

Cooperative Least Square Parameter Identification by Consensus within the Network of Autonomous Vehicles

Mehdi Jalalmaab, Mohammad Pirani, Baris Fidan, and Soo Jeon
University of Waterloo

ABSTRACT

In this paper, a consensus framework for cooperative parameter estimation within the vehicular network is presented. It is assumed that each vehicle is equipped with a dedicated short range communication (DSRC) device and connected to other vehicles. The improvement achieved by the *consensus* for parameter estimation in presence of sensor's noise is studied, and the effects of network nodes and edges on the consensus performance is discussed. Finally, the simulation results of the introduced cooperative estimation algorithm for estimation of the unknown parameter of *road condition* is presented. It is shown that due to the faster dynamic of network communication, single agents' estimation converges to the least square approximation of the unknown parameter properly.

CITATION: Jalalmaab, M., Pirani, M., Fidan, B., and Jeon, S., "Cooperative Least Square Parameter Identification by Consensus within the Network of Autonomous Vehicles," *SAE Int. J. Passeng. Cars – Electron. Electr. Syst.* 9(2):2016, doi:10.4271/2016-01-0149.

INTRODUCTION

Vehicle to vehicle (V2V) and vehicle to infrastructure (V2I) communication technologies are developed and standardized in last decade to enable autonomous and semi-autonomous vehicles to share their information within the vehicular network. Autonomous vehicles are equipped with communication devices such as dedicated short range communication (DSRC) to have reliable communication with short latency [1].

Besides the ability to share ego vehicle states and characteristics, such as position, heading, speed and size of the vehicle, the agents may act as a *sensor network* and disseminate their perception and estimation about the driving environment e.g. road's geometry, friction, traffic light's state, et.

Single agent online parameter estimation may not be reliable enough for critical problems of collision avoidance control, particularly in circumstances that measurement's noise is unavoidable, and sensor malfunction is probable. Additionally, on-line estimation requires persistently exited (PE) inputs to the system [2], which may not be available in many instances. Therefore, cooperative estimation can play important role to minimize the estimation error and eliminate the effect of a faulty agent in the network.

By establishment of a vehicular sensor network, a sensor fusion scheme would be required to compute the maximum-likelihood estimate of the unknown parameters. In network topologies with a central data fusion, each agent sends its data to the center, and the maximum-likelihood estimation would be computed in the fusion center. In the presence of the road-side units (RSU), the vehicular network topology can be considered as a centralized topology; but the availability of RSU is not guaranteed always and the vehicular sensor

network should be robust to situation with no data center. Therefore, decentralized sensor fusion schemes will have advantages against central ones. In this type of data fusion, each agent only exchange data with its neighbours and performs the local computing of the maximum likelihood estimation without any global knowledge about the network topology [3].

Consensus estimation algorithm is a decentralized multi-agent estimation approach to reach an agreement regarding a certain parameter that depends on the state of all agents. The consensus algorithm also specifies how the information should exchange between an agent and all of its neighbours on the network [4]. Cooperative or multi-agent estimation within the vehicular network have been studied for applications such as traffic flow estimation [5] and vehicles positions [6, 7, 8].

In this paper, a *consensus algorithm* for cooperative parameter estimation within a vehicular network is introduced and simulated to study the performance of the algorithm for different number of network nodes and edges. In the Simulink simulation environment, a DSRC device model plus sensor model with noise are also considered inside the cooperative estimation loop to achieve more realistic simulation results and show the ability of consensus algorithm to compensate the measurement noises.

As a case study, the consensus algorithm is employed to improve road condition estimation for an adaptive collision avoidance controller. Road condition parameter, or maximum friction coefficient, is an time-varying and unknown parameter for the vehicle dynamic which can effect the autonomous vehicle control system dramatically in slippery road conations. Therefore, it is necessary that the control system has an estimation about this parameter [9]. A linear

model for the vehicle motion with time-varying constraints for states and input are formulated for MPC scheme. Bounds of constraints are formulated as the functions of the unknown parameter of road friction coefficient. Therefore an indirect adaptive MPC problem with unknown parameters in constraints is formulated to achieve more robust collision avoidance decision set.

The paper is organized as follows: First, parameter estimation by consensus is introduced. Second, distributed road condition estimation by employing the consensus algorithm is explained. Then the adaptive collision avoidance problem is formulated which is including vehicle model, input, states and safety constraints and model predictive controller formulation. At the end, the simulation results are demonstrated and conclusions are provided.

PARAMETER ESTIMATION BY CONSENSUS

Cooperative parameter estimation can increase the reliability of the estimation and enables the vehicles to make predictions of the parameter and use them in the model predictive control structure. In this section we briefly discuss about decentralized parameter estimating by consensus for a network of vehicle systems.

A sensor network can be represented with a *communication graph*. The undirected communication graph is denoted by $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V} = \{v_1, v_2, \dots, v_n\}$ is a set of nodes (or vertices) and $\mathcal{E}$ is the set of edges. The neighbours of node $v_i \in \mathcal{V}$ are given by the set $\mathcal{N}_i = \{v_j \in \mathcal{V} \mid (v_i, v_j) \in \mathcal{E}\}$. The degree of node v_i is denoted by d_i . As an example, a communication graph with 5 nodes and 5 edges is shown in Fig. 1, where $\mathcal{N}_1 = \{2, 3\}$ and $\mathcal{N}_3 = \{1, 2, 4\}$. $d_4 = 2$ and $d_5 = 1$.

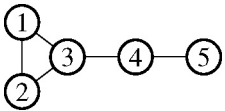


Figure 1. An example of a graph $G = \{\mathcal{V}, \mathcal{E}\}$.

For a network of n vehicles, it is assumed that each vehicle estimates the unknown scalar parameter θ with some error due to the measurement noise and incomplete estimation model:

$$y_i = \theta + v_i, \quad i = 1, \dots, n \quad (1)$$

where y_i is the vehicle i parameter estimation and v_i is the estimation error, which is a random variable with zero mean and variance σ_i . In this method, each vehicle updates its own estimation of the road condition with the following rule [3]:

$$\hat{\theta}_i(k+1) = W_{ii}\hat{\theta}_i(k) + \sum_{j \in \mathcal{N}_i} W_{ij}\hat{\theta}_j, \quad (2)$$

where W_{ij} is the linear weight on $\hat{\theta}_j$ at node i . In the vector case, (2) will turn into

$$\hat{\theta}(k+1) = W\hat{\theta}(k). \quad (3)$$

The necessary conditions for convergence of above consensus estimation law is discussed in [3,10]. At the end, all agent estimations will converge to:

$$\hat{\theta}_c = \frac{1}{n} \sum_{i=1}^n y_i, \quad (4)$$

which is the average of all measurements. While each agent estimation error variance is σ^2 , the consensus estimation $\hat{\theta}_c$ mean square error is significantly less, σ^2/n . Thus, we can say consensus improve the quality of estimation and finds the maximum likelihood estimation value based on all available estimation.

The weighting matrix of (3) should be chosen in way that it guarantees the convergence of the all nodes estimation to $\hat{\theta}_c$. The necessary and sufficient conditions for convergence of distributed estimations to the average $\hat{\theta}_c$ for any initial set of $\hat{\theta}_i(0) \in \mathbb{R}^n$ are [11]:

$$\mathbf{1}^T W = \mathbf{1}^T, \quad (5)$$

$$W\mathbf{1} = \mathbf{1}, \quad (6)$$

$$\rho(W - \mathbf{1}\mathbf{1}^T/n) < 1, \quad (7)$$

where $\mathbf{1}$ is a vector which all elements are equal to 1, and ρ denoted the spectral radius of a matrix.

DISTRIBUTED CONSENSUS ROAD CONDITION ESTIMATION

The road friction has a significant role in vehicle motion dynamics. Accordingly, collision avoidance control system requires a reliable estimate of this parameter to generate safe and effective commands. For this in mind, in this section we present an approach for cooperative road friction coefficient estimation by consensus in the vehicular network. First, the parameter identification (PI) scheme for a single agent is introduced. Then, the combination of PI and consensus algorithm for on-line cooperative road condition estimation is presented.

Tire Friction Modeling

The tire friction model is a connection between the kinematic properties of the tire (slip or relative velocity) to the dynamic properties of the tire (friction force). Various kinds of tire models exist in the literature which are mainly categorized into two groups as *static tire models* and *dynamic tire models*. In the static tire models, like Pacejka model, there is an algebraic relationship between the slip and the friction force on the tire [42]. The dynamic

dynamic models, this relationship is shown in the form of a differential equation. Dahl [13] and LuGre [14] models are some of the well known dynamic tire friction models. We use LuGre tire friction model because of its capabilities in capturing various aspects of the physical system more realistically.

The LuGre tire model is in the following form:

$$\dot{z} = v_r - \sigma_0 \theta \frac{|v_r|}{g(v_r)} z, \quad (8)$$

$$\mu = \sigma_0 z + \sigma_1 \dot{z} + \sigma_2 v_r, \quad (9)$$

where z is the internal state, $v_r = v_x - R_{eff}\omega$ is the relative velocity, μ is the normalized longitudinal force on the tire $\mu = \frac{F_x}{F_z}$. The effective radius of the tire is shown by R_{eff} and θ is a representative for the road condition coefficient. Function $g(v_r)$ is formulated below for some $\alpha > 0$:

$$g(v_r) = \mu_c + (\mu_s - \mu_c) e^{-|\frac{v_r}{v_s}|^\alpha}. \quad (10)$$

Least Square Parameter Identification (LSPI)

Based on (8)-(9) we can write a standard parameter identification equation as the following

$$f(t) = \theta^*(t)\phi(t), \quad (11)$$

where

$$\begin{aligned} f(t) &= \sigma_0 z(t) + (\sigma_1 + \sigma_2) v_r - \mu, \\ \phi(t) &= \sigma_0 \sigma_1 \frac{z(t) v_r}{g(v_r)}. \end{aligned} \quad (12)$$

where θ^* is the parameter to estimate, and $f(t)$ and $\phi(t)$ are determined by measurement. An observer same as Ref. [15] is used for estimation of $z(t)$. Here we assume that $\theta(t)$ is a piecewise constant parameter, thus we have $\dot{\theta} = 0$. We can make the adaptive law by robust recursive least square estimation with projection. The projection is due to the fact that $\theta_1 \leq \theta \leq \theta_2$ and the robustness of the adaptive law is due to the fact that there are some uncertainties in the measurements, specially in v_r which is obtained from subtraction of the integral of noisy a_x and the angular velocity ω . It is assumed that the constant parameters σ_0 , σ_1 and σ_2 are known. The estimation model for θ and the recursive least square algorithm with forgetting factor are formulated as follow [2]:

$$\hat{f}(t) = \theta(t)\phi(t), \quad (13)$$

$$\varepsilon = \frac{f(t) - \hat{f}(t)}{m_s^2(t)} \quad (14)$$

$$m_s^2(t) = 1 + \alpha(t)\phi^T(t)\phi(t) \quad (15)$$

$$\dot{\theta}(t) = P(t)\varepsilon(t)\phi(t), \quad (16)$$

$$\dot{P}(t) = \beta P(t) - P(t) \frac{\phi(t)\phi^T(t)}{m_s^2(t)} P(t), \quad (17)$$

where $\varepsilon(t)$ is estimation error, $m_s^2(t)$ is the normalization signal, (16) is the adaptive law, $\theta(0) = \theta_0$ and $P(0) = P_0 = Q_0^{-1}$. Based on Theorem 3.7.1 in Ref. [2], if ϕ/m_s is *persistently excited*, then the recursive least-square algorithm with forgetting factor guarantee that $\theta(t) \rightarrow \theta^*$ as $t \rightarrow \infty$. The convergence is exponential if $\beta \geq 0$.

Cooperative Road Condition Estimation by Consensus

A network of n vehicles are considered. Each agent is connected to number of d_i surrounding agents, $i, i = 1, \dots, n$. The cooperative road condition estimation algorithm employed in this paper is as followed: First, by the LSPI technique introduced in this paper, each agent estimates the road condition $\hat{\theta}_{i,LSPI}$ individually, and initialize $\hat{\theta}_i(0) = \hat{\theta}_{i,LSPI}$. Secondly, each agent broadcasts and propagates $\hat{\theta}_i(k)$ within the network recursively by consensus algorithm.

Since the process of LSPI takes some time to regulate the estimation error, to capture the time varying nature of the road condition parameter, reinitializing of $\hat{\theta}_i(0)$ repeats at $f_{PI} = 1/T_{PI}$ Hz, where roughly speaking, T_{PI} is a settling time of LSPI estimation dynamic. vehicular network communications performs with larger frequency of $f_{VN}, f_{PI} \ll f_{PI}$, therefore the consensus algorithm would have enough time to iterate and converge to $\hat{\theta}_c$.

Moreover, the LSPI adaptive law (16) should be initialized by some $\theta_{0,LSPI}$. At the beginning of estimation process, this parameter can be chosen as zero or the most probable value for road condition. But after one iteration of consensus road condition estimation and computation of $\hat{\theta}_c$, this value can be considered as the initial value of LSPI.

The simple weighting scheme of *constant edge weight* [16] is selected for the consensus algorithm. All the edge weights set equal to a constant α , then the self-weights of nodes are chosen to satisfy convergence conditions:

$$W_{ij} = \begin{cases} \alpha & i, j \in \mathcal{E}, \\ 1 - d_i \alpha & i = j, \\ 0 & \text{otherwise.} \end{cases} \quad (18)$$

As an example, constant edge weight scheme for network of Fig. 1 is: