Effective Real-Time Wireless Control of an Autonomous Guided Vehicle

C. Lozoya, P. Martí, M. Velasco, J.M. Fuertes
Dept. Enginyeria de Sistemes, Automàtica i Informàtica Industrial
Universitat Politècnica de Catalunya
Pau Gargallo 5, 08028 Barcelona, Spain
Email: {camilo.lozoya, pau.marti, manel.velasco, josep.m.fuertes}@upc.edu

***Abstract*— Wireless communication systems used in industrial environments must guarantee that the information is sent and received within precise time-bounds. However the nature of the radio channels and the medium access control (MAC) generates random communication delays. For networked control systems, these delays can cause severe performance problems. This paper presents an autonomous guided vehicle (AGV) path tracking control system whose closed-loop consists on a vehicle connected over a wireless network to the controller. To mitigate the negative effects of delays, we propose a Kalman-based network delay estimator in the controller, that provides effective control performance. Our approach is compared to previously proposed statistical estimation algorithms by evaluating the vehicle's travelling time and path deviation.**

I. INTRODUCTION

Industrial applications involving mobile systems can benefit from the use of wireless communication technologies. The deployment of wireless technologies in the industrial environment plays an important role in the simplification of costly and complex infrastructures and also in the improvement of flexibility in the factory. The localization and tracking of components, the coordination of autonomous transport vehicles and mobile robots, as well as applications involving distributed control are all areas in which wireless technologies could be used in an industrial environment [1].

It is known that AGV's are useful for the movement of material and for the productivity of automated systems. Fast AGV's with obstacle avoidance capabilities can maneuver themselves among manufacturing lines carrying components [2]. Usually, AGV control is done locally. An alternate implementation involves a remote control. This schema opens up a full set of new applications such as coordination of tasks in a multi-AGV systems, providing at the end more flexible manufacturing systems.

Based on the benefits provided by the remote control, an adequate architecture for AGV systems may consist in an external controller sending and receiving, through a wireless network, the control commands to the vehicle. However, in a wireless AGV system, a major challenge is the existence of a network delay that may degrade the overall system performance. See [3] for an illustrative analysis of the effect of time delays in the performance of networked control systems.

Wireless Local Area Network (WLAN) based on the IEEE 802.11 standard (commercially known as WiFi), was originally conceived as the wireless extension of Ethernet, and now has become one of the most mature wireless communication network [4]. However, communication features like reliability and timeliness are significantly more difficult to meet in WiFi networks, because of the adverse properties of the radio channels and the characteristics of the medium access of control [5]. The properties of the radio channels like low capacity of the shared radio channel, path loss and the channel errors, cause network delays since data packets often need to be retransmitted. The medium access of control used by WiFi is based on the carrier sense multiple access (CSMA) mechanism, which presents random and long access delays when there is a heavy network traffic load.

Consequently, for time sensitive applications relying on WiFi links such as networked control systems, the characteristics of the radio channels and the medium access control must be taken in consideration and managed properly. As it has been already shown on [6] and [7] for other industrial networks such as CAN or Profibus, taking into account effects or properties of the medium access control in the design of networked applications permits to improve overall control performance.

Different approaches has been proposed in order to manage the negative effects of network delays. In [8], a virtual polling algorithm is implemented in the application layer to improve network performance by reducing data collisions, however this approach does not consider the impact of the delays caused by the nature of the radio channels. In [9], a redundant transmission protocol is described as a mechanism to mitigate the errors due to the unreliable wireless channel, but the delays caused by the medium access control are not taken in consideration. In [10], a gain scheduler middleware is proposed in order to modify the controller output with respect to current network conditions, but the performance of the statistical estimation algorithms can be improved.

The contribution of this paper consists in the use of the Kalman filter algorithm [11] to predict the time delay over a WiFi network. An AGV path tracking control system is used to measure the performance of the estimation algorithm. The performance of the Kalman filter algorithm is compared with previously proposed estimation algorithms. With a correct delay estimation, the remote controller is capable of predicting the position of the vehicle and generate the adequate control commands.

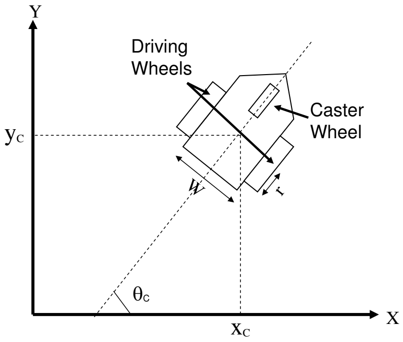

Fig. 1. Schematic of the AGV

In this work a simulation model has been implemented, in order to evaluate the effectiveness of the proposed approach by measuring the vehicle's path deviation and the total travelling time. The simulation results show that an accurate estimation of the network delay allows more precise control.

This work is divided in the following sections: Section II describes the characteristics of the AGV path tracking system. Section III details the Kalman filter algorithm as the delay estimation process. Section IV explains how the simulation model was implemented. Section V describes the set up of the simulation process, and finally in Section VI the simulation results are analyzed.

II. SYSTEM DESCRIPTION

The AGV we considered consists in a differential drive mobile robot with two driving wheels and one caster wheel. As shown in Fig. 1, the AGV current position is given by x_C, y_C and the current posture angle is θ_C. A generalization of the quadratic curve approach proposed in [12] is used as a path tracking algorithm for this control system.

A. AGV Kinematic Model

In the AGV kinematic model, the vehicle states can be calculated from the rotational velocities of the right and left wheels, defined by ω_r and ω_l respectively.

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} \frac{r}{2} & \frac{r}{2} \\ \frac{r}{W} & -\frac{r}{W} \end{bmatrix} \begin{bmatrix} \omega_r \\ \omega_l \end{bmatrix} \tag{1}$$

where v is translational velocity, ω is the rotational velocity, r is the wheel radius, and W represents the distance between the two wheels of the vehicle.

The AGV next position and posture, after a Δt time interval has elapsed, is obtained from the translational and rotational velocities.

$$x_{C(k+1)} = x_{C(k)} + v\Delta t \cos(\theta_{C(k)} + \omega\frac{\Delta t}{2}) \tag{2}$$

$$y_{C(k+1)} = y_{C(k)} + v\Delta t \sin(\theta_{C(k)} + \omega\frac{\Delta t}{2}) \tag{3}$$

$$\theta_{C(k+1)} = \theta_{C(k)} + w\Delta t \tag{4}$$

where Δt represents the sampling time interval.

B. Local Path Tracking Control

The goal of the path tracking control is to drive the vehicle to a desired location or reference point by controlling the speed of the vehicle's right and left wheel in such a way that the deviation from the shortest path is minimized.

A route planner provides to the controller the route information in the form of a set of reference points, using the reference vector $[x_R, y_R, \theta_R]$. The controller is periodically receiving the vehicle current position (x_C, y_C, θ_C). The controller reads from the vector the first reference point compares with the current position, and calculates and sends to the vehicle the require speed of the right and left wheels $(\omega_{R,r}, \omega_{R,l})$. In the next sampling period the controller reads the new vehicle position, then calculates and sends the new speed values. This loop continues until the vehicle reaches the current reference point. After this, the controller reads the next reference point from the vector and starts the loop again. When every reference point has been reached, the vehicle finally completes the route.

The control algorithm is implemented with the following sequence of steps:

1) Obtain the error vector $[e_x, e_y, e_\theta]$ from the difference between the current position and the reference point.

$$\begin{bmatrix} e_x \\ e_y \\ e_\theta \end{bmatrix} = \begin{bmatrix} \cos\theta_C & \sin\theta_C & 0 \\ -\sin\theta_C & \cos\theta_C & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_R - x_C \\ y_R - y_C \\ \theta_R - \theta_C \end{bmatrix} \tag{5}$$

2) Estimate the linear and angular velocities $(\hat{v}_{R(k)}, \hat{\omega}_{R(k)})$, as an intermediate step. For complete information on how to obtain equations (6) and (7) refers to [12].

$$\hat{v}_{R(k)} \simeq K_{(k)} \tag{6}$$

$$\hat{\omega}_{R(k)} \simeq 2A_{(k)}K_{(k)} \tag{7}$$

where

$$A_{(k)} = sign(e_x)\frac{e_y}{{e_x}^2} \tag{8}$$

$$K_{(k)} = sign(e_x)\frac{\alpha}{1 + |A_{(k)}|} \tag{9}$$

and α is a positive constant used as a speed factor. The vehicle will move faster with a higher value of α, and slower with a lower value.

3) Calculate the speed of the right and left wheels that the vehicle requires in order to reach the reference point.

$$\omega_{R,r} = \frac{\hat{v}_{R(k)}}{r} + \frac{W\hat{\omega}_{R(k)}}{2r} \tag{10}$$

$$\omega_{R,l} = \frac{\hat{v}_{R(k)}}{r} - \frac{W\hat{\omega}_{R(k)}}{2r} \tag{11}$$

where $\omega_{R,r}$ is the angular velocity for the right wheel, and $\omega_{R,l}$ is the angular velocity for the left wheel.

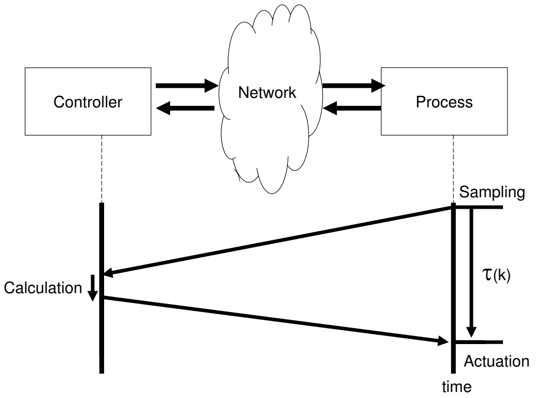

Fig. 2. Round Trip Time Delay.

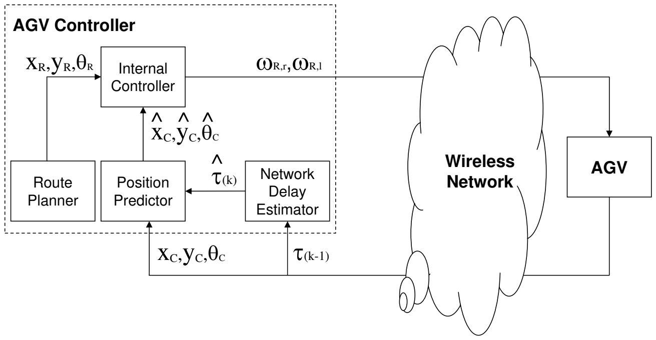

Fig. 3. Dataflow of the Path Tracking Control over a Network.

C. Delay Metric

In order to carry out effective remote path tracking control, we require to measure the network delay. One of the most common metrics in a communication network, in order to quantify the network performance, is the round trip time (RTT) delay which reflects the traffic load [13]. In a networked control system, the RTT delay is defined as the time in between, when a request is transmitted from the process station and when the control data is received. Therefore, as shown in Fig. 2, the RTT delay, symbolized by $\tau_{(k)}$, includes the network delay from sampling to calculation, the calculation time and the network delay from calculation to actuation.

D. Remote Path Tracking Control

In the remote path tracking control scheme it has to be considered that delays may provoke an inconsistent operation of the previous algorithm. If we perform remote path tracking control, the previous algorithm still holds but the estimated vehicle position must be considered instead of the current one. For a local controller, the control algorithm uses the current position. However, since the proposed approach considers a remote controller, the current position received has to be adjusted to account for the wireless network delay.

The proposed AGV path tracking networked control system is based on the schema defined in [10] since a modular structure can be implemented over an existing control system. As a difference, the proposed system includes a delay estimation algorithm based on the Kalman filter.

The sequence of activities executed at each sampling period in the networked control system, illustrated in Fig. 3, is the following:

- In the AGV.
  1) Obtain vehicle current position (x_C, y_C, θ_C).
  2) Calculate the previous RTT delay value $\tau_{(k-1)}$.
  3) Send both values to the remote controller.
- In the controller, after receiving the data sent by the vehicle.
  1) The Route Planner (RP) provides the next reference point (x_R, y_R, θ_R), if the previous one has been already reached.
  2) The Network Delay Estimator (NDE) takes the previous RTT delay value $\tau_{(k-1)}$.
  3) The NDE estimates the next value of the RTT delay $\hat{\tau}_{(k)}$, using the Kalman filter algorithm.
  4) The Position Predictor (PP) uses the estimated RTT delay value $\hat{\tau}_{(k)}$ and the current position of the vehicle (x_C, y_C, θ_C). Since the PP knows the kinematical behavior of the AGV, according with (1)-(4), calculates the estimated position of the vehicle $(\hat{x}_C, \hat{y}_C, \hat{\theta}_C)$.
  5) The PP sends the estimated position to the Internal Controller.
  6) The Internal Controller generates and sends the control signals $(\omega_{R,r}, \omega_{R,l})$ to the AGV.
- In the AGV.
  1) Update wheels velocity upon reception of the control signals.
  2) Wait for the next sampling period.

Since the AGV path tracking control is time sensitive, it requires an accurate estimation of the RTT delay. It is important to notice that the AGV receives the control information under the assumption that the vehicle is in a specific location at time $t + \hat{\tau}_{(k)}$, but the control data is actually received at time $t + \tau_{(k)}$, see Fig 4. The time when the vehicle sends the initial sampling message is represented by t. If the difference between the estimated RTT delay $\hat{\tau}_{(k)}$ and the actual RTT delay $\tau_{(k)}$ is higher, then the performance degradation of the AGV will increase. An accurate estimation of the RTT delay will benefit the AGV performance compared with a schema where the RTT delay is no estimated or the estimation is poor. This is the reason of the importance of an accurate estimation.

III. NETWORK DELAY ESTIMATION

This section describes the characteristics of the Kalman filter algorithm, used in the proposed networked control system, in order to estimate the RTT delay. Statistical algorithms (Mean Value, Median Value, Max Value) to estimate RTT delay have been already proposed in [14]. However Kalman filter presents the advantage of being an algorithm that implements

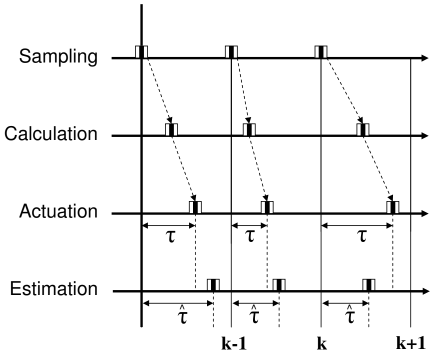

Fig. 4. Estimated RTT delay versus actual RTT delay.

a dynamic closed loop system that does not required to store historical data.

The Kalman filter is a recursive estimator. This means the the estimation of the current state is obtained from the previous estimation and the current measurement.

The Kalman filter has two distinct phases: predictor (time update) and corrector (measurement update). The predictor phase uses the previous estimation to produce the *a priori* estimation of the current state (equations (12) and (13)). In the corrector phase, measurement information from the current state is used to refine the prediction and obtain the *a posteriori* estimation (equations (14), (15) and (16)). The *a posteriori* estimation is used in the next predictor phase.

For the purpose of this algorithm, the estimated RTT delay $\hat{\tau}_{(k)}$, corresponds to the Kalman filter's *a priori* estimation.

In the predictor phase, the *a priori* estimation of the current state is obtained by:

$$\hat{x}^{-}_{(k)} = A\hat{x}_{(k-1)} + Bu_{(k)} \tag{12}$$

where A is a scalar value that represents the constant feedback gain, $\hat{x}_{(k-1)}$ defines the previous *a posteriori* estimate of the process state, B is a scalar value that represents the constant input gain, and $u_{(k)}$ represents the current input.

The *a priori* estimation of the covariance error is given by:

$$P^{-}_{(k)} = A^2 P_{(k-1)} + Q \tag{13}$$

where $P_{(k-1)}$ is the previous *a posteriori* estimate of the covariance error, and Q is the constant covariance value of the process noise.

In the corrector phase, a Kalman gain value is obtained previous to the calculation of the *a posteriori* estimation.

$$K_{(k)} = \frac{HP^{-}_{(k)}}{H^2 P^{-}_{(k)} + R} \tag{14}$$

where $K_{(k)}$ is the Kalman gain, H defines the constant measurement gain, R is the covariance value of the measurement noise.

Then *a posteriori* estimation of the current state is obtained by:

$$\hat{x}_{(k)} = \hat{x}^{-}_{(k)} + K_{(k)}(z_{(k)} + H\hat{x}^{-}_{(k)}) \tag{15}$$

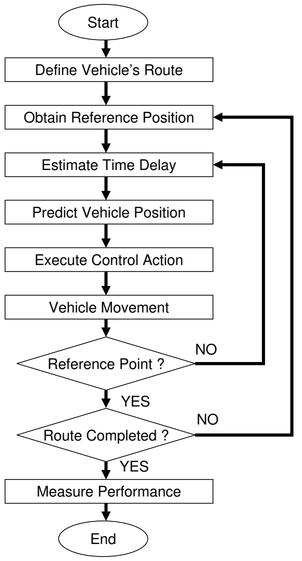

Fig. 5. Flowchart of the AGV Path Tracking Model.

where $z_{(k)}$ is the measured output of the system.

The *a posteriori* estimation of the covariance error is given by:

$$P_{(k)} = P^{-}_{(k)}(1 - HK_{(k)}) \tag{16}$$

IV. SIMULATION MODEL

This section describes the characteristics of the AGV path tracking control model implemented in Matlab for simulation purposes.

A. Model Flowchart

The algorithm that implements the AGV path tracking model is illustrated in Fig. 5.

1) The Route Planner Module provides a set of reference points to the Internal Controller, in order to set the vehicles route.
2) The Internal Controller reads the first reference point.
3) The Network Delay Estimator module estimates the next RTT value using the Kalman filter algorithm.
4) The Position Predictor module, estimates the vehicle's position at the time when the control data will arrive to the vehicle.
5) The Internal Controller calculates the wheels reference velocities and sends this information to the vehicle.
6) The vehicle moves according with the kinematic model. The control commands provided by the controller are received and applied to the vehicle after the actual RTT delay has been completed.
7) If the vehicle has reached the reference point, then the Internal Controller reads the next point.
8) If not, the cycle continues with the estimation of the next RTT value.

9) If the vehicle has completed the defined route, then the cycle ends by calculating the performance values based on the cost functions defined on Section V.
10) If not, the cycle continues by reading the next reference point.

B. Delay Generator

The implemented model includes a Delay Generator Module. This module generates the current RTT delay values $\tau(k)$ based on a gamma distribution density function. Gamma distribution is used to represent the behavior of a WiFi network, since according with different measurement results, the random and variable RTT delays of a WiFi network present an unimodal and asymmetric distribution with a long tail in the right side [13].

C. Algorithm Parameters

For the simulations and according with the distribution of the delay values, the Kalman filter parameters have been selected as follows:

- The feedback gain A is fixed and equals to one.
- There is no control input, so $u_{(k)} = 0$.
- The measurement output is obtained directly from the current state, so $H = 1$.
- The covariance value of process noise Q is zero.
- The covariance value of measurement noise R is greater than one.
- The initial value of $P_{(k-1)}$ is different from zero.

V. Simulation Setup

This section describes the different scenarios used during the simulation process in order to compare the Kalman filter algorithm performance with respect to other approaches. Also the cost functions used for evaluation are described.

A. Simulation Scenarios

Different scenarios has been considered for the simulation. With the exception of the first scenario, it is considered that the network presents a gamma distributed random RTT delay $\tau_{(k)} = [0..100ms]$.

- No Delay. There is no delay in the wireless network $\tau_{(k)} = 0$, so the AGV receives instantaneously the control information. It represents the behavior of the model considering an ideal communication environment or a local controller.
- Ideal Estimation. The estimated RTT values equal to the current RTT delay, $\hat{\tau}_{(k)} = \tau_{(k)}$. This scenario includes the adverse effects of the delay, but the effects are minimized by an ideal estimation.
- No Estimation. The controller does not make any delay compensation. This represents the worst case scenario, since the delay effects are maximized.
- Kalman Filter Estimation. Estimated RTT values $\hat{\tau}_{(k)}$, based on Kalman filter algorithm, are provided in the system in order to adjust the control information. The measurements obtained in this scenario are used to evaluate the performance of the Kalman filter algorithm.
- Statistical Estimation. Estimated RTT values $\hat{\tau}_{(k)}$, based on statistical algorithms, are provided in the system in order to adjust the control information. The measurements obtained in this scenario are used to evaluate the performance of the three statistical algorithms (Mean Value, Median Value, Max Value).

The first three scenarios are constructed for the sake of the comparative analysis. Note that the No Estimation scenario provide the worst case data, and the No Delay and the Ideal Estimation represent the best case scenarios.

B. Cost Functions

Two cost functions are used to evaluate the performance of the proposed model. The first cost function, J_1, represents the deviation of the AGV from the reference points.

$$J_1 = \sum_{t_0}^{t_f} D(p_c, p_r) \tag{17}$$

where D is the distance between the AGV's current position p_c and the next reference point p_r defined by the Route Planner, and t_0 is the initial time (for the proposed model $t_0 = 0$) and t_f is the final time when the AGV completes the route.

The second cost function, J_2 is the total travelling time spent by the AGV to reach the final destination.

$$J_2 = t_f - t_0 \tag{18}$$

VI. Simulation Results

This section shows the simulation results of the AGV path tracking model. As mentioned in section V different scenarios were simulated.

A. RTT Delay Impact

The first evaluation looks into the negative impact that the RTT delay has in the path tracking control (No Estimation scenario). This is compared to the control obtained when the Kalman algorithm is applied (Kalman Filter Estimation scenario).

As shown in Fig. 6, the trajectory followed by the vehicle has a larger deviation for the No Estimation scenario compared with the Kalman Filter Estimation scenario. Consequently, the figure clearly shows that the Kalman filter mitigates the error.

The adverse effects of the RTT delay affect both cost functions. The vehicle has a larger deviation from the original path when no estimation process is available, and it also has a higher travelling time. This is caused by the late control information received by the vehicle that is based on a wrong assumption on the vehicle's current conditions.

B. Estimation Algorithm Comparison

Both cost functions were used to evaluate the performance of the Kalman filter algorithm and the statistical algorithms compared with the best and worst cases. Three routes were defined in order to evaluate different movements of the vehicle,

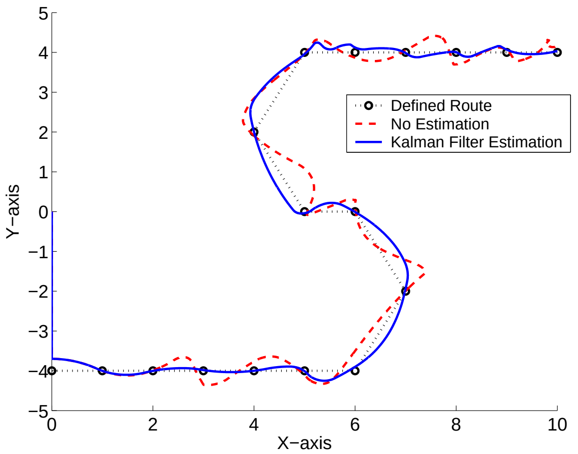

Fig. 6. Vehicle's trajectories for No Estimation and Kalman Filter Estimation scenarios.

TABLE I
PERFORMANCE EVALUATION

Simulation Scenario	$J_1(m)$	$J_2(s)$
No Delay	21.23	45.0
Ideal Estimation	26.22	51.3
No Estimation	39.61	65.3
Kalman Filter Estimation	30.71	56.3
Statistical Estimation: Mean Value	32.95	59.1
Statistical Estimation: Median Value	32.91	58.8
Statistical Estimation: Max Value	33.95	59.8

and also different RTT data sets were used during simulations to cover a wider combinations of values.

Table I shows the average values of J_1 and J_2 of the simulation runs for each scenario. The worst performance is observed when there is no estimation process. The best performance is obtained when a no delay communication network is assumed. It is observed that the Kalman filter algorithm obtains a better performance compared with the three statistical estimation algorithms.

The simulation results demonstrate that the RTT delay degrades the performance of a time sensitive system like the AGV path tracking control. This performance degradation can be reduced if an accurate estimation process is included in the system. Among the estimation algorithm the Kalman filter presents the best performance considering the path deviation and the travelling time.

VII. CONCLUSIONS

The AGV path tracking networked control system represents an industrial real-time application that can benefit from the advantages provided by wireless communication. However, WiFi also brings new challenges. For networked control systems, WiFi induced delays may prevent successful operation. This paper presents an approach in order to reduce the negative effects of the network delays, and improve the performance of the AGV networked control system.

Through the simulation results, it is shown that the Kalman filter used as a network delay estimation algorithm, provides a better performance compared with other algorithms based on statistical operations. This paper demonstrates that the use of a network delay estimator can improve the overall performance of a networked control system that uses wireless communication.

For future works the performance of the AGV path tracking system may be improved by reducing the number of communication messages between the controller and the vehicle. This can be done if communication tasks are only allowed when a threshold deviation from the defined path occurs.

REFERENCES

[1] A. Willig, K. Matheus and A. Wolisz, *Wireless Technology in Industrial Network*, Proceedings of the IEEE, Vol 93, Issue 6, pp. 1130-1151, June 2005.

[2] V. K. Kongezos and C. R. Allen, *Wireless communication between AGV's (Autonomous Guided Vehicle) and the industrial network CAN (Controller Area Network)*, Proceedings of the 2002 IEEE, International Conference on Robotics and Automation, Vol. 1, pp. 434-437, May 2002.

[3] P. Martí, J. Yépez, M. Velasco, R. Villá and J. M. Fuertes, *Managing Quality-of-Control in Network-Based Control Systems by Controller and Message Scheduling Co-design*, IEEE Transactions on Industrial Electronics, Vol. 51, Issue 6, Dec. 2004.

[4] P. A. Wiberg and U. Bilstrup, *Wireless Technology in Industry - Applications and User Scenarios*, Proceedings of the IEEE International Conference on Emerging Technologies and Factory Automation, pp. 123-131, Oct. 2001

[5] F. De Pellegrini, D. Miorandi, S. Vitturi and A. Zanella, *On the Use of Wireless Networks at Low Level of Factory Automation Systems*, IEEETransactions on Industrial Informatics, Vol. 2, Issue 2, pp. 129-143, May. 2006

[6] M. Velasco, P. Martí, R. Castané, J. Guardia and J. M. Fuertes, *A CAN Application Profile for Control Optimization in Networked Embedded Systems*, In 32th Annual Conference of the IEEE Industrial Electronics Society (IECON06), Paris, France, Nov. 2006.

[7] V. Casanova, J. Salt, A. Cuenca, V. Mascars, *Networked Control Systems Over Profibus-DP: Simulation Model*, Proceedings of the joint IEEE CCA/CACSD/ISIC, Germany, October 2006.

[8] S. Lee, K. G. Lee, M. H. Lee and F. Harashima *Integration of Mobile Vehicles for Automated Material Handling Using Profibus and IEEE 802.11 Networks*, IEEE Transactions on Industrial Electronics, Vol. 49, Issue 3, pp. 693-701, June 2002.

[9] D. Maniezzo, M. Cesana, P. Bergamo, M. Gerla and K. Yao *Real-Time Caption Streaming over WiFi Network*, Proceedings of International Conference of Information Technology: Research and Education, pp. 316-320, Aug. 2003.

[10] Y. Tipsuwan and M. Y. Chow, *Gain Scheduling Middleware for Networked Mobile Robot Control*, Proceeding of the 2004 American Control Conference, pp. 4313-4318, June-July 2004.

[11] G. Welch and G. Bishop, *An Introduction to the Kalman Filter*, (2001), University of North Carolina at Chapel Hill. [Online]. Available: http://www.cs.unc.edu/ welch/kalman/kalmanIntro.html

[12] K. Yoshizawa, H. Hashimoto, M. Wada and S.M. Mori, *Path Tracking Control of Mobile Robots using a Quadratic Curve*, Proceeedings of IEEE Intelligent Vehicle Symposium, pp. 58-63, Sep. 1996.

[13] D. Gunawardena, P. Key and L. Massoulie, *Network Characteristics: Modelling, Measurements and Admision Control*, Eleventh International Workshop on Quality of Service (IWQoS 2003), Monterey, CA, 2003.

[14] Z. Li, R. Vanijjirattikhan, M. Y. Chow and Y. Viniotis, *Comparison of Real-time Network Traffic Estimator Models in Gain Scheduler Middleware by Unmanned Ground Vehicle-Based Controller*, 32nd Annual Conference of IEEE, Industrial Electronics Society, Nov. 2005.