackerman steering that is adapted for autonomous navigation. It weights 700 Kg. and is 2 meters long.

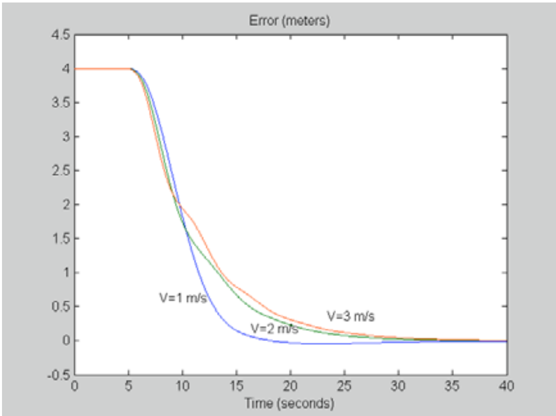

Fig. 5. Fractional controller system response.

Automatic steering has been implemented by using a 80W DC motor that is connected to the steering column through a reduction gear and an electromagnetic clutch. Traction power is achieved by a 2 CV DC motor. Both motors are actuated through a motion control card.

Fig. 6. ROMEO 4R front side.

The vehicle is provided with a wide set of sensors, including a gyroscope and encoders for dead-reckoning estimation, a GPS receiver, a 2D laser range finder, 2 cameras and 10 sonars attached around the vehicle.

The on-board control system is composed of two computers connected by an ethernet link (see Figure 7). An industrial Pentium 133 PC, with 64 Mb RAM memory, and Debian GNU/Linux 2.2.r4 carries out position estimation and low level control, i.e. speed and steering angle. Path tracking is also performed in this computer. The other computer is similar and used for image processing issues

Fig. 7. ROMEO 4R back side.

The fractional controller has been implemented in C code, as the whole ROMEO 4R control system.

As shown in Figure 8, for a 1 m/s speed experiment, the real behaviour is very similar to the one predicted by simulation.

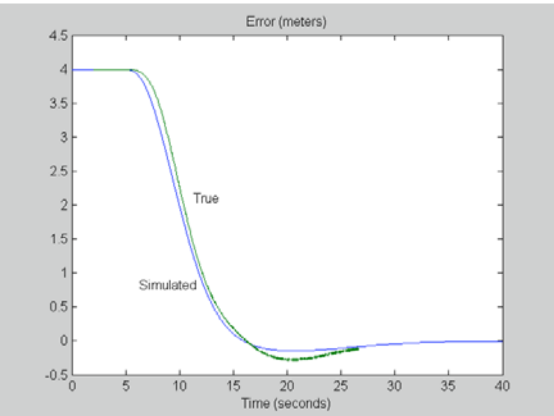

Fig. 8. Simulated vs true error.

7. SUMMARY AND CONCLUSIONS

The fractional controller applied to the ε-controller path tracking technique has shown a good performance for changes in navigation speed, in simulation and in experiments carried out with the ROMEO 4R vehicle. Future work will include tuning methods to design the controller, a more detailed comparison with other path tracking algorithms and extensive experimentation.

ACKNOWLEDGMENT

This work has been supported in part by the CYCIT TAP99-0926-C04-01. The authors would like to thank the students Rafael Martin, Victor Blanco and Jose Victor Acevedo for their efforts and work with the ROMEO 4R vehicle.

REFERENCES

Bahl, V. (2002). *Modeling and Control of a Class of Autonomous Wheeled Mobile Robots*. MSc thesis, Utah State University.

Davidson, M. and V. Bahl (2001). The scalar epsilon-controller: A spatial path tracking approach for ODV, ackerman, and differentially-steered autonomous wheeled mobile robots. *Proceedings of the IEEE International Conference on Robotic and Automation*, pp. 175-180.

Elkaim, G.H., M.L. O'Connor, T. Bell, and B.W. Parkinson (1997). System identification and robust control of farm vehicles using CDGPS. *Proceedings of ION GPS-97*.

Ollero, A. (2001). *Robótica. Manipuladores y Robots Móviles*. Marcombo, Spain.

Orsoni, B., P. Melchior, A. Oustaloup, T. Badie, and G. Robin (2002). Fractional motion control: Application to an xy cutting table. *Nonlinear Dynamics*, Vol. **29**, no. 1-4, pp. 297-314, 2002

Orsoni, B., P. Melchior, and A. Oustaloup. (2001). Davidson-cole transfer function in path tracking design. *Proceedings of the 6th European Control Conference*, pp. 1174-1179, Porto (Portugal).

Suarez, J.I., B.M. Vinagre, and Y.Q. Chen (2003). Spatial Path Tracking of an Autonomous Industrial Vehicle using Fractional Order Controllers. *Submitted to 11th International Conference on Advanced Robotics, ICAR 2003*.

Vinagre, B.M., I. Podlubny, A. Hernández and V. Feliu (2000). Some approximations of fractional order operators used in control theory and applications. *Fractional Calculus and Applied Analysis*, Vol. **3**, no.3, pp. 231-248.