Intelligent Sensor Validation And Sensor Fusion For Reliability And Safety Enhancement In Vehicle Control

Authors
Agogino, Alice
Goebel, Kai
Alag, Sanam

Publication Date
1995

CALIFORNIA PATH PROGRAM
INSTITUTE OF TRANSPORTATION STUDIES
UNIVERSITY OF CALIFORNIA, BERKELEY

Intelligent Sensor Validation and Sensor Fusion for Reliability and Safety Enhancement in Vehicle Control

Alice Agogino
Kai Goebel
Satnam Alag

California PATH Research Report

UCB-ITS-PRR-95-40

This work was performed as part of the California PATH Program of the University of California, in cooperation with the State of California Business, Transportation, and Housing Agency, Department of Transportation; and the United States Department of Transportation, Federal Highway Administration.

The contents of this report reflect the views of the authors who are responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policies of the State of California. This report does not constitute a standard, specification, or regulation.

November 1995

ISSN 1055-1425

INTELLIGENT SENSOR VALIDATION AND SENSOR FUSION FOR RELIABILITY AND SAFETY ENHANCEMENT IN VEHICLE CONTROL

Final Report
PATH Project
MOU-132

June 30, 1995

Principal Investigator

Alice Agogino
Tel.: (510) 642-6450
Fax.: (510) 643-8982
aagogino@euler.Berkeley.EDU

Graduate Research Assistants
Kai Goebel
Satnam Alag

Department of Mechanical Engineering
University of California at Berkeley
Berkeley, CA 94720

TABLE OF CONTENTS

Final Report

PATH
MOU132

Intelligent Sensor Validation and Sensor Fusion for Reliability and Safety Enhancement in Vehicle Control

Alice M. Agogino, Kai Goebel, and Satnam Alag
Dept. of Mechanical Engineering
UC Berkeley

Abstract

In this report we present an evaluation of methods for validation and fusion of sensor readings obtained from multiple sensors, to be used in tracking automated vehicles and avoidance of obstacle in its path. The validation and fusion is performed in two modules which are part of a larger five-module hierarchical supervisory control architecture. This supervisory control architecture operates at two levels of the Automated Vehicle Control Systems (AVCS): the regulation and the coordination level. Supervisory control activities at the regulation layer deal with validation and fusion of the sensor data, as well as fault diagnosis of the actuators, sensors, and the vehicle itself. Supervisory control activities at the coordination layer deal with detecting potential hazards, recommending the feasibility of potential maneuvers and making recommendations to avert accidents in emergency situations. In this grant we formulated the need for an hierarchical approach and then focussed in depth on the two modules sensor validation and sensor fusion. Tracking models were introduced for the various operating states of the automated vehicle, namely vehicle following, maneuvering, i.e. split, merge, lane change, emergencies, and for the lead vehicle in a platoon. The Probabilistic Data Association Filter (based on Kalman filtering) is proposed for the formation of real time validation gates and for fusing the validated readings. A topology for an influence diagram which captures the interaction of the various vehicle components and the sensing equipment, as well as the algorithmic sensor validation algorithms were developed. Furthermore, experiments for characterization of the optical triangulation longitudinal sensor developed by Qualimatrix Corp. were carried out. The other two longitudinal sensors, namely the radar and the sonar sensor, were tested as well. These tests were conducted under both dynamic and static test conditions as well as under vibrations.

1.0 Introduction

In order to perform its basic functions the Intelligent Vehicle Highway System (IVHS) requires a large number of sensors for control at the coordination level, the engine level, and for sensing and communication between the vehicles and the IVHS main controller. These basic functions include longitudinal control, lateral control, platooning, and maneuvering techniques such as lane change, exiting the automated lane, etc. In the automated vehicle, which is being controlled based on these sensor readings, changes in some sensor values directly affect the controllers output initiating necessary control actions. Therefore, for all subsystems to work well and reliably the IVHS system requires high sensor data fidelity. Unfortunately, most generally sensor readings are uncertain and errors do exist in sensor readings due to the imperfect nature of the sensors and the additive noise inherently added to the readings. In addition, there is uncertainty in the vehicles ever changing stochastic environment as a consequence of its interaction with other vehicles in its neighborhood as well as with components within the car. This uncertainty coupled with the fast dynamic nature of the complex IVHS system could lead to potentially hazardous (emergency) conditions with possibly fatal consequences. An inconsistent sensor reading could be a result of a system or process failure, and it is important to distinguish between these types of failures. It is equally important to differentiate between different failures in either class. In any case it is necessary to implement alternative policies in real time to handle unexpected situations to maximize overall reliability and safety of the IVHS system.

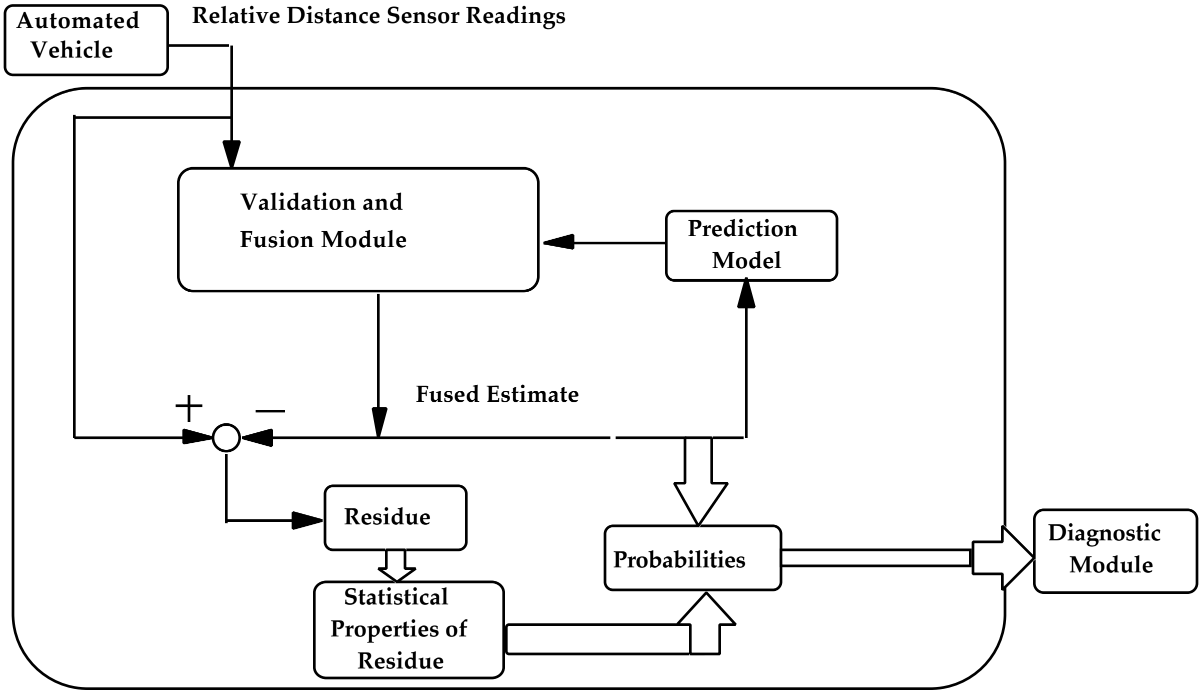

Automated Vehicle
Relative Distance Sensor Readings
Validation and Fusion Module
Prediction Model
Fused Estimate
+
−
Residue
Probabilities
Diagnostic Module
Statistical Properties of Residue

Fig.1: Schematic of Validation and Fusion with link to Diagnostic Module

Data driven supervisory control activities for IVHS are concerned with fault detection, fault isolation, and control reconfiguration of the many sensors, actuators and controllers that are used in the control process. In order to carry out corrective actions (control maneuvers) that maintain the overall integrity of the IVHS system, the sources of uncertainty will be considered before arriving at the final diagnosis of the vehicle state. We see the solution to the diagnosis as a five-module approach. The modules (Alag, Goebel, and Agogino, 1995a) are: 1.) Sensor validation, 2.) Sensor fusion, 3.) Fault

Detection, 4.) Hazard analysis, and 5.) A safety decision maker. The first two modules, sensor validation and sensor fusion, are addressed in this project. Within the automated vehicle control system (AVCS) hierarchy, these five modules can be found on two levels, the regulation layer and the coordination layer as an intermediate supervisory controller. It combines the advantages of having access to the data of the regulation layer as well as the information from the coordination layer. It operates in every vehicle and serves the purpose of real time monitoring and diagnosis of the components in the vehicle and between the vehicle and its environment (other objects in its neighborhood). It predicts incipient failures and recommends suitable corrective actions. The main emphasis of this research is to solve the first two modules of the hierarchical architecture for this supervisory control methodology in the IVHS context and to outline some of the potential techniques that could be used for each module and the approach taken by us to build the supervisory controller.

Fig. 1 shows the schematic of validation and fusion. Raw sensor data enter this module. They form with the fused estimates residues from which statistical properties are extracted. Probabilities for sensor data are calculated and can be used in the diagnostic module. The fused estimates also serve to update the prediction model. Fig. 2 shows the flowchart of the methodology. Sensor data from the vehicle, in this case longitudinal distance readings, go first through the algorithmic sensor validation module for evaluation. Depending on the result together with information about other vehicles within the platoon obtained through communication channels and information from the decision analytic module, it will be decided whether a model based approach can be used or not. If the former is the case, a fused estimate is obtained after the data are validated through validation gates to calculate probabilities for each sensor reading. If no model can be used, information about sensor accuracy is used to arrive at a fused estimate. Fig. 3 shows the classification of the vehicle state that is used for validation and fusion. Within the platoon, the first vehicle is classified as the leader, all other are follower vehicles. Two states can be defined for our purposes: a desired one and an undesired one. The undesired state is always potentially hazardous. For follower vehicles, desired states can be categorized as steady and transient. The latter is a state during a maneuver, the former denotes regular following activity. For leader vehicles, desired states are categorized depending on what is in range of the vehicle. Models of ranging order and/or the ASV are used for each situation as will be elaborated on later.

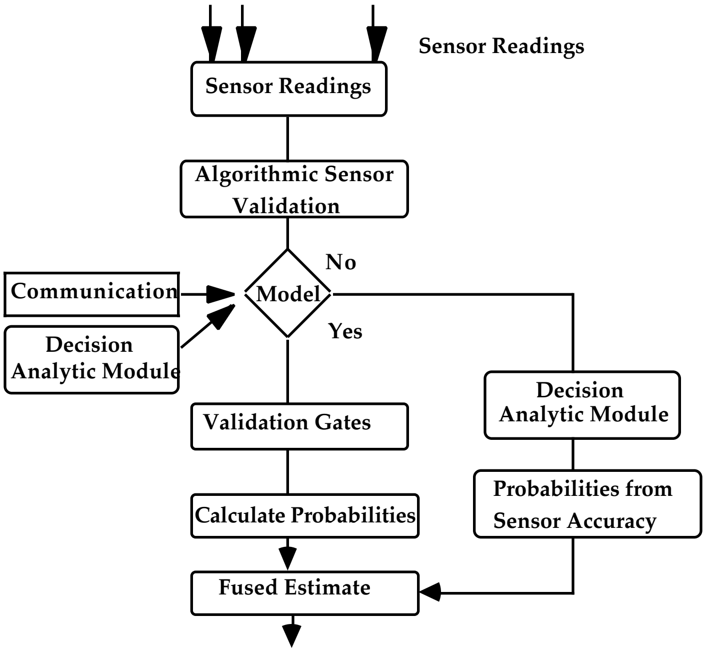


Fig. 2: Flowchart of the Sensor Validation and Fusion Methodology for the Longitudinal Sensors

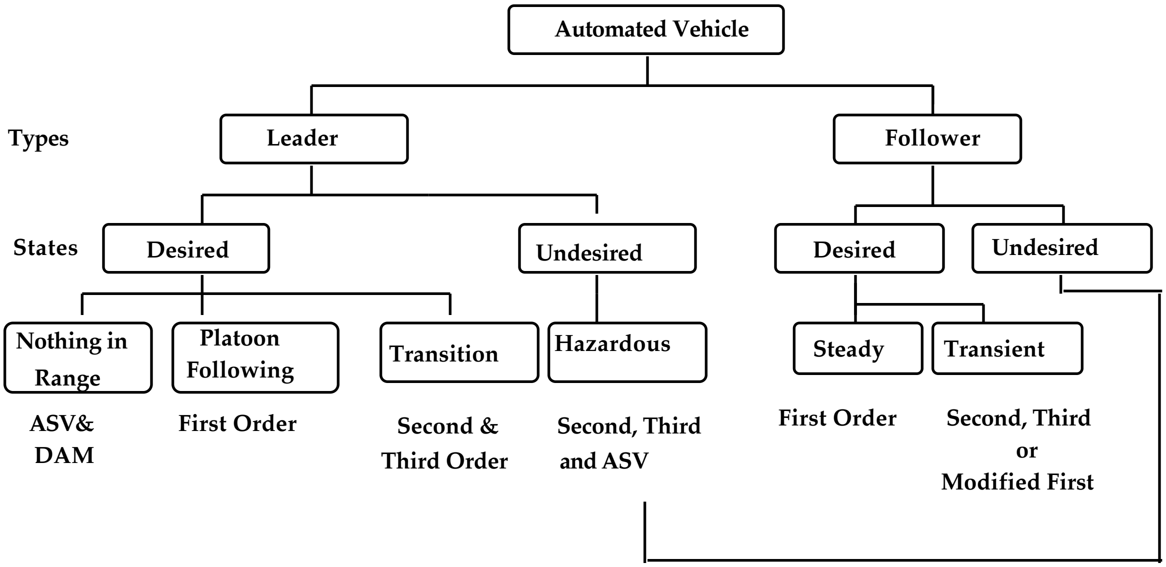


Fig. 3: Classification of the Vehicle operating states

2.0 Methodology

First, the significance of the supervisory controller will be explained. Next, we explain how the supervisory controller is divided up into two levels within the AVCS architecture. The different modules which the controller consists of are introduced and tools for each module are enumerated.

2.1 Supervisory Control Architecture

Control of the automated vehicle in the IVHS can be divided into two operations, the first deals with the basic lateral and longitudinal control of the vehicle, while the second operation deals with various *supervisory tasks*. The supervisory tasks consist of indicating undesired or unacceptable process states (Ramamurthi and Agogino, 1993) and to take appropriate actions in order to avoid an accident. Normally, faults in the vehicle on the IVHS will affect its performance and control. A fault can be considered as any unacceptable deviation in the characteristic property of the vehicle (like a tire burst), the actuator, the sensors and the controllers. Component failure, energy supply failures, environmental failures, maintenance errors etc. are some of the common sources of failure (Pouliezos and Stavrakankis, 1994).

In IVHS, *supervisory controllers* are required for two different tasks: first to detect faults in the vehicle and the second to detect faults (or deviations) in the environment around which the vehicle is operating. An example of the latter is the development of unforeseen faults in the vehicles' neighborhood due to sudden change in the expected configurations of the collection of vehicles which could be caused by a failure of either the sensors or incorrect response to a stimulus of the controller in one of the automated lanes. This calls for real time fault tolerant action on part of the supervisory controller. Another extreme case is one in which the tires burst and then the entire dynamics of the vehicle changes. In order to accommodate the changes in the engine controller dynamics, the intermediary controller needs to react to the new situation in real time and to come up with an exigency response to either see the maneuver through to completion or to abort it in a fashion which is consistent with the configurations of all the interacting vehicles in a manner which is feasible, safe and quick in its response. For instance the arrival of an unexpected obstacle (e.g. load fallen off a truck, or an animal) during normal platooning operations as well as midway during the maneuver needs real time decisions to move the vehicle or vehicles involved into a position which avoids any collision or, if unavoidable, to a configuration which minimizes the total damage involved.

The supervisory controller also plays an important part during maneuvers. The elementary maneuvers based on the path planning achieved by the coordination layer as suggested in (Hsu et al., 1991), require the individual vehicle to interact with the environment including the vehicles in neighboring lanes via the appropriate longitudinal sensors and communication facilities. Once the vehicle has completed its protocol and has the sequence of maneuvers implemented which it has to carry out, its interaction with the environment and the requirement that it closely monitors changes in them and reacts to them in real time is vital to maintain a high degree of reliability and safety. The vehicle intending a lane change interacts with the vehicles of the neighboring lanes, which may be either members of a platoon or free agents as shown in Fig. 4. Once the sequence of maneuvers of the system is decided, it is important to have an intermediary supervisory advisor to monitor the progress of the maneuver such as lane merge, split or lane change and see it through to a satisfactory end.

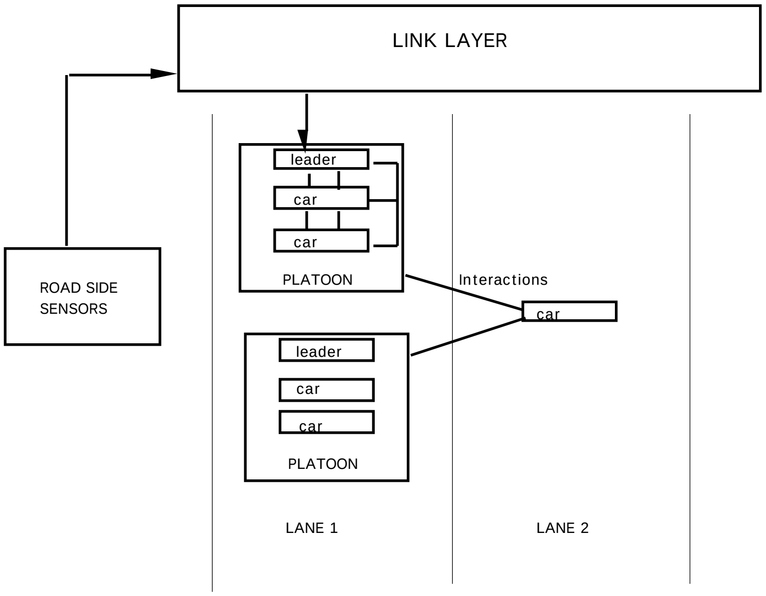


Fig. 4 : Interactions between vehicle and platoons during maneuvers

Since safety is of prime importance in an IVHS system, it is imperative to first validate and fuse the uncertain readings obtained from the numerous sensors. Therefore, the first task to be carried out by the supervisory controller is to validate the sensor readings and get an estimate for the various parameters to be used in the monitoring and fault diagnosis part. Our experience has shown that real time fault diagnosis for complex systems benefits from an hierarchical information processing structure with selecting faults on one level, focusing in more detail on these candidate faults at a higher level, and finally looking for facts to confirm the ultimate diagnosis and make repair or recovery recommendations at yet a higher level. The approaches taken include the integration of heuristics and model-based reasoning, procedures for fusing qualitative and quantitative data for developing probability assessments, and explicit reasoning about the time constraints inherent in real-time processing of large amounts of data.

We therefore propose a multi-level architecture for real time monitoring and diagnosis of the automated vehicle, which consists of the modules for sensor validation, sensor fusion, fault diagnosis, hazard analysis, and a safety decision advisor. Before the different modules of the system are introduced, a brief discussion shall illustrate how the supervisory controller proposed here is integrated into the AVCS. Fig. 5 shows the outline of the complex hierarchical structure of the AVCS system control architecture which in addition to the link, coordination, regulation, and physical layer consists of the network layer at the top.

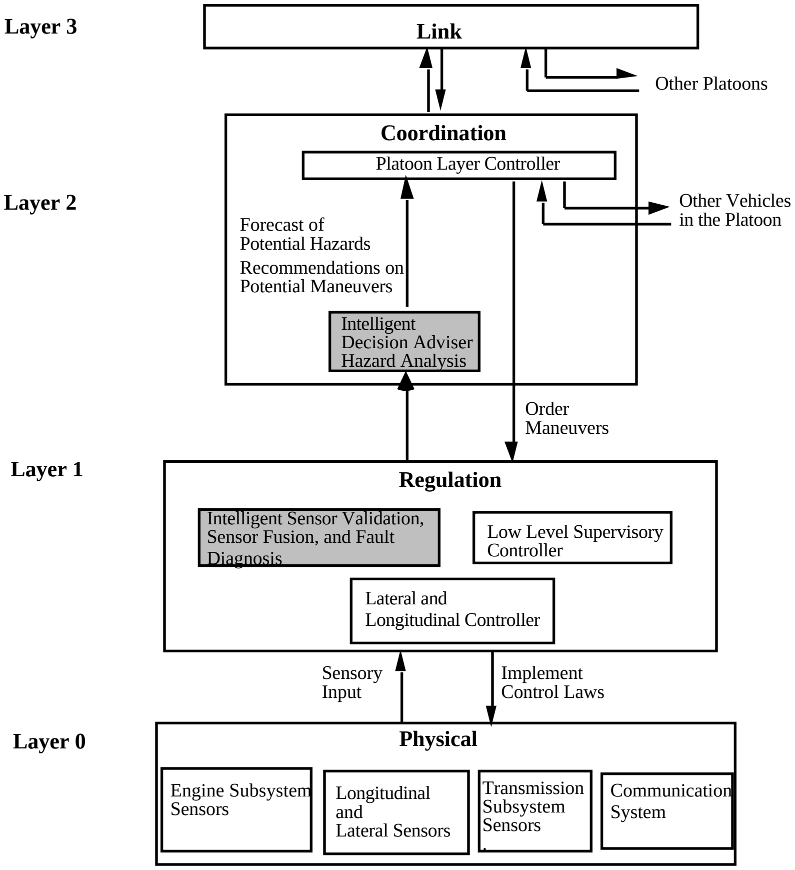


Fig. 5: Position of Intelligent Sensor Validation, Sensor Fusion, Fault Diagnosis, Hazard Analysis, and Intelligent Decision Advisor in the AVCS Control Hierarchy

The task of the network layer controller is to assign a route to each vehicle entering the system. Below this is the link layer controller, one for a long segment of each highway. Its task is to assign a path to each vehicle entering the highway and target velocity for the aggregate traffic. The remaining tasks are distributed among individual vehicles (Varaiya and Kurzhanski, 1988). The coordination layer in each vehicle is responsible for planning its path as a sequence of three elementary maneuvers, and for coordinating with neighboring vehicles the implementation of each maneuver. The regulation layer below it is responsible for executing a pre-computed feedback control in response to a command from the coordination layer as well as performing lower-level control tasks (Varaiya, 1991, Varaiya and Shladover, 1991). Currently there is no element that acts as a supervisor in the sense that the information of redundant sensors (both hardware and analytical) is coordinated and sifted for inconsistencies. This need (Hsu et al., 1991, Patwardhan et al., 1992a) between the vehicle sensors and the platoon layer must be responded to ensure proper operation of the system. Many parts of the system still assume that the communication systems and sensors work perfectly. This assumption is not realistic. Although uncertainty is taken into account in some cases, (Hedrick and Garg, 1993, Patwardhan et al., 1992b), there exists no element that looks at the sensor readings from all relevant sensors at the physical level as well as information from the communication. We propose a supervisory decision advisor which considers the uncertainties in sensor readings to form a link between the coordination level controller and the regulations level controller and to rectify aberrant sensor readings by taking into account the information of several partly redundant sources. This supervisor decision maker will be subject of a future proposal.

Sensor validation and sensor fusion take place at the regulation layer (shaded box on the regulation layer in fig. 5). Input are data from the sensors of the physical layer. The fault diagnosis of the various subsystems is also located at the regulation layer. Some of the reasons for uncertainty in sensor information are measuring device error, environmental noise, and flaws or limitations in the data acquisition and processing systems. Extracting information from raw data is often difficult because of noise, missing data or occlusions. Phenomena may show up at disparate locations and can have a variety of time scales, from low frequency signals to high frequency vibrations. Therefore, the regulation layer seems to be the appropriate place for sensor validation and sensor fusion as it permits access to various sensor values at the same time. On the coordination layer, output from the sensor validation and fusion module as well as from the fault diagnosis module are used to perform hazard analysis and intelligent decision making because this central place within the control architecture allows for integration of all relevant information. The modules are displayed isolated in fig. 6. Input and output of each module are shown on the right hand side.

2.2 The Modules within the Control Hierarchy

On the lowest level is the sensor validation module. It is responsible to detect sensor failures and sensor faults. After the validation, in the case of multiple sensors, or a group of sensors measuring a set of related quantities sensor fusion takes place. Redundancies of the sensors as well as correlation of processes measured by different sensors are utilized to find fused or corrected sensor values using Bayesian and Kalman filtering techniques. Based on the results of the sensor validation and fusion modules, a fault diagnosis module looks at potential failures of the various subsystems and calculates their respective probability. Here, use is made of subsystem influence diagrams which capture the influences of various failures on subsystem parameters. This information will then be used in a hazard analysis module to compute the probability of various hazards. Finally, an intelligent decision adviser is proposed which provides recommendations on potential maneuvers and other actions to the coordination level controller. This decision advisor has to come up with the optimal decision in real time. Since reaction time has to be small, optimization techniques which are generally very slow are trained off-line to obtain optimum responses for various scenarios. These can be implemented by means of look-up tables and pattern recognition systems and then used on-line in real time. In this way a link between the vehicle sensors and the coordination layer is achieved. This greatly improves the integrity of the system in diverse and adverse conditions.

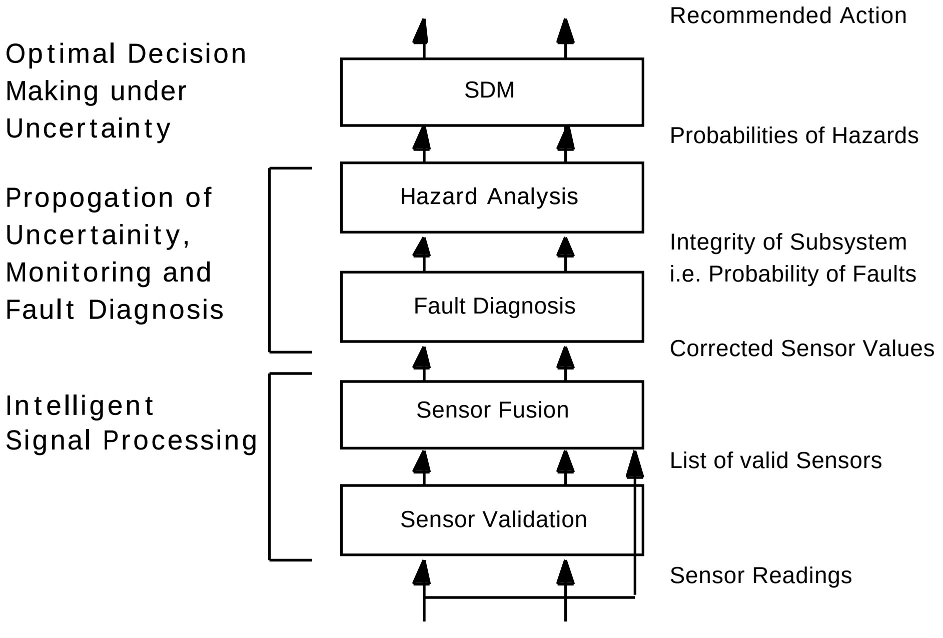


Fig. 6: Framework of 5 modules for sensor validation, fusion, and fault diagnosis, hazard analysis, and safety decision maker

2.2.1 Sensor Validation Module

In the IVHS system a string of closely spaced vehicles, called platoons will be traveling under automated control (at high velocities). Depending on their position on the automated highway, the vehicles can be divided into two types: those that are in the platoon (follower vehicles) and those which lead the platoon (leader) or which are traveling alone (equivalent to a platoon of length one). The tracking tasks for these two vehicle states is different. It is therefore necessary to distinguish between these two cases in our validation and fusion methodology. To deal with the various situations, we further divide the operating conditions into several states. There are three states for the follower vehicles: steady state, transient, and hazardous states as displayed in fig. 3. The first two are desired states while the third is undesired. A vehicle under automatic control is defined to be in the steady state when it is in a platoon and is trying to follow the one before it at a fixed (known) distance. Whenever the vehicle carries out a maneuver such as a split (leaving the platoon), merge (joining the platoon) or lane change it is defined to be in a transient state. This state involves relative acceleration between the vehicles and includes the state of the lead vehicle in the platoon. The last state, the hazardous state is defined as the state when the vehicle carries out emergency maneuvers to avoid a catastrophe (accident). The lead vehicle, on the other hand, has only two states, a desired state, and a hazardous state. If the lead vehicle's distance to the next object is either big enough, e.g. when the distance to the next object is greater than the minimum for safe distance or the object is out of range, then the state is a desired state. If the distance to the next object is to small, we say it is a hazardous state. Other hazardous states exist beyond the one outlined above (Hitchcock, 1992).

Fig. 7 shows the methodology followed for the validation and data fusion process. We first begin by building models for the operating states of the two types of vehicles: follower and lead, During operation these models are used to build validation gates for the sensor data by using a Kalman filter estimate. Inherent with the IVHS system is the availability of additional information such as the velocity and acceleration of the lead vehicle in the platoon. This is the velocity and acceleration that other vehicles in the platoon are trying to follow. In addition to the desired velocity and acceleration, the coordination layer controller also transmits information about maneuvering techniques (merge, split and lane change) to the vehicles in a platoon. This information is transmitted through the communication channel and can be used for switching between the models. In the absence of information a test of hypothesis can be carried out to determine the states of the vehicle. To estimate the deviation in the process from the model and the sensor noise we use the Kalman filtering algorithm to form validation gates. We also employ an Algorithmic Sensor Validation (ASV) (Kim et al., 1992) which uses first principles and checks for maximum physical bounds of sensor readings This method is in sequence before the validation gates and they are used together for the validation process. After the validation process, a Bayesian method namely the Probabilistic Data Association Filter is used for data fusion.

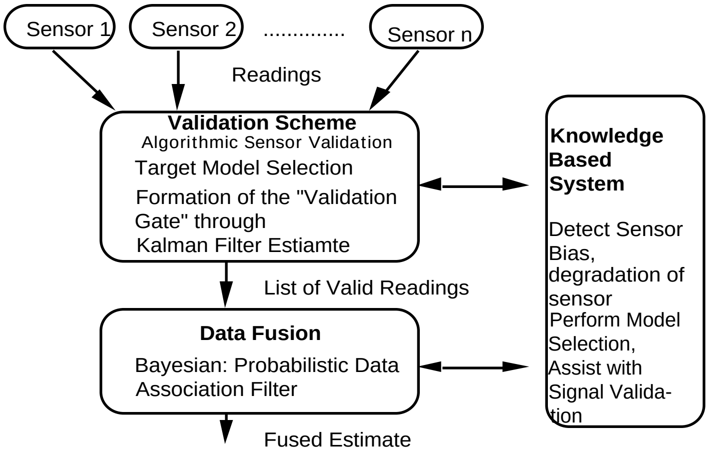

Sensor 1
Sensor 2
....
Sensor n
Readings
Validation Scheme
Algorithmic Sensor Validation
Target Model Selection
Formation of the "Validation Gate" through Kalman Filter Estiamte
Knowledge Based System
Detect Sensor Bias, degradation of sensor
Perform Model Selection, Assist with Signal Valida-tion
List of Valid Readings
Data Fusion
Bayesian: Probabilistic Data Association Filter
Fused Estimate

Fig. 7: Flowchart of the Methodology

2.3 Model Building

Follower and lead vehicle will be modeled differently because of their distinct operating conditions. In particular, the operating states (steady, transient, and hazardous for follower, and desired and hazardous for lead vehicle) are considered and different models for each case are introduced.

2.3.1 Modeling Follower Vehicles

Consider a vehicle in the IVHS under automatic control moving in its lane as shown in Fig. 8 (I,J) is a fixed inertial reference frame, while the frame (i,j) is fixed to vehicle (n-1) and moves along with vehicle (n-1). n is the target (either another vehicle or another arbitrary object) in its lane whose position, velocity and acceleration with respect to vehicle (n-1) we are trying to estimate, i.e.

$$\ddot{\Delta}_x = \ddot{X}_n I - \ddot{X}_{(n-1)} I = (\ddot{x}_n - \ddot{x}_{(n-1)}) i \overset{\Delta}{=} \ddot{x} i, \dot{x} i, x i \qquad (1)$$

where

$\ddot{x}$ is the relative acceleration
$\dot{x}$ is the relative velocity
x is the relative position

Next, we build models to be used in the validation process that describe the various operating conditions of the vehicles in the IVHS.

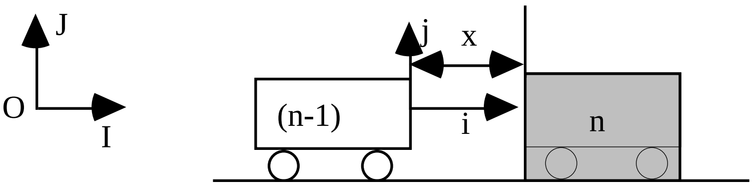


Fig. 8 Automated Vehicle (n-1) Behind an Object in its Lane

2.3.1.1 Steady State

Steady state is defined as the state in which the automated vehicle in a platoon is following the one before it at a constant (zero) relative velocity, i.e. at $\dot{x} = 0$. However, in practice the velocity will undergo at least slight changes. We model these changes as a continuous time white noise $\tilde{v}$ as follows,

$$\dot{x}(t) = \tilde{v}(t) \qquad (2)$$

$$E\left[\tilde{\text{v}}(t)\right] = 0 \qquad (3)$$

$$E\left[\tilde{\text{v}}(t)\tilde{\text{v}}(\tau)\right] = q(t)\delta(t-\tau) \qquad (4)$$

where

$E\left[\tilde{\text{v}}(t)\right]$ is the expected value of $\tilde{\text{v}}(t)$

q(t) is the covariance of $\tilde{\text{v}}(t)$.

The discrete time state equation is

$$x(k+1) = x(k) + v(k) \quad (5)$$

$$Q = E\left[v(k)v^{\mathrm{T}}(k)\right] \qquad (6)$$

where

Q is the covariance of v(k).

2.3.1.2 Transient State: Maneuvers

Varaiya (1991) defines three kinds of maneuvering techniques for the vehicles in IVHS: merge, split and lane change. Merge is the procedure by which a vehicle joins a platoon, while in split it leaves the platoon. In lane change the vehicle moves to the next lane. Smooth trajectories have been designed for these procedures (Narendran, 1994). These trajectories have been designed to keep the vehicle jerk (ride quality) and acceleration within acceptable limits. For example, for a lane change maneuver the trajectory is relatively simple as it requires a lateral position change of a fixed distance which is the distance between the centers of adjoining lanes and the final reference for trajectory design (i.e. center of target lane) is fixed. In general, for merge and split the desired spacing and desired relative velocities at the beginning and end ($t_0 \& t_f$, respectively) are known. Using these four conditions a desired spacing profile can be generated which is of the form

$$sp_d(t) = c_0 + c_1 t + c_2 t^2 + c_3 t^3 \qquad (7)$$

where

sp_d is the desired spacing profile

c_o, c_1, c_2, c_3 are constants.

Trajectories for various conditions (with and without platoon acceleration) have been developed by Narendran and Hedrick, (1994) to keep vehicle jerk (ride quality) and acceleration within acceptable limits. The important point is that the desired relative distance and relative velocity of the vehicles during the maneuver is known and can be used for our sensor validation procedure. In this case, we model the residuals, i.e. the difference between the actual distance and the desired distance

$$\begin{aligned} r(t) &= x(t) - x_d(t) \\ \dot{r}(t) &= \dot{x}(t) - \dot{x}_d(t) \end{aligned} \qquad (8)$$

where

r is the residual

x_d is the desired distance.

As in the previous case, $\dot{r}(t)$ should be ideally zero. As before we model the changes in $\dot{r}(t)$ as a continuous time white noise, to obtain equations similar to those given in (2), (3) and (4).

2.3.1.3 Hazardous State

The hazardous state demands special attention because it is here where the safety of the system is at stake. Five major hazards (Hitchcock, 1991), and sequence of events that precursor these hazards have been identified. For example, hazard A is defined as the condition when the distance between two vehicles in a platoon is less than the safe inter-platoon distance. To this end, two models have been developed which deal with specific hazardous situations. One assumes a constant velocity between the target and the follower vehicle (which is too close). The other one assumes a constant acceleration between the target and the follower vehicle (which is too close). More models will be developed when more hazardous states are considered. These would be a combination of the constant distance model (explained above), the constant velocity and constant acceleration models (discussed next).

For the first case with constant velocity we model the changes in the relative acceleration as a continuous time white noise $\tilde{\nu}$, $\ddot{x}(t) = \tilde{\nu}(t)$, where $\tilde{\nu}$ has the same properties as in (3) and (4). We can also define the process noise directly in discrete terms as

$$\mathbf{v}(k) = \Gamma \nu(k) \qquad (9)$$

where

$\nu(k)$ is a scalar valued zero-mean white noise sequence

$$\Gamma = \begin{bmatrix} \frac{T^2}{2} & T \end{bmatrix}^{\mathrm{T}} \qquad (10)$$

$$\mathbf{x}(k+1) = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \end{bmatrix} + \Gamma \sigma_v \qquad (11)$$

$$E[\nu(k)\nu(j)] = \sigma_v^2 \delta_{kj} \qquad (12)$$

$$\mathbf{Q} = \Gamma \sigma_{\mathbf{v}}^2 \Gamma^{\mathrm{T}} = \begin{bmatrix} \frac{T^4}{4} & \frac{T^3}{2} \\ \frac{T^3}{2} & T^2 \end{bmatrix} \sigma_v^2 \qquad (13)$$

where

σ_v is the standard deviation

Q is the covariance of the process noise.

Here, the implicit assumption is that the relative acceleration between the vehicle and the target undergoes constant acceleration $\nu(k)$ during sampling period k and that these accelerations are independent from period to period. Therefore, for one sampling period the change in relative velocity is $\tilde{\nu}$(t)T and the change of the relative distance is $\tilde{\nu}$(t)T^2/2. The main difference between the continuous and the discrete case is that in the discrete case the assumption is piece wise constant white noise, while for the other case the assumption is continuous time white noise (Bar-Shalom, 1993).

For the second case with constant relative acceleration between the two vehicles $\dddot{x}(t) = 0$. This model is applicable in case of emergency braking. However, in practice the acceleration will never be exactly constant. We model its changes by means of a zero-mean white noise as follows $\dddot{x}(t) = \tilde{\nu}(t)$. The smaller the variance of the noise the more constant is the acceleration. In this case the state vector and the continuous time state equation is

$$\dot{\mathbf{x}}(t) = \mathbf{A} \begin{bmatrix} x \\ \dot{x} \\ \ddot{x} \end{bmatrix} (t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tilde{\nu}(t) \qquad (14)$$

Similarly, the discrete time state equation with sampling time T is

$$\mathbf{x}(k+1) = \mathbf{F}\mathbf{x}(k) + \mathbf{v}(k) \qquad (15)$$

$$\mathbf{F} = e^{\mathbf{A}T} = \begin{bmatrix} 1 & T & \frac{T^2}{2} \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix} \qquad (16)$$

$$\mathbf{Q} = E[\mathbf{v}(k)\mathbf{v}'(k)] = \begin{bmatrix} T^5/20 & T^4/8 & T^3/6 \\ T^4/8 & T^3/3 & T^2/2 \\ T^3/6 & T^2/2 & T \end{bmatrix} q \qquad (17)$$

where

Q is the covariance matrix for the process noise discretized from continuous time.

As for the other cases we can directly define the process noise in the discrete case. $\mathbf{v}(k) = \Gamma\nu(k)$ where $E[\mathbf{v}(k)\mathbf{v}^{\mathrm{T}}(k)] = \sigma_v^2\delta_{kj}$ and $\Gamma = [T^2/2 \quad T \quad 1]$. In this case, we assume that the acceleration is a discrete time Wiener process, i.e. non stationery random process with mean zero and variance αt . $\nu(k)$ is the acceleration increment in a sampling time ("jerk"). The process noise covariance is

$$\mathbf{Q} = \Gamma\sigma_{\mathbf{v}}^{\mathbf{2}}\Gamma^{\mathrm{T}} = \begin{bmatrix} T^4/4 & T^3/2 & T^3/2 \\ T^3/2 & T^2 & T \\ T^2/2 & T & 1 \end{bmatrix} \sigma_v^2 \qquad (18)$$

An estimate for the covariance matrix for the sensor noise R can be obtained by testing the sensors under test conditions and for various distances, i.e. obtaining accuracy versus range of the sensor. The covariance for the process noise has one unknown which needs to be estimated. As the various operating states of the IVHS become more standardized the unknown parameter can be estimated. As a guideline, the changes in the velocity over a sampling period T are of the order of $\sqrt{Q_{22}} = \sqrt{qT}$ for the constant velocity model. For the constant acceleration model, the changes in the acceleration over a sampling interval T are of the order of $\sqrt{Q_{33}} = \sqrt{qT}$. A particle range for choosing σ_v is $0.5\Delta a_M \le \sigma_v \le \Delta a_M$, where Δa_M is the maximum relative acceleration increment over a sampling period (Bar-Shalom, 1993).

If the statistical properties for the various models cannot be estimated reliably then an adaptive filtering approach can also be used. For this purpose, several algorithms are available (Astrom and Eykhoff 1971, Chin 1979, Jazwinski 1969 and Mehra 1970).

2.3.2 Modeling the Lead Vehicle

The lead vehicle in a platoon sets the velocity and the acceleration for the other vehicles in the platoon. This is determined by the conditions on the highway. A number of different scenarios are possible, for example there may be no target in the operating range of the longitudinal sensor, or the target may be a part of another platoon or the target could be a stationery target (e.g. a stalled car in the same lane), etc. The lead vehicle is more prone to misinterpret longitudinal sensor data than the follower vehicle. The short distance range sensor readings will in most cases be out of range. False readings can occur when the signal is reflected by roadside objects in which case it could be mistaken as an object on the lane. To tackle the problems of misinterpretation a rule-based module helps to categorize different states for which models are applied. The distance to the object sensed, the road conditions, the speed, and the acceleration all play a role in the design of these rules.

The desired state occurs when the lead vehicle either follows another platoon in a safe distance or the next object is out of range. Other states are considered as transient. These cases can be accommodated by the methods described for the hazardous state of the follower vehicle but with a different set of sensor signals. An example for the use of the constant velocity model is resembled by the case when a lead vehicle moving with a constant velocity detects a stationery object in its path. For the time instant between detecting the object and taking corrective action to its findings (such as braking), the constant velocity model would be active. Once the deceleration procedure is initiated, the constant (negative) acceleration model would be active. The rule-base module is used for switching between these models.

2.4 Deviation from the Process Model: Sensor Validation

A sensor validation scheme should fulfill the following two tasks: detection and diagnosis. The former involves discovering a malfunction in a sensor while the latter can be subdivided into three stages: localization (establishing which sensor has failed), identification (determining the type of failure) and estimation (indicating the effect and extent of failure) (Young and Clarke, 1989). We use two methods for sensor validation, one is model based, while the other does not rely on a model. The latter is based on first principles and on physical constraints while the model based validation uses Kalman filtering techniques.

2.4.1 Algorithmic Sensor Validation

We first use the Algorithmic Sensor Validation filter (ASV), a technique which is not model-(operating states of the vehicles developed in the previous section) based. This is useful for detecting outlier readings. It compares the difference between the sensor readings and the validated reading at the previous sampling time to the maximum possible change that is possible in the relative distance between the two vehicles in one sampling time. This is obtained by looking at the physical constraints of the system. For example, the acceleration range of a vehicle lies between [-5,2] m/s^2 (during normal operation, the range is even smaller). Therefore, the maximum relative acceleration between two vehicles (assuming worst case) is 7 m/s^2. Knowing the approximate operating velocity of the vehicle (between [0, 30] m/s) one can calculate the maximum possible change in distance between the two vehicles. In general, an upper absolute bound can be obtained by

$$\left|x(k)-x(k-1)\right|_{\max}=u_{\max.rel}T+\frac{1}{2}a_{rel.\max}T^2 \qquad (19)$$

where

$u_{\max.rel}$ is the maximum relative velocity possible between the two vehicles

$a_{rel.\max}$ is the maximum relative acceleration between the vehicle and the target

This technique is active at all times and is not bound to a particular state or model. Therefore, it is the first check within the system for process deviations.

2.4.2 Model Based Validation

After the ASV, the model based validation takes place. In particular, a validation gate is obtained by using a Kalman filtering estimate applied to an appropriate model for the vehicle state. Readings that lie outside the validation gate are classified as faulty. Unlike the ASV filter, the validation gate is possible only when the vehicle is in a state for which a model exists. First, the principle of Kalman filtering is outlined and then the process of formation of the validation gates is shown. An estimator computes an estimate for the parameter of interest based on a combination of the observations. There are a number of methods employed for parameter estimation, such as estimation based on likelihood, Bayesian estimate, Least square estimate, and the minimum mean-square error estimation. These four techniques are equivalent under the assumptions employed by the Kalman filter. The Kalman filter is a form of optimal estimation (in the statistical sense) characterized by recursive (i.e. incremental) evaluation, an internal model of the dynamics of the system being estimated, and a dynamic weighting of incoming evidence with ongoing expectation that produces estimates of the state of the observed system. The *a priori* information to the filter is the system dynamics (presented in the previous section) and the noise property of the system and the measurements that can be estimated from the historic data.

2.4.2.1 Kalman Filtering

We begin by reviewing the principle of Kalman filtering which is used in the validation process (Bar-Shalom and Fortmann, 1988, Chui and Chen, 1991, Bar-Shalom and Li, 1993, Grewal and Andrews, 1993). Consider a discrete time dynamic system described by

$$\mathbf{x}(k+1) = \mathbf{F}(k)\mathbf{x}(k) + \mathbf{G}(k)\mathbf{u}(k) + \mathbf{v}(k) \tag{20}$$

where

- **x(**k**)** is the state at the time k
- **u(**k**)** is the (known) input or control signal
- **v(**k**)** is a sequence of zero-mean, white, Gaussian process noise with covariance **Q(**k**)**
- **F** is the system model
- **G** is the gain through which the input is multiplied

A number of sensors *i = 1, .. , m*, are considered to take observations $\mathbf{z}_i$(k)of the state according to the observation equation

$$\mathbf{z}(k) = \mathbf{H}(k)\mathbf{x}(k) + \mathbf{w}(k) \tag{21}$$

where

- **z(***k***) =** $\left[\mathbf{z}_{\mathbf{1}}^{\mathbf{T}}(k),..,\mathbf{z}_{\mathbf{m}}^{\mathbf{T}}(k)\right]$ is the stacked observation vector
- **w(***k***)** is a sequence of zero-mean white Gaussian measurement noise with covariance **R(**k**)**
- **H** is the observation model

The initial state is assumed to be Gaussian with mean $\hat{\mathbf{x}}(0|0)$ and covariance **P(**0|0**).** The two noise sequences and the initial state are assumed to be independent, i.e. we assume

$$E[\mathbf{w}(k)] = E[\mathbf{v}(k)] = \mathbf{0} \tag{22}$$

$$E[\mathbf{w}(k)\mathbf{w}^{T}(j)] = R\delta_{kj} \tag{23}$$

$$E[\mathbf{v}(\mathbf{k})\mathbf{v}^{\mathbf{T}}(j)] = Q(\mathbf{k})\delta_{kj} \tag{24}$$

$$E[\mathbf{w}(k)\mathbf{v}^{\mathbf{T}}(j)] = \mathbf{0}\,. \tag{25}$$

For the above system the Kalman filter provides a recursive solution for the estimate $\hat{\mathbf{x}}(k|k)$ of the state **x(**k**)** in terms of the estimate $\hat{\mathbf{x}}(k-1\,|\,k-1)$ and the new measurements **z**(k).

The one step prediction of the state is

$$\hat{\mathbf{x}}(k+1|k) = \mathbf{F}(k)\hat{\mathbf{x}}(k|k) + \mathbf{G}(k)\mathbf{u}(k) \tag{26}$$

$$\hat{\mathbf{x}}(k+1|k+1) = \hat{\mathbf{x}}(k+1|k) + \mathbf{W}(k+1)\mathbf{v}(k+1) \tag{27}$$

$$\mathbf{v}(k+1) = \mathbf{z}(k+1) - \mathbf{H}(k+1)\hat{\mathbf{x}}(k+1|k) \tag{28}$$

$\mathbf{v}(k+1)$ is called the innovation or measurement residual. The filter gain **W(**k+1**)** is

$$\mathbf{W}(k+1) = \mathbf{P}(k+1|k)\mathbf{H}^{\mathrm{T}}(k+1)\mathbf{S}^{-1}(k+1) \tag{29}$$

where

- **P(**k+1**|**k**)** is the one step prediction covariance
- **S**(k+1) is the measurement prediction covariance

$$\mathbf{P}(k+1|k) \overset{\Delta}{=} E\left[\tilde{\mathbf{x}}(k+1|k)\tilde{\mathbf{x}}^{\mathrm{T}}(k+1|k)|\mathbf{z}(1)..\mathbf{z}(k)\right] = \mathbf{F}(k)\mathbf{P}(k|k)\mathbf{F}'(k) + \mathbf{Q}(k) \tag{30}$$

$$\tilde{\mathbf{x}}(k+1|k) \overset{\Delta}{=} \mathbf{x}(k+1|k) - \hat{\mathbf{x}}(k+1|k) = \mathbf{F}(k)\tilde{\mathbf{x}}(k|k) + \mathbf{v}(k) \tag{31}$$

The measurement prediction covariance is

$$\mathbf{S}(k+1)\overset{\Delta}{=}E\left[\tilde{\mathbf{z}}(k+1|k)\tilde{\mathbf{z}}^{\mathrm{T}}(k+1|k)|\mathbf{z}(1)..\mathbf{z}(k)\right]=\mathbf{H}(k+1)\mathbf{P}(k+1)\mathbf{H}^{\mathrm{T}}(k+1)+\mathbf{R}(k+1) \quad (32)$$

Fig. 9 shows the flow chart of a Kalman filter.

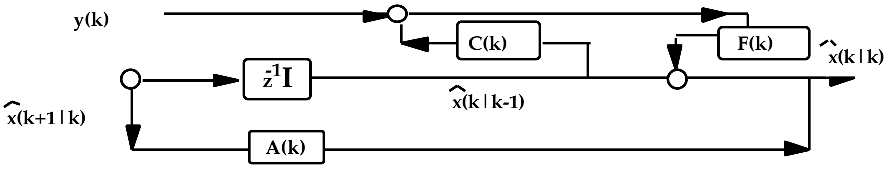


Fig. 9: Flow Chart of Kalman Filter

2.4.2.2 Validation Gates

For the longitudinal sensors the following method is used for validation. It is assumed that the true measurement of the distance between the vehicles at sampling time *k+1* is normally distributed, conditional on the sensor readings up to sample *k,* i.e.

$$p\left[\mathbf{z}(k+1)|\,\mathbf{z}(1)..\mathbf{z}(k)\right]=N\left[\mathbf{z}(k+1);\hat{\mathbf{z}}(k+1|k),\mathbf{S}(k+1)\right] \quad (33)$$

where

S(k+1) is the associated measurement prediction covariance matrix obtained by the Kalman filtering process as given in (13).

Based on this one can define a region in the measurement space where the measurement will be found with some (high) probability (for example a 3 sigma bound corresponds to a confidence of 99.8%)

$$\begin{aligned}\tilde{V}_{k+1}(\gamma)&\overset{\Delta}{=}\left\{\mathbf{z}\!:\left[\mathbf{z}-\hat{\mathbf{z}}(k+1)|k\right]\mathbf{S}^{-1}(k+1)\left[\mathbf{z}-\hat{\mathbf{z}}(k+1|k)\right]\le\gamma\right\} \quad (34)\\ &=\left\{\mathbf{z}\!:\nu'(k+1)\mathbf{S}^{-1}(k+1)\nu(k+1)\le\gamma\right\}\end{aligned}$$

where

$\tilde{V}_{k+1}(\gamma)$ is the region defined above is called the validation region or the gate. It is the ellipse (or ellipsoid) of probability concentration-the region of minimum volume that contains a given probability of mass under the Gaussian assumption as shown in Fig. . Measurements that lie within the gate are considered valid; those outside are labeled as questionable.

The parameter γ is obtained from tables of chi-square distribution (example γ =9 corresponds to a validation gate with a confidence of 99.8%). This validation process limits the region in the measurement space where the next measurement should be present. Measurements outside the validation region are too far from the expected location and thus are very unlikely.

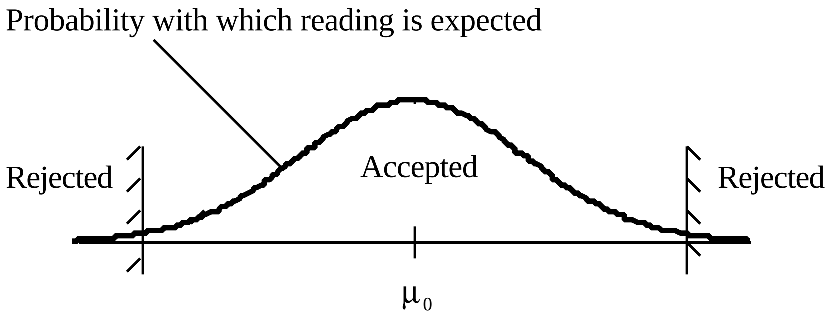


Fig. 10: Validation Gate

2.5 Data Fusion:

Data fusion can be tackled by several different methods. For example by Generalized Evidence Processing Theory (Thomopoulos, 1990), Bayesian (Agogino et al., 1988; Kim et al., 1992) and Dempster-Shafer methods (Blackman 1990) and linear estimators (Ayache and Faugeras 1988). After sensor validation the set of measurements consist of correct and incorrect measurements the latter originating from clutter or false alarms. Clutter means detections or returns from nearby objects, weather, electromagnetic interferences, acoustic anomalies, false alarms, etc., that are generally random in number, location and intensity.

The simplest approach for tracking a target in a cluttered environment is to select the validated measurement that is closest to the predicted measurement and use it in the tracking filter as if it were the correct one. This results in the nearest neighbor standard filter (NNSF).

2.5.1 Using *a Priori* Information

The fusion process is similar to the one presented by Hashemipour et al. (1988). Here, the readings are fused together using the *a priori* information available about each of the nodes. Let, $(y_1(k),..,y_m(k))$ be the validated redundancy measurements that lie in the validation region for each variable and $x_j(k+1|k)$ be the local estimates obtained by using the Kalman filter equations and these *m* estimates. These, are then fused together to obtain the centralized global estimate $\hat{x}(k+1|k+1)$ using

$$\hat{x}(k+1|k+1) = Z(k+1)$$

$$\begin{bmatrix} M^{-1}(k+1)\hat{x}(k+1|k) + \\ \sum_{j=1}^{m}\left[Z_j^{-1}(k+1)\hat{x}_j(k+1|k+1) - M_j^{-1}(k+1)\hat{x}_j(k+1|k)\right] \end{bmatrix}$$

$$C^T(k)V^{-1}(k)C(k) = \sum_{j=1}^{m} C_j^T V_j^{-1} C_j$$

$$M(k+1) = A(k)Z(k)A^T(k) + B_w(k)W(k)B_w^T(k), \quad M(0) = X_0$$

$$Z^{-1}(k+1) = M^{-1}(k+1) + \sum_{j=1}^{m}\left[Z_j^{-1}(k+1) - M_j^{-1}(k+1)\right]$$

For validation each node sets up a validation region given by

$$\nu_j^T(k+1)S_j^{-1}(k+1)\nu_j(k+1) \le \gamma$$

$$\nu_j(k+1) = y_j(k+1) - C(k+1)\hat{x}(k+1|k)$$

$$S_j(k+1) = V_j(k+1) + C_j(k+1)M(k+1)C_j^T(k+1)$$

where V_j is the variance of the sensor noise at each node, *M(k+1)* is the covariance of the *central (global estimate)* obtained by a process enumerated above.

2.5.2 Using *a Posteriori* Information

It may be possible that more than one measurement lies in the validated region. This implies that all of these could have originated from the target. Hence, one should try and use all these measurements for estimating the distance of the target. One method to do this is to split the track into multiple hypotheses every time more than one measurement is found in the validation region. The likelihood function can then be used for deciding which hypothesis to accept. We use the Bayesian approach for data fusion which is suitable for this application as it can be effectively implemented in real time. Here, the probabilities of the validated readings being correct are calculated. These probabilities are calculated for each sequence of measurements by using the Probabilistic Data Association Filter (Bar-Shalom, 1990), which is used for the IVHS system.

Let the set of validated measurements at time *k* be

$$\mathbf{Z}(k) \overset{\Delta}{=} \{\mathbf{z_i}(k)\}_{i=1}^{m_k} \qquad (35)$$

where

Z(k) is the set of validated measurements

zi(k) is the validated measurement

m_k (which is also a random variable) is the number of measurements in the validation region.

The cumulative set of measurements is denoted as $Z^k \overset{\Delta}{=} \{Z(j)\}_{j=1}^{k}$

In PDAF we make the following assumption about the past (for ease of computation)

$$p[\mathbf{x}(k)|Z^{k-1}] = N[\mathbf{x}(k); \hat{\mathbf{x}}(k|k-1), \mathbf{P}(k|k-1)] \qquad (36)$$

i.e. the state is assumed to be normally distributed (Gaussian) according to the latest estimate and covariance matrix. We define the following events

$\theta_i(k) \overset{\Delta}{=} \{\mathbf{z_i}(k)\}$ is the target originated measurement, $i = 1,..,m_k$

$\theta_0(k) \overset{\Delta}{=}$ none of the measurements at time *k* is target originated

Let the probabilities with which these events occur be denoted as

$$\beta_i(k) \overset{\Delta}{=} P\{\theta_i(k) \mid Z^k\}, \ i = 0,1,..,m_k \qquad (37)$$

conditioned on Z^k. In view of the above assumption, these events are mutually exclusive and exhaustive, hence

$$\sum_{i=0}^{m_k} \beta_i(k) = 1 \qquad (38)$$

The procedure that yields these probabilities is known as the probabilistic data association (PDA). The conditional mean of the state at the time *k* is

$$\hat{\mathbf{x}}(k\,|\,k) = E\left[\mathbf{x}(k)\,|\,Z^k\right] = \sum_{i=0}^{m_k} E\left[\mathbf{x}(k)\,|\,\theta_i(k), Z^k\right] P\{\theta_i(k)\,|\,k\}$$

$$= \sum_{i=0}^{m_k} \hat{\mathbf{x}}(k\,|\,k)\beta_i(k) \tag{39}$$

where $\hat{\mathbf{x}}(k\,|\,k)$ is the updated state estimate conditioned on the event $\theta_i(k)$ that the i^{th} validated measurement is correct. From the Kalman Filter estimate

$$\hat{\mathbf{x}}(k|k) = \hat{\mathbf{x}}(k|k-1) + \mathbf{W}(k)\nu_i(k), \; i = 1,..,m_k \tag{40}$$

where

$$\nu_i(k) \overset{\Delta}{=} \mathbf{z}_i(k) - \hat{\mathbf{z}}(k\,|k-1) \tag{41}$$

is the corresponding innovation. For i=0, i.e. if none of the measurements is correct, the estimate is

$$\hat{\mathbf{x}}_0(k\,|\,k) = \hat{\mathbf{x}}(k\,|\,k-1) \tag{42}$$

Hence, the state update equation of the PDAF is

$$\hat{\mathbf{x}}(k|k) = \hat{\mathbf{x}}(k|k-1) + \mathbf{W}(k)\nu(k) \tag{43}$$

$$\nu(k) \overset{\Delta}{=} \sum_{i=1}^{m_k} \beta_i(k)\nu_i(k) \tag{44}$$

where $\nu_i(k)$ is the combined innovation.

The error covariance associated with the updated state estimate is

$$\mathbf{P}(k|k) = \beta_0(k)\mathbf{P}(k|k-1) + \left[1 - \beta_0(k)\right]\mathbf{P}^{\mathrm{c}}(k|k) + \tilde{\mathbf{P}}(k) \tag{45}$$

$$\tilde{\mathbf{P}}(k) = \mathbf{W}(k)\left[\sum_{i=1}^{m_k} \beta_i(k)\nu_i(k)\nu^{\mathrm{T}}(k) - \nu(k)\nu^{\mathrm{T}}(k)\right]\mathbf{W}'(k) \tag{46}$$

$$\mathbf{P}^c(k|k) \overset{\Delta}{=} \left[\mathbf{I} - \mathbf{W}(k)\mathbf{H}(k)\right]\mathbf{P}(k|k-1) \tag{47}$$

A probabilistic inference is made on the number of measurements in the validation region, i.e.

$$\beta_i(k) \overset{\Delta}{=} P\{\theta_i(k)\,|\,Z^k\}$$

$$= P\{\theta_i(k)\,|\,Z(k), m_k, Z^{k-1}\} \qquad i = 0,1,..,m_k \tag{48}$$

Using Bayes' rule

$$\beta_i(k) = \frac{1}{c}\, p\left[Z(k)|\theta_i(k), m_k, Z^{k-1}\right] P\{\theta_i(k)|m_k, Z^{k-1}\} \qquad i = 0,1,.., m_k \tag{49}$$

The associated probabilities are given by

$$\beta_i(k) = \frac{e_i}{b + \sum_{j=1}^{m_k} e_j} \quad \& \quad \beta_0(k) = \frac{b}{b + \sum_{j=1}^{m_k} e_j} \tag{50}$$

where

$$e_i \overset{\Delta}{=} e^{\left[-\frac{1}{2}\nu_i'(k)\mathbf{S}^{-1}(k)\nu_i(k)\right]} \tag{51}$$

$$b \overset{\Delta}{=} \left(\frac{2\pi}{\gamma}\right)^{\frac{n_z}{2}} \lambda V_k c_{n_z}\left(1 - P_D P_G\right) / P_D \tag{52}$$

P_G is the probability that the correct measurement falls in the validation region
P_D is the target detection probability

n_Z is the dimension of the measurement **z** and c_{nZ} is the volume of the n_Z dimensional hypersphere (c_1=2, c_2= p, c_3=4 p/3, etc.) (Fortman et al., 1983)

Assuming a Poisson density the probabilities of the events conditioned only on the number of validated measurements are

$$\gamma_i(m_k) = \begin{cases} \dfrac{P_D P_G}{P_D P_G m_k + (1 - P_D P_G)\lambda V_k{}^{\mathrm{T}}} & i = 1,..,m_k \\ \dfrac{(1 - P_D P_G)\lambda V_k}{P_D P_G m_k + (1 - P_D P_G)\lambda V_k{}^{\mathrm{T}}} & i = 0 \end{cases} \qquad (53)$$

The volume of the elliptical (i.e. Gaussian based) validation region is $V_k = c_{n_z} \gamma^{n_z/2} |S(k)|^{1/2}$

2.6 Rule Based Module

The multifunctional rule-based module performs model selection and assists in the sensor validation process. It also detects sensor bias and the degradation of sensors. For model selection, it uses information from the longitudinal sensors as well as from the communication channel. This way, it can determine whether a preceding vehicle (or other object) is moving with a constant speed or acceleration. Information about hazards found in preceding vehicles are transmitted to the follower vehicles and can be used for this purpose. Signals from the longitudinal sensors of the lead vehicle are difficult to interpret because roadside objects and vehicles on other lanes can be misinterpreted as objects on the lane. The distance and allowable bandwidth of the sensor will give information about the validity of such readings. Also of importance is roadside information which gives information, for example about the radius of the road which allows for a potential correction of the reading. Depending on which sensors will be used for the lead vehicle, the rule-based module will be modified and supplemented.

To detect abnormal operation of the sensors and for testing malfunction hypothesis we use validation modules. The rule-based system also contains specifications about the various sensors, such as the measurement range, accuracy, effect of changing environment on sensor performance etc. Furthermore, complete malfunction (failure) of the sensors is relatively easy to detect, but when failure occurs, it can lead to catastrophic events. Therefore, it is imperative to detect latent malfunctions in the sensor to predict the degradation in the sensor performance. For this purpose a sliding window (set of several successive measurements) can be used. The statistical properties of the measurement residue $w^T(k)$ (which is the *fused* Kalman filter estimate for the relative distance between the vehicles given by $w^T(k)=\mathbf{z}(k)-\hat{x}(k/k)$)). A failure signature typically takes the form of residual biases that characterize the specific failure (Ray and Luck, 1991). It is relatively simple to check for sensor bias when multiple sensors are present. The measurement residue ideally should be zero mean, white and Gaussian. An estimate of the sensor bias can be obtained by looking at the mean of measurement residue over a number of sliding windows. If the mean is non zero and remains constant then it can be attributed to the bias in the sensor. This can be tested by a test of hypothesis[1]: there is a change in the mean of measurement residue over a number of successive sliding windows. The statistical properties mean, variance and whiteness for various sliding windows can be used to detect any changes in the sensor performance by comparing it against those under normal conditions and for fault signatures contained in the rule-based system. For example, a gradual increase in the variance over time can be a symptom for degradation of the sensor performance. To detect slow variations in the sensor performance the drift test can be carried out periodically off-line (Pouliezos and Stavrakakis, 1994). For this the steady state test which aims at determining whether the examined variables are in static or dynamic state is carried out. For the drift test, the steady state test is first carried out for two different windows: small and large. If the steady state test, utilizing the small window detects a steady state for a duration equal to the large window and if for the same duration the steady state test utilizing the large window detects a dynamic state,

[1] Another methods for this (for example using dynamic belief networks) can be found in Alag, Agogino, and Hsueh 1995.

then a drift is present.

3. Example

To illustrate the methodology we use three examples. The first example explains in detail the validation, fusion and sensor bias detection methodology. The second example shows the validation procedure applied to data obtained from platooning tests carried out at high speed on the freeway (here, a single sensor was used, hence the fusion methodology cannot be illustrated). The third example is to illustrate the fusion process for the constant velocity process. The data was obtained from platooning test set ups using a radar, a sonar, and an optical sensor.

3.1 Constant Distance Case

To illustrate the methodology proposed here, we first consider the case of constant distance in a platoon. In a normal platooning operation, the possible changes in the distance between the vehicles (process deviation) can be estimated as

$$\Delta x = \Delta u_{relative} T + \frac{1}{2} \Delta a_{relative} T^2 \qquad (54)$$

where

$\Delta u_{relative}$ is the relative velocity between the vehicles

$\Delta a_{relative}$ is the relative acceleration between the vehicles

Assuming a sampling time of 0.02 seconds, a conservative estimate for change in distance is 0.02 m under normal conditions. Hence, for the simulations the covariance of the process **Q** was taken as .02*.02. The initial distance between the vehicles was taken as 4 meters. Three sensors are used with data generated as follows. The covariance of the sensor noise **R** was taken as 0.5 for all three sensors for the first fifty samples. The covariance of the sensor noise for sensor 1 was increased to 1 for samples 50 to 80. The same was done for sensors 2 and 3 for samples 60 to 80 and samples 70 to 80. respectively. The process covariance was changed to .04*.04 for samples 80 to 90 and to .01*.01 from 90 to 100. A sensor bias of 0.25 meters was introduced in sensor 1 from sample 100 to 150. This shows the robustness of the method to unmodeled disturbances. The outputs of the three sensors and the actual distance is shown in Fig. 11.

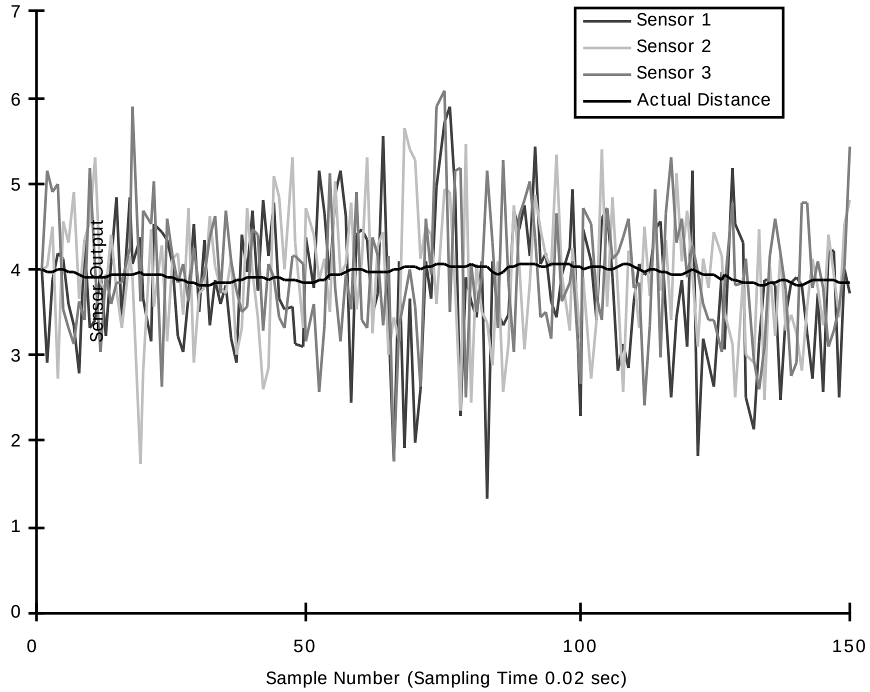


Fig. 11: The outputs from the three sensors and the actual value of the process used in example 1.

Fig. 12 shows the fused estimate along with the actual value of the process and an estimate obtained by averaging the values of the three sensors. As can be seen clearly the fused estimate follows the actual process value very well, in spite of unmodeled disturbances and changes in the process.

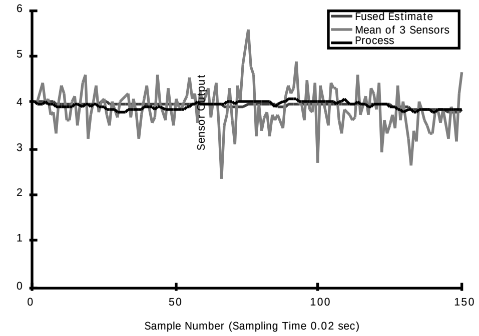


Fig. 12: Fused estimate, the average values from the sensors and the actual process value

Fig. 13 and Fig. 14a show the normalized innovations for the three sensors. Here, a validation gate corresponding to a confidence of 95.9% (innovation should be less than 6) was used for the sensor validation process.

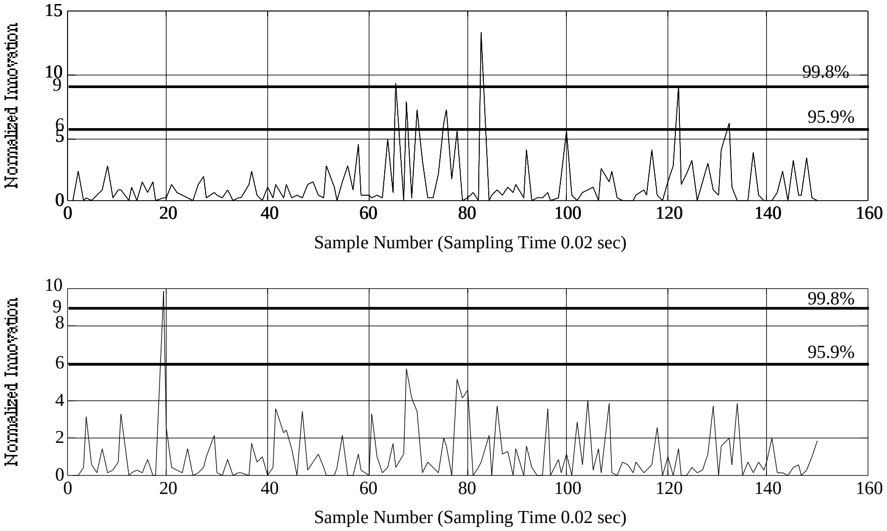


Fig. 13: Normalized Innovations for Sensor 1 and 2 used in the validation process

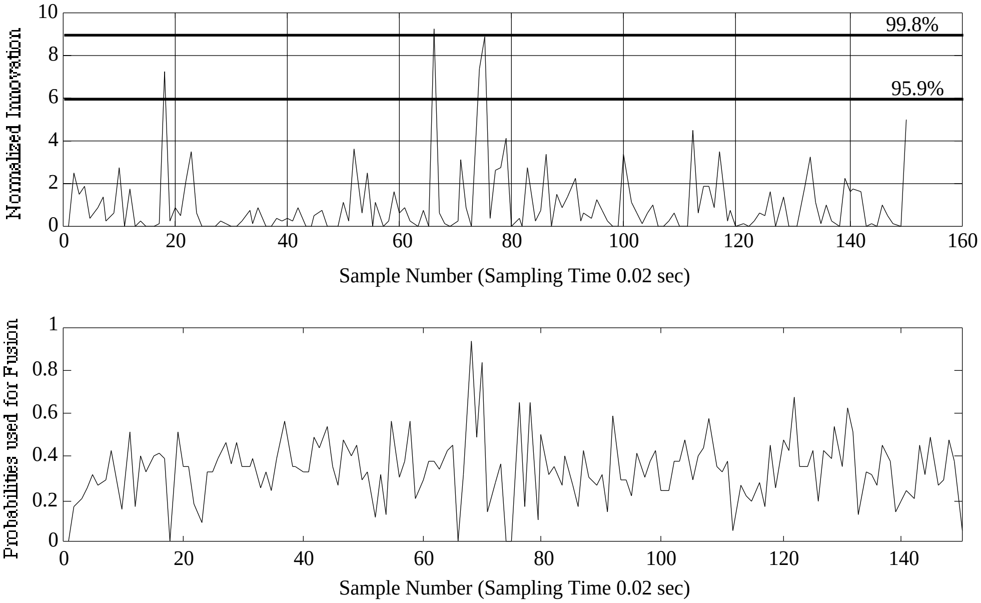


Fig. 14: a. Normalized innovations for sensor 3 used in the validation process
b. Probability with which sensor value 3 was used in the fusion process

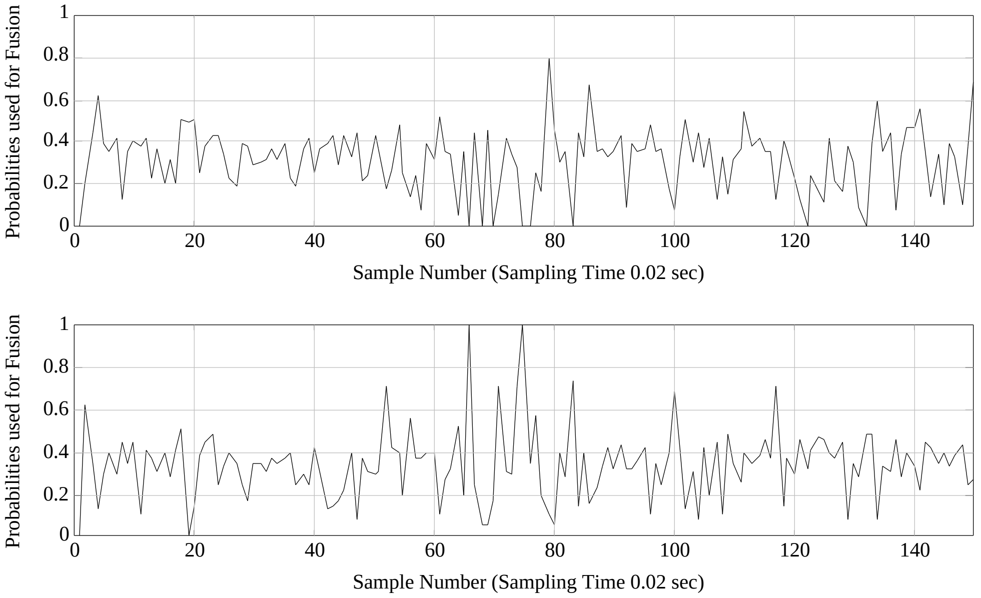


Fig. 15: Probabilities with which sensor readings 1 and 2 were used in the fusion process

To illustrate the sensor bias detection methodology a bias of -0.25 meter was introduced in the readings of sensor 1 from samples 101 to 150. Fig. 16a shows the sensor 1 residue (difference of sensor output and the *fused* estimate) for the first 100 samples, while Fig. 16b shows the sensor 1 residue for the remaining samples. As stated earlier, in the absence of sensor bias the sensor residue should be ideally zero. An estimate for the sensor bias can be obtained by the magnitude of the mean of sensor bias. Since sensor 1 readings (up to sample 100), sensor 2 and 3 readings were simulated so as not to have a bias, the mean of their residues should be close to zero. It is -0.0867 for the first 100 readings for sensor 1 and it changes to -.3246 for the next 50 readings (for which a bias of -.25 was introduced). The means of sensor residues for sensor 2 and 3 are -.0.0319 and 0.0055 which are close to 0 as expected. Fig. 16 shows the sensor residue for sensor 1, while Fig.

17 shows the sensor residue for sensor 2 and 3.

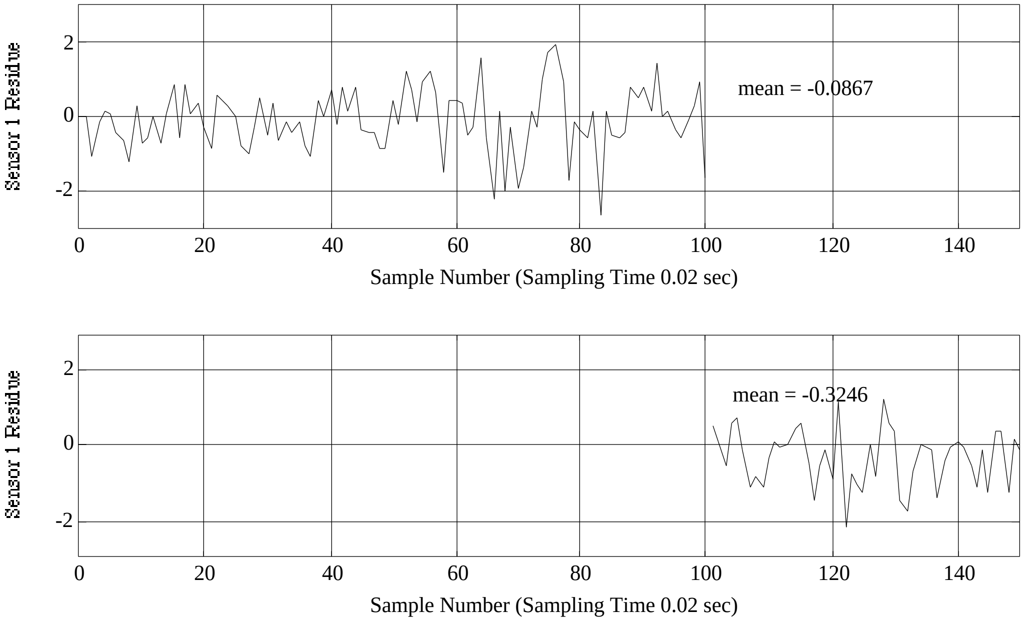


Fig. 16: Sensor residue for sensor 1.

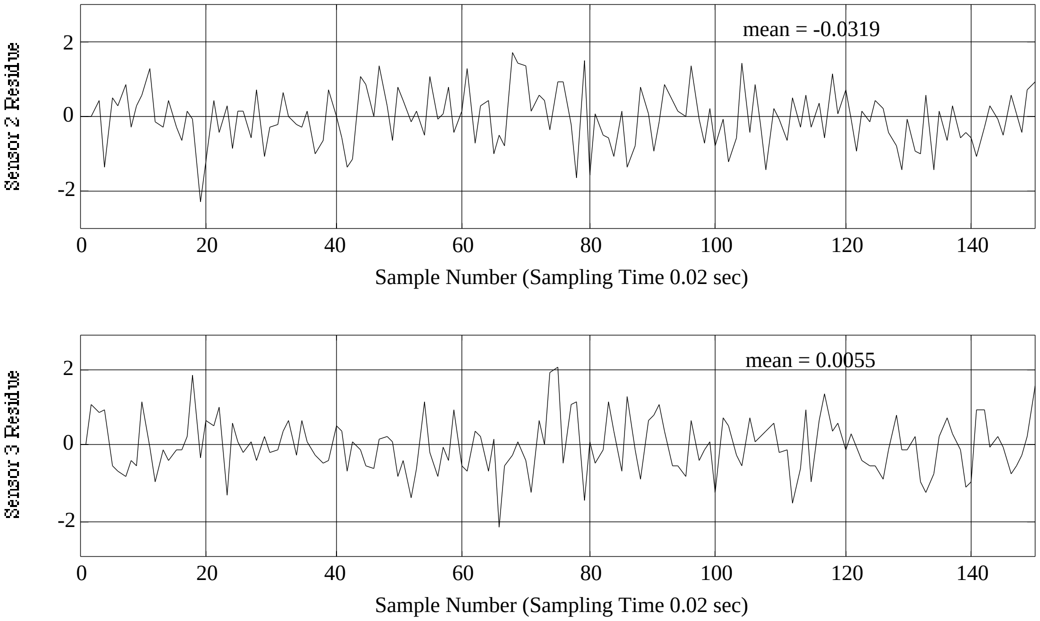


Fig. 17: Sensor residue for sensors 2 and 3

3.2 Follower Vehicle: Normal Operation within a Platoon

The platooning tests were carried out recently in San Diego. The fixed distance that the vehicles were trying to maintain was 4 meters. Fig. 18 shows the sensor output along with the estimate from the Kalman filter, for data used in normal platooning operation. A constant gain Kalman filter was used for this purpose, with Q= .02*.02 and R=.01. Fig. 19 shows the normalized innovation which is used for the sensor validation. For a confidence of 99.8%, the normalized innovation should be less than 9, while for a confidence level of 95.9% the normalized innovation should be less than 6.

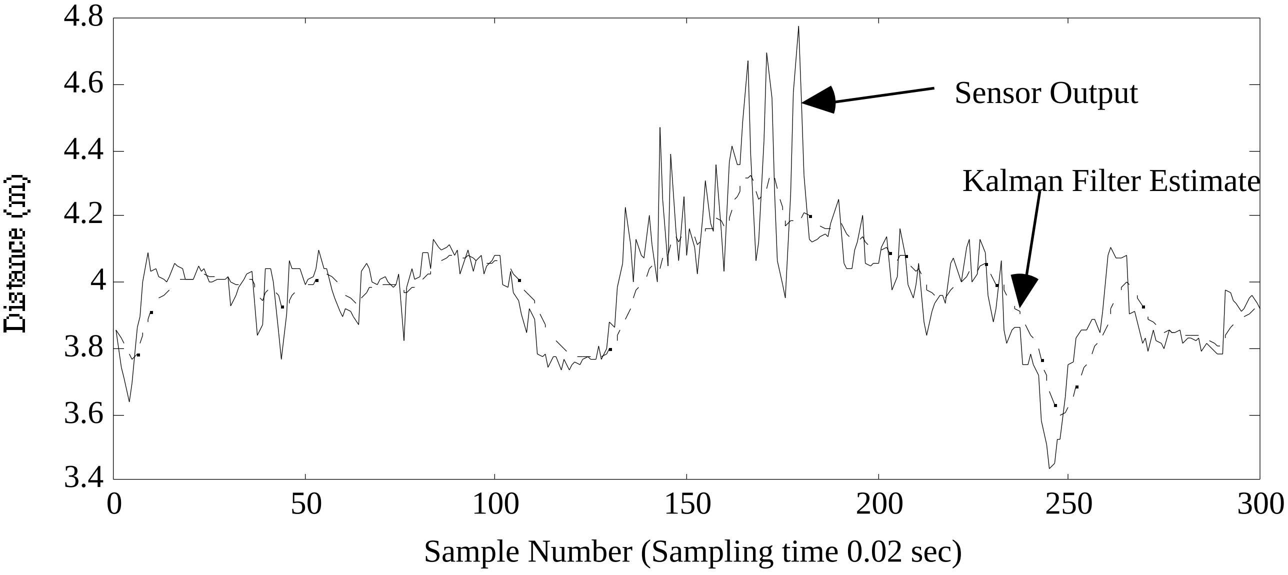
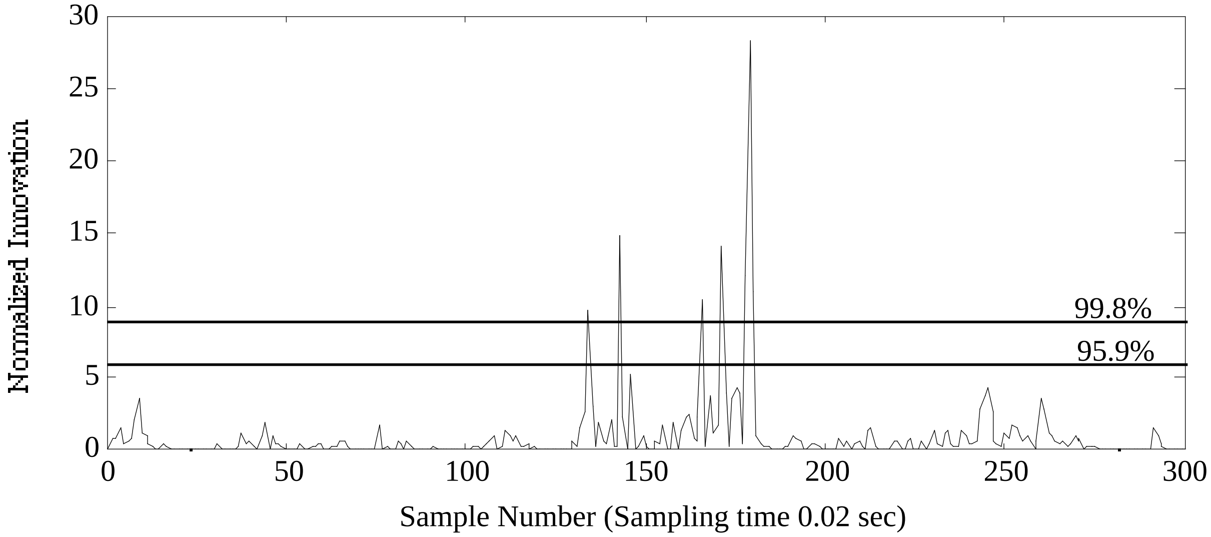


Fig. 18: Sensor Output and the Kalman Filter Estimate using the Normal Platooning Model



Fi

g. 19: Normalized Innovations used for Sensor Validation

3.3 Lead Vehicles: Constant Velocity Case

The data fusion procedure is shown for the constant velocity case on data collected during experiments with fixed object on the lane and a moving lead vehicle. Here, output for three sensors namely sonar, radar and an optical sensor were fused using the PDAF algorithm. For this purpose, the constant velocity model was used. The process noise covariance was taken as 1, and the covariance of the sonar, radar and optical sensor was taken as .0001, .001 and .01, respectively. A gate with a confidence of 99.8% was used. Fig. 20 shows the sensor readings and the PDAF value. Fig. 21 illustrates how one can switch from a second order model to a third order model without loss of the target. It has to be kept in mind that all these models are approximations and hence do not totally match reality. Therefore, it is important to judiciously match the operational state with the model. If a model of higher order than required is used, the estimation error increases with the complexity. For example, is there is no relative acceleration and one uses a third order Wiener model, extimation error for both position and velocity increase. As a rule of thumb (Bar-Shalom, 1993), a third order nearly constant acceleration model is applicable when changes in acceleration are small compared to the acceleration.

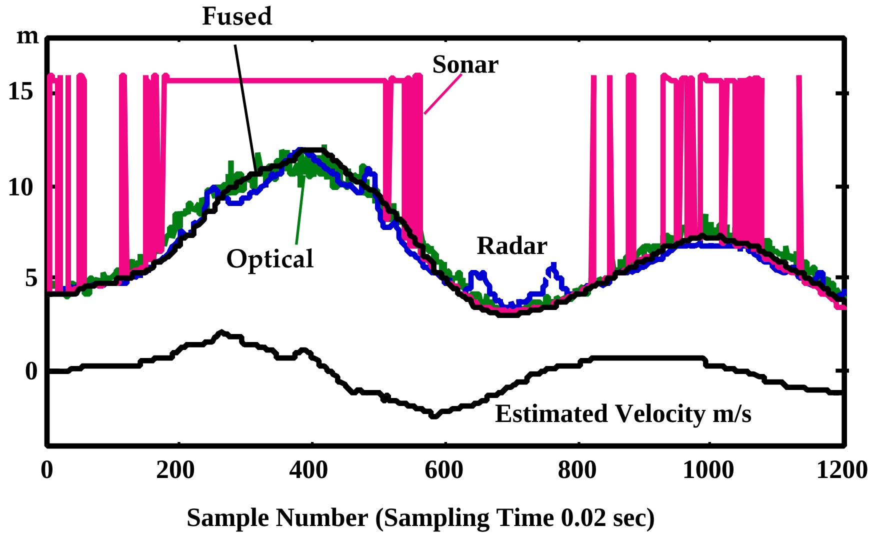


Fig. 20: Sensor Output and PDAF value for data fusion

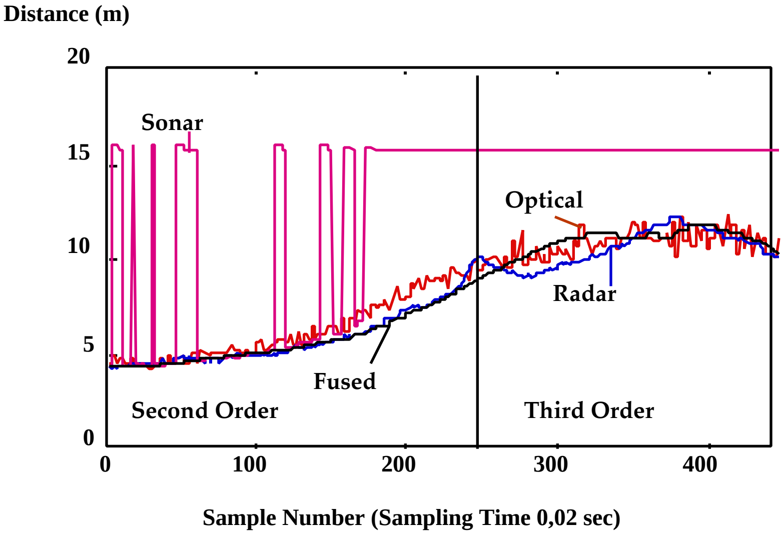


Fig. 21: Switching Between Models

4.0 Fault Detection and Diagnosis in the Vehicle

For the safety of the IVHS it is imperative to detect and diagnose faults in the actuator, sensors, controllers and in the vehicle itself (such as a tire burst). A common and important fault detection method is the so called model-based fault detection. This approach basically relies on analytical redundancies which arises from the use of analytical relationships describing the dynamic interconnection between various system components. As opposed to physical or hardware redundancy which uses measurements from redundant sensors for fault detection purposes, analytical redundancy is based on the signals generated by the mathematical model of the system being considered. These signals are then compared with the actual measurement obtained from the system. The difference between these two signals are know as *residues*. The decision as to whether a fault has occurred is made on the values of the residues. In the absence of noise and modeling error, the residual vector is equal to zero under fault-free conditions. Hence, a non zero value of the residual vector indicates the existence of the faults. The effect of faults has to be separated from noise and modeling errors. In the simplest case this is done by comparing the residual magnitudes with threshold values (Duyar and Merrill, 1992, Duyar et al., 1994).

There are three different ways of generating fault-accentuated signals using analytical redundancy: parity checks, observer schemes, and detection filters, all of them using state estimation techniques. The resulting signals are used to form decision functions as, for example likelihood functions. The basis for the decision on the occurrence of a fault is the fault signature, i.e. a signal that is obtained from some kind of faulty system model defining the effects associated with a fault. A diagnostic model is obtained by defining the residual vector in such a manner that its direction is associated with known fault signature. Furthermore, each signature has to be unique to one fault to accomplish fault isolation. The drawback of these estimation techniques is that an accurate system model is normally required.

Rule based expert systems have also been investigated (Paasch and Agogino, 1993) for sensor-based fault detection and diagnosis problems. Fault diagnosis using rule-based expert system needs an extensive database of rules and the accuracy of the diagnosis depends on the rules. Pattern recognition techniques, using neural networks are particularly well suited for fault diagnosis when the process model is not known or is very complicated (Agogino et al., 1992, Sorsa and Koivo, 1993).

Associated with any method used for fault diagnosis and more generally with the state of the automated vehicle in IVHS are sources of uncertainty. In order to diagnose the vehicle state (for the remaining two modules), it is necessary to account for these sources of uncertainty and propagate them to the final diagnosis of the vehicle state. It is therefore necessary to use a method that can represent the uncertainty inherent in the system. Representation of uncertainty is a central issue in artificial intelligence and has been addressed in many different ways. Some of the approaches used in representing uncertainty include the Bayesian approach, the certainty factor model, belief functions based on Dempster-Shafer theory of evidence, possibility theory, non-monotonic logic, etc. (Kruse and Clark, 1993). Our belief is that for the IVHS system an eclectic approach of these methods is probably required. One such method which is very suitable for IVHS implementation is the Bayesian method of influence diagrams. Here, influence diagrams can be integrated with residual generation to monitor the performance of the vehicle.

4.1 Bayesian Influence Diagrams

Influence diagrams were developed to facilitate automating the modeling of complex decision problems involving uncertainty (Miller et al. 1976, Olmstead 1984, Shachter 1984) The SRI researchers found that influence diagrams provided a compact graphical framework for representing the interrelationships between the variables involved in the problem under consideration. An influence diagram is a graph-theoretic structure for representing and solving decision and probabilistic inference problems. The knowledge representation can be viewed from three hierarchical levels: topological, functional and numerical. At the topological level an influence diagram is an acyclic directed network with nodes representing variables critical to the problem and the arcs representing their interrelationships. The nature of these interrelationships are further specified at the functional level. Formal calculi have been developed for deterministic functions and probabilistic relationships based on either Bayesian and fuzzy probabilities. Finally, the functional form associated with each influence is quantified at the numerical level.

The topological is perhaps the most powerful level of the influence diagram. It is at this level that the structure of complex decision problems can be extracted from domain experts. The intuitive graphical representation consists of nodes and directed arcs. The variables can be of three types: 1) rectangular decision nodes, in which the variable is under the control of the decision-maker, 2) circular state nodes, which corresponds to the uncertain quantities not under the control of decision-maker, and 3) a single diamond-shaped value node, in which the utility function is specified for a decision problem. The interpretation of the relationships represented by the arcs depends on the type of the nodes they connect. Arcs going into state nodes represent *conditional influences* as shown in Figs. 22 a,b&d. The influence can be deterministic, probabilistic or fuzzy. Arcs between state nodes can be reversed through legal topological transformations on the diagram according to Bayes rule and providing a cycle is not introduced. Arcs to and from decision nodes serve a different function and can not be reversed without changing the basic structure of the decision model. An influence arc from a decision node to a state node (Fig. 22d) indicates causality in the sense that each decision option restricts the event space of the state variable. Arcs going into decision nodes are *informational* and show which variables will be known at the time a decision is made (Figs. 22c). *No-forgetting* arcs are placed between decision nodes to signify that decisions are sequential in time and the value of past decisions is remembered (Fig. 22e). Arcs into the single value node signify which nodes directly influence the goal (Fig. 22f). An influence diagram without decision nodes is equivalent to a Bayes' belief network .

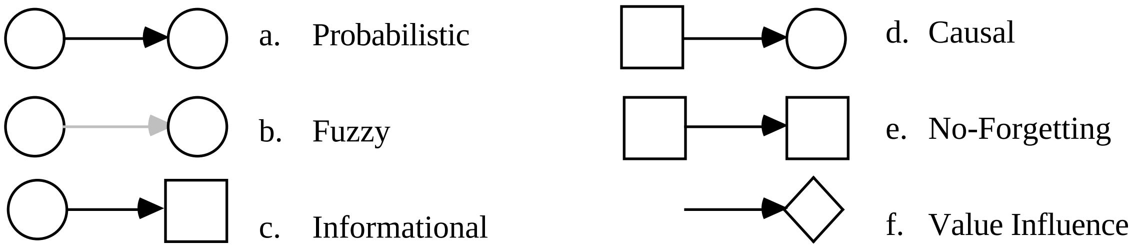


Fig. 22: Six Interpretations of Arcs.

The lack of an arc is in some ways a stronger statement of the modeler's knowledge of the system than the existence of an arc. The presence of an arc indicates that a *possible* dependency exists, while the lack of an arc states strongly that *no dependency* exists. Consider the two node influence diagram in Fig. 23a. The state nodes represent the variables *x* and *y*, and the arc between them indicates that a conditional influence may exist. At a qualitative level, a variable is said to *influence* a state node *y* if information about *x* gives new information about *y*. At a probabilistic level the influence diagram in Fig. 23a represents the following expansion of the joint probabilities of state variables *x* and *y* : Pr(*x, y* | H) = Pr(*y* | *x,* H) Pr(*x* | H), where Pr(*y* | *x* ,H) represents the probability distribution for *y* conditioned on *x* and the history or state of information H. The lack of the arc between the two state nodes in Fig. 23b indicates *conditional independence* ; knowing *x* gives no new information about the state of *y*. The joint probability in Fig. 8b is represented by the expansion: Pr(*x, y* | H) = Pr(*y* | H) Pr(*x* | H).

(a) x influences y (b) x is independent of y

Fig. 23: Two State Node Influence Diagrams

A frequently occurring problem in many industrial situations is what has been termed the *Diagnostician's Problem*. The problem will be stated in general terms and then used to illustrate how this paradigm fits into real time applications in system control.

Consider a system with the set of state variables denoted by $\mathbf{X} = \{x_1, x_2 \ldots x_n\}$. Let $\mathbf{S} \in \mathbf{X}$ be a set of *sensors* or *observables,* i.e. $\{ s_1\ s_2 \ldots s_m \}$ within the system. The diagnostician might be called upon to assess the likelihood for failures of various combinations of hypothesized states given some combination of sensor readings. The problem can be made more tractable with the addition of the set $\mathbf{T} \in \mathbf{S} \in \mathbf{X}$ of *trigger* sensors $\mathbf{T} = \{t_1, t_2, \ldots t_k \}$ which might be Boolean in operation, viz. - on or off. The diagnostician must perform a diagnostic inference given that a certain subset of these trigger sensors go on. Trigger features can play a role as trigger limits for sensor validation.

Viewing the paradigm of the Diagnostician's Problem, in general, three categories of state nodes can be identified.

Sensor Nodes : A sensor node, as mentioned earlier, might represent a measurement sensor that could be directly 'read' by an operator or controlling system. It might also represent a physical state of the system that is immediately obvious to the operator by means of human sensor capabilities, such as sight, hearing or smell.

Possible Failure Nodes : As their name implies, these represent states of physical components in the system which may be the cause or symptom of, or contributes to, the initiation of the diagnostic search. Since a sensor itself may fail, some or all of the sensor nodes may belong to this class.

Intermediate Nodes : These are neither sensor or possible failure nodes - rather they represent nodes which have influences from or to those nodes. They generally represent intangibles in the problem which cannot 'fail' or which are not measured directly or precisely by any sensor in the system. They are useful in modeling complex systems and in providing a structure for updating system parameters with human and statistical knowledge obtained historically over the operation of the system under consideration.

The diagnostic problem does not stop at probabilistic inference. Once the diagnostician has available to him/her the likelihood of certain events, the next stop is to decide what course of action to take. The influence diagram can be extended to include the diagnostic decision and a value node which captures the cost (or utility) of various combinations of events and decisions. The solution to the controller's problem requires finding the set of decisions (or control instructions) that give the optimal expected cost (or utility) as represented by the diamond-shaped value node.

Influence diagrams can be used to monitor the performance of the vehicle. These essentially link symptoms to the vehicular system failures, like a tire burst, controller failure or a sensing failure. This takes a rather microscopic view of matters and requests data from the engine, transmission and other subsystem sensors of each car it is contained in, and the sensors measuring the vehicle characteristics like speed, acceleration, distance between cars and lateral acceleration. At this level of monitoring, those sensors which reflect the state of the individual vehicle are more important than the ones which advise on the possibility of an accident.

4.2 Hazard Analysis

The other level of supervisory control is to monitor the environment around the vehicle. For this purpose we carry out a hazard analysis following Hitchcock’s definition (Hitchcock, 1991). He defines hazards as those factors that, if avoided will mean that accidents cannot occur. Several hazards were investigated and analyzed via fault tree analysis. Five major hazards, and sequence of event that precursor these hazards have been identified. For example, hazard A is defined as the condition when the distance between two vehicles in a platoon is less than the safe interpolation distance. The approach of influence diagram can be used for this purpose. For this a multi-level hierarchical influence diagram is needed to make a statement about the probability of a hazard. The lower level of the hierarchical influence diagram deals with fault diagnosis and this is used as an input to the higher level influence diagram which calculates the probability of the various hazards.

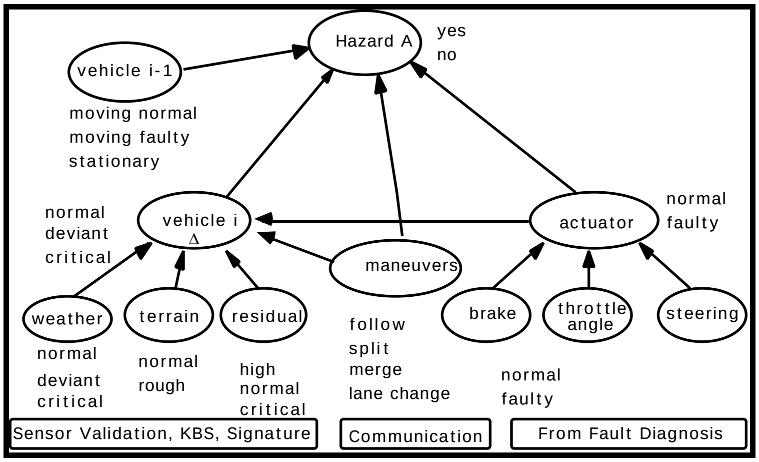


Fig. 24: Influence Diagram for Hazard Analysis

The approach can be best illustrated by an example. Fig. 24 shows a simplified influence diagram that can be used to calculate the probability of a catastrophe occurring (through hazard A) in a specified period of time in the future. There is node (vehicle i) representing the state of the vehicle being monitored. It in this example can take three states {low, normal, critical}. The probability of the vehicle being in any of these states is dependent on the nature of the terrain {normal, rough}, the weather conditions {normal, deviant, critical}, the residue of the longitudinal distance (difference of actual distance and the required distance), the state of the actuators and whether the vehicle is undergoing a maneuver. Input to these set of nodes is the result of the sensor validation and fusion modules. There is another set of nodes which represents the state of the actuators of the vehicle {normal, faulty}. The input to these nodes is the result from the fault diagnosis and the lower level influence diagrams. In this rather simplified state of the system, we also have nodes that represent the state of the vehicle in front, and the nature of the system, i.e. whether it is in the platooning mode or is carrying out maneuvering techniques such as lane change, split or merge. Fig. 25 shows a more detailed topology of an influence diagram for the state of a vehicle and Fig. 26 shows an influence diagram for a sensor model for a longitudinal sensor.

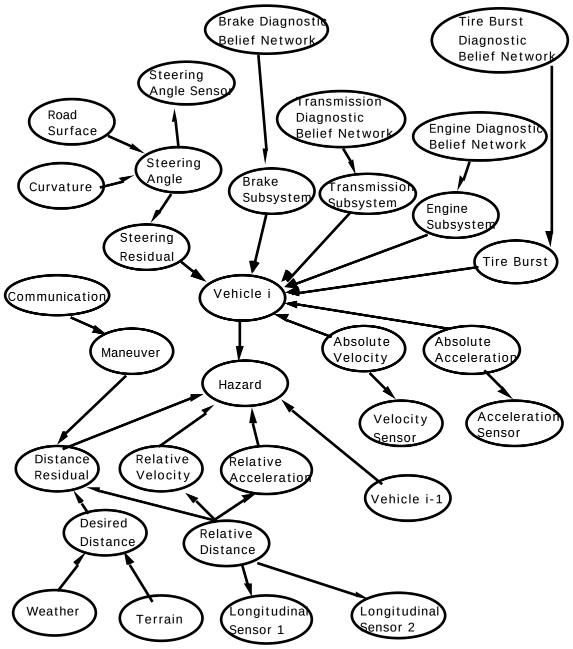


Fig. 25: Belief Network repesenting the state of the vehicle

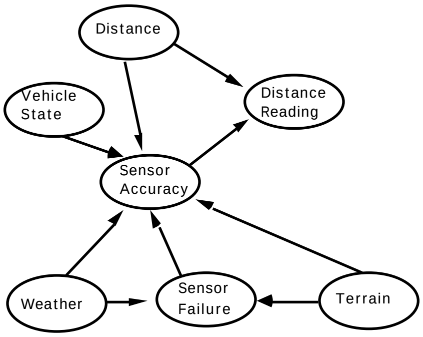


Fig. 26: Sensor Model for a Longitudinal Sensor

4.3 Safety Decision Maker (SDM)

Once the diagnosis of the present state of the system is obtained during an emergency situation or failure mode (probabilities of any of the hazards occurring is greater than a certain threshold), a decision module is triggered which proceeds to make recommendations to the coordination level controller. For this a systematic study needs to be carried out to identify various classes of failure modes and rate them according to some combination of criteria such as severity of the impending accident, and use this in deciding in real time what the optimal corrective actions should be. For instance, if it is sensed during a maneuver that the proposed sequence is unsafe to proceed to completion, we need to decide in real time the course of action which would be the most safe according to certain criteria. This involves enumerating the various feasible states to which the vehicles in the interacting vicinity can proceed to and chooses between them to locally optimize the decision to change the path planning initiated by the platoon in the first place.

The state of the IVHS system on the platoon level can be modeled as an optimization problem, with possible states of each of the vehicles defining the feasible stochastic search space. The state of individual vehicles in a platoon are the design variable, to be used in the optimization process. The objective function in this case would be multi-attribute consisting of a list of possible hazards and the probability of each occurring obtained from the Intelligent Decision Module.

The approach of influence diagrams can also be used for the decision process. A typical influence diagram for this task would be similar to the one shown in fig. 27. Here, the cost of a potential action is dependent on the possibility of various hazards occurring and the decision that is made. The decision made is dependent on the possibility of various hazards occurring, the state of the vehicle and the one before it, results from the fault diagnosis and whether the vehicle is carrying out any maneuver.

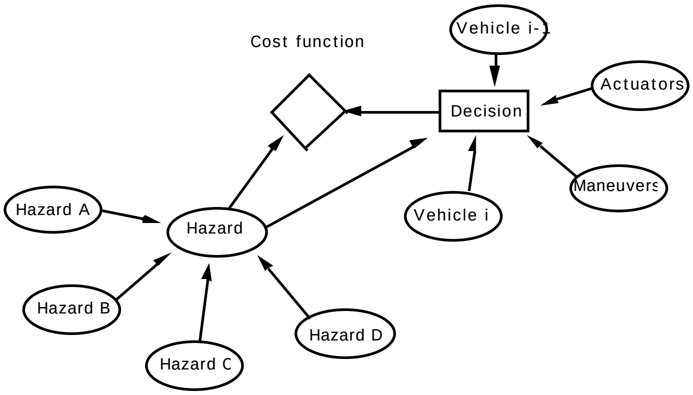


Fig. 27: Influence Diagram for Decision Making

The SDM will be a knowledge based system which can exhaustively deal with various scenarios. It will be trained off-line to avoid the often lengthy optimization algorithms because the IVHS system has to react in the shortest possible time under the paradigm of safety. The real time implementation will therefore use look up schemes and pattern recognition techniques to find the safest action. Fig. 28 shows the operation principle for the SDM.

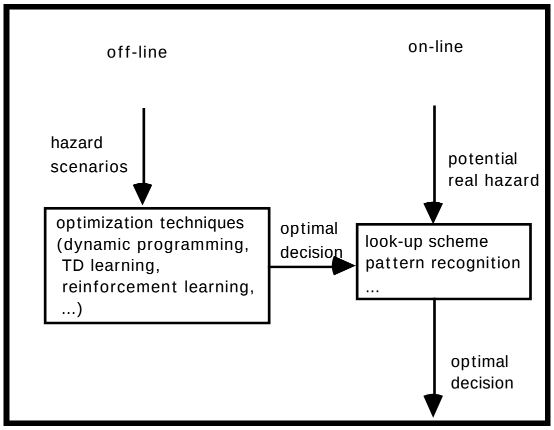


Fig. 28: Operation Principle for Safety Decision Maker

An example of the output of the system could be (in case of emergency), hazard A likely to occur in the next *t* seconds, result of the fault diagnosis module showed a tire burst. Recommended action: carry out sequence #128, i.e. the precompiled procedure to move the vehicle out of the automated lane, in case of a tire burst.

5.0 Results from Experiments

Experiments were carried out to characterize the various longitudinal sensors, namely radar sensor, sonar sensor, and optical sensor with special emphasis on the optical triangulation sensor. A test stand was developed to measure the distance while assuring proper alignment of the car with the source for the sensor. A camera tripod served as the base on which a 3m long horizontal aluminum bar was mounted. This was used to move the source on a prescribed path. Angular alignment was accomplished by taking advantage of the degrees of freedom of the head of the tripod, adjustment was done according to an angle measurement device thus allowing for compensation for irregularities in the ground topography. Fig. 29 shows the schematic of the test stand in a top view.

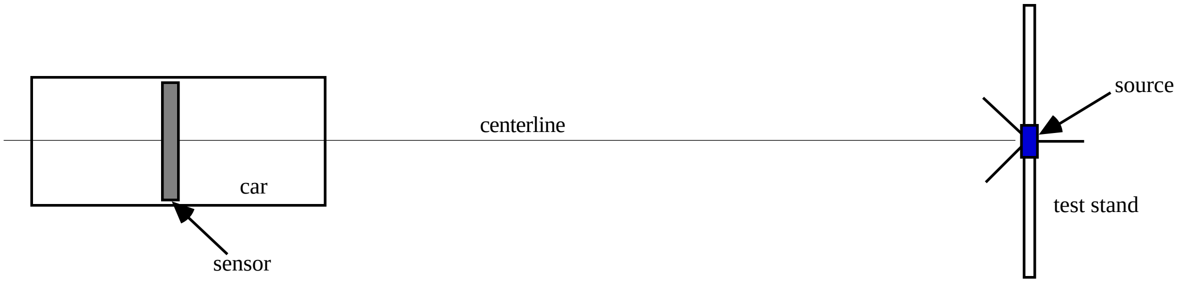


Fig. 29: Schematic of the test setup (top view).

Test data were taken for the static and dynamic case. For the static case the source was fixed at distances 1m apart. This was carried out from one end of the sensor range to the other. Data were sampled at 50 Hz for a minute to get about 3000 samples at each test point. Lateral testing was also done by moving the source perpendicular to the centerline of the car. This way we tried to determine the foot print of the sensors.

For the dynamic case, the source was fixed on one car and the car containing the camera was moved back and forth behind the stationery lead car. Several conditions were tested for, e.g. driving straight behind the lead car, driving at an offset left and right, approaching the car with varying speed, driving over uneven surface while approaching the lead car, approaching the car at an angle. Fig. 30 shows the sensors mounted on the vehicle. The various conditions are schematically displayed to illustrate the case under consideration. For all cases we sampled the optical sensor as well as the radar sensor and the sonar sensor.

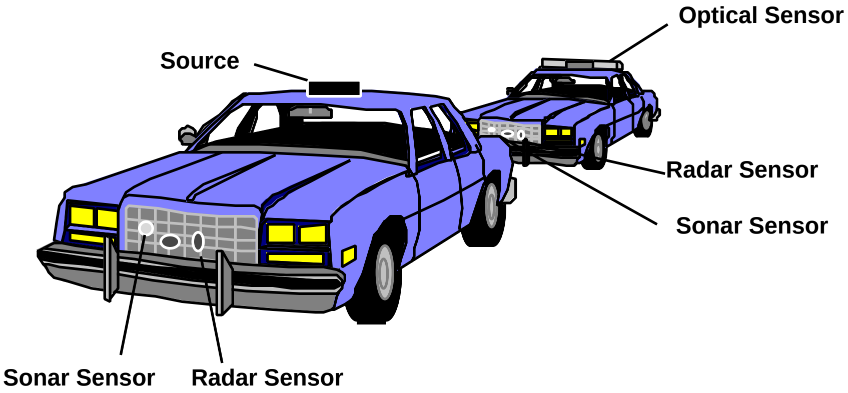


Fig. 30: Platooning Test Set Up

5.1 Static Testing

The three sensors were first tested under static conditions, i.e. the vehicles were not moving. Rather, the data were sampled at distances ranging from 1m bumper to bumper to 30m (fig. 31). The linearity characteristics as well as the accuracy vs. range are subsequentely listed. For the optical triangulation sensor, the dropout rate vs. range is also listed.

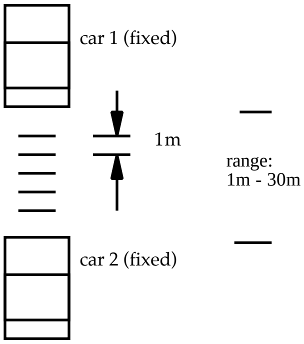


Fig. 31: Schematic of test set up

5.1.1 Optical sensor

Optical Triangulation Technology

- Principle of operation: optical triangulation
  - infrared light source
  - two position detectors

– active sensor, i.e. detectors and target (source) are powered

- Output: longitudinal as well as lateral distance
- Observation:
  - sensitivity to light intensity
  - aberrant readings at close range (<2m) due to saturation and at larger distance where light detected is too week; we recommend that adaptive gain control (AGC) be utilized to solve this problem
  - sensitivity to tilt, i.e. angle between source and sensor
  - sensitivity to fog, dust, smoke, haze, rain, etc. which affect light intensity
  - Robust due to the use of a prescribed target, i.e. the IR source

Figs. 32 through 34 show the spread, the measurement error, and dropout rate of the optical triangulation sensor.

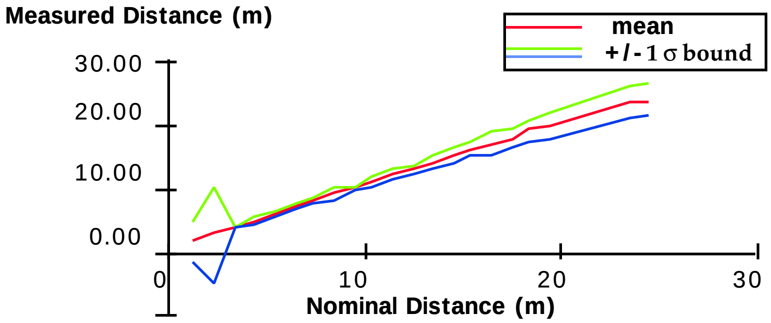


Fig. 32: Spread (1s bound) of optical triangulation sensor

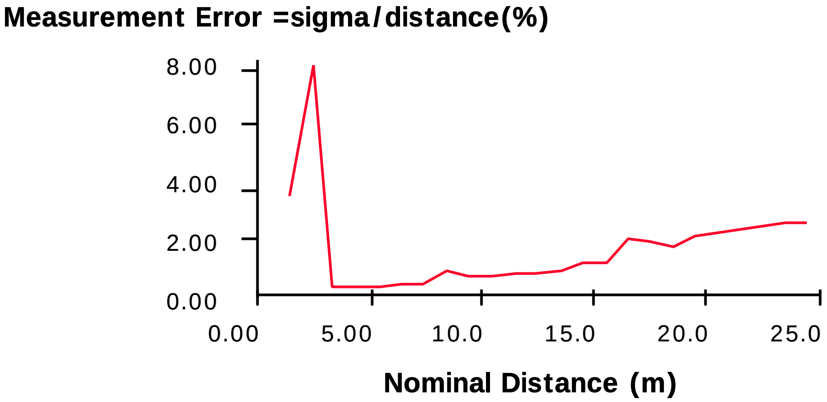


Fig. 33: Acuracy vs. Range of Optical Triangulation Sensor

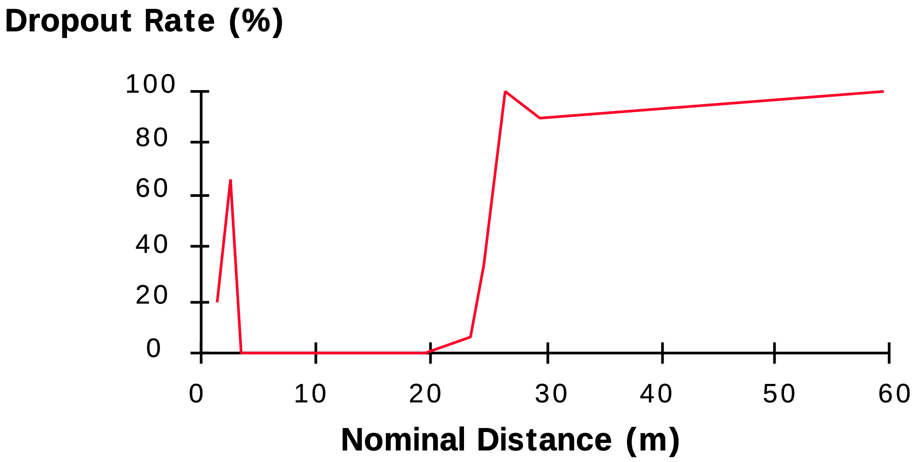


Fig. 34: Dropout Rate vs Range of Optical Triangulation Sensor

5.1.2 Radar Sensor

Radar Sensing Technology

- Radar transmits and receives energy with millimeter wavelength
- Frequency: 24 GHz
- Observation:
  - can receiving signal from unknown target. e.g. signal can bounce under preceding car or go through the rear window
  - almost always returns some value as opposed to default value even when the actual distance is out of its operating range
  - sensitivity to offset and tilt

Figs. 35 and 36 show the spread and the measurement error respectively, of the radar sensor.

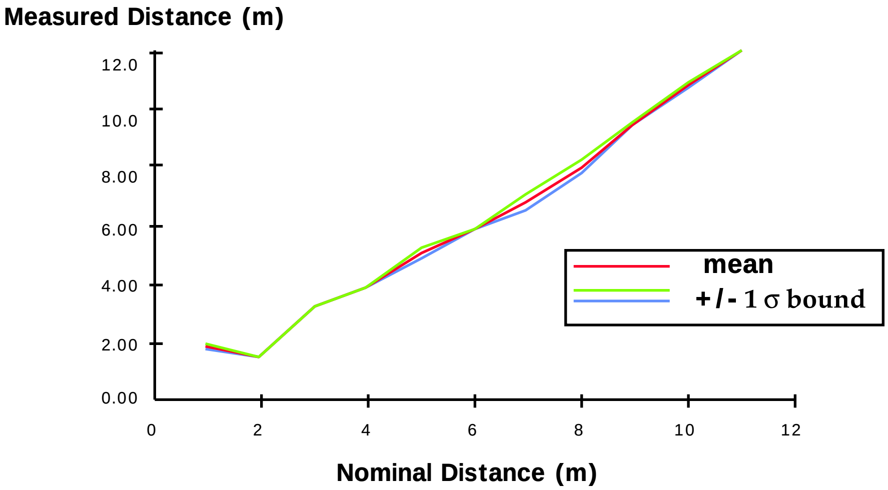


Fig. 35: Spread (1s bound) of Radar Sensor

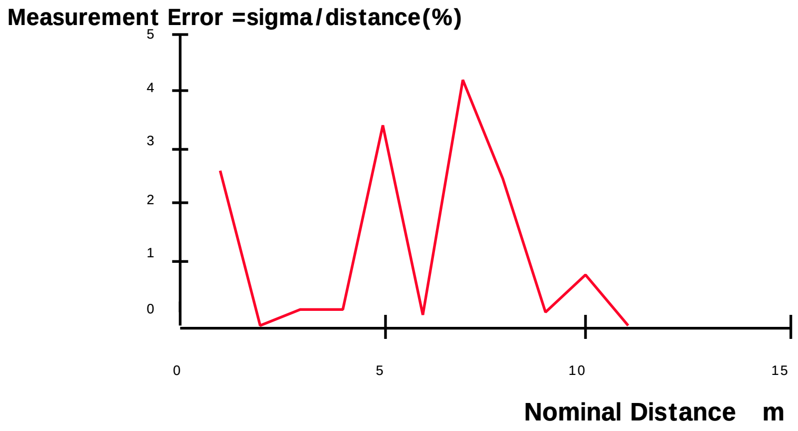


Fig. 36: Accuracy vs. Range of Radar Sensor

5.1.3 Sonar Sensing Technology

- Distance calculated by multiplying the time of flight between transmitted pulse and a returned echo and sound speed.
- Sonar sensors have a wide beam width and are sensitive to specular surfaces.
- Due to large wavelength (about 1/8 in) many surfaces appear smooth, which leads to large specular reflections.
- Errors can be caused by atmospheric effects, such as change in speed of sound due to variations of temperature and humidity
- Observation:
  - Sonar signal attenuates very quickly. Therefore useful measurements are limited in range to about 5m.

Figs. 37 and 38 show the spread and measurement error, respectively of the sonar sensor.

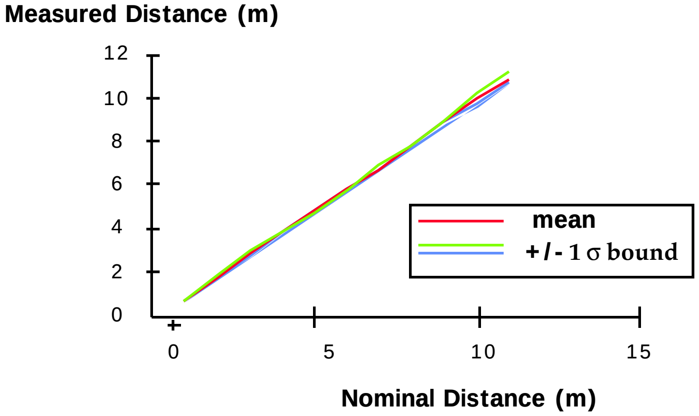


Fig. 37: Spread (1 s bound) of Sonar Sensor

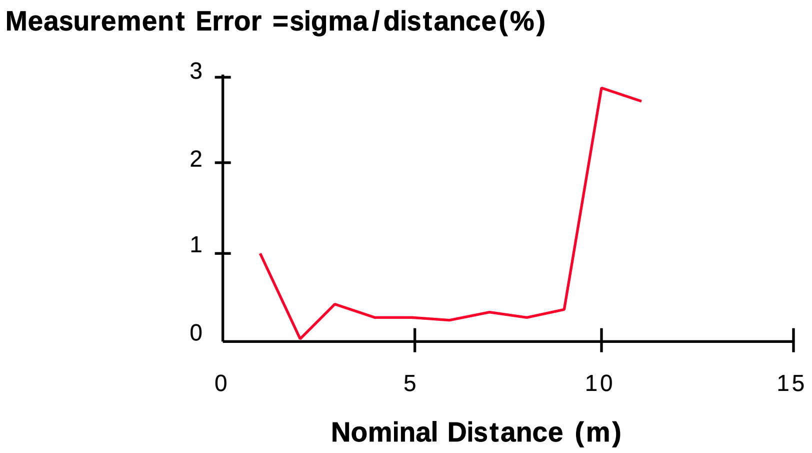


Fig. 38: Accuracy vs. Range of Sonar Sensor

Extensive results of static testing for the optical triangulation sensor can be found in the appendix. They cover the distribution of the sensor at distances ranging from 6 meters to 65 meters.

5.2 Results of Dynamic Tests

The different cases considered are:

- driving straight, varying range
- driving at an offset to left (1m) and right (1m)
- driving through a dip, slow (5km/h) and fast (50km/h), and on uneven pavement (30km/h)
- driving straight behind the lead car while inducing vibration through breaking
- driving at an angle (20°)

Fig. 39 shows the schematic of the setup and the convention for the terms used.
For the first few cases, the distance was also calculated from the velocity sensor and displayed as a "virtual" sensor. The radar sensor displays an offset in some cases and not in others which is the result of ongoing calibration on the sensor. The same is true for the overflow problem of the optical sensor which we detected. In later experimants, this problem was fixed.

All sensors have their characteristic signature which makes them useful in a particular situation. No sensor shows superior performance throughout the whole range and operating conditions

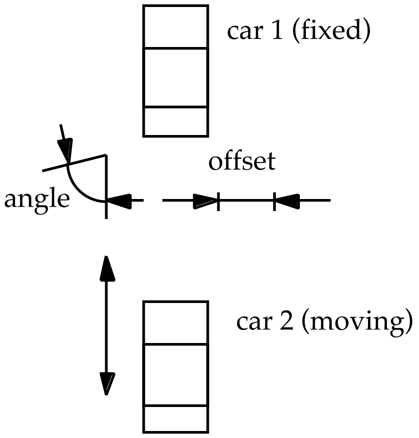


Fig. 39: Setup: Dynamic Testing

5.2.1 Driving straight behind the lead car

Fig. 40 - 42 show different ranges for driving behind the fixed lead car on even pavement. In comparison to the other sensors, the optical sensor has more spread of the readings at all times. This is true although the readings are averaged over 5 samples within the sensor. The earlier observation of higher variance at the turnaround which was suspected due to data acquisition error also occurs at stand still. It seems that the data spread increases as the speed decreases. The spread is also higher when the sensor reaches the limits of its range. Most obvious is the overflow problem at close range which had not been fixed at that time. It appears as if the data are continued with a negative sign after some out of bounds readings. We also observed a saturation effect when the source was held at close range for more than three seconds. In that case, the sensor would not give proper readings when moved away. The system had to be shut down and restarted to allow recovery from that effect. If held only a short period of time at close range, this saturation effect does not occur. The limit distance when this problem occurred was about 7.5m from source to camera - or 2m from bumper to bumper.

The radar sensor has smoother values over its operating range. This range ends at about 10m, where it assumes a default value of 15m.

Similarly, the sonar sensor has a cut off at 10m. However, there does not exist a default value for the sonar sensor. Rather, it takes on random values between 1.5m and -1.5m.

One ambiguity with the radar and sonar sensors is that the obtained signal could be reflected from the rear end of the trunk, the bumper, or some other part of the car. The Optical sensor has the advantage of a prescribed distance, namely from source to camera.

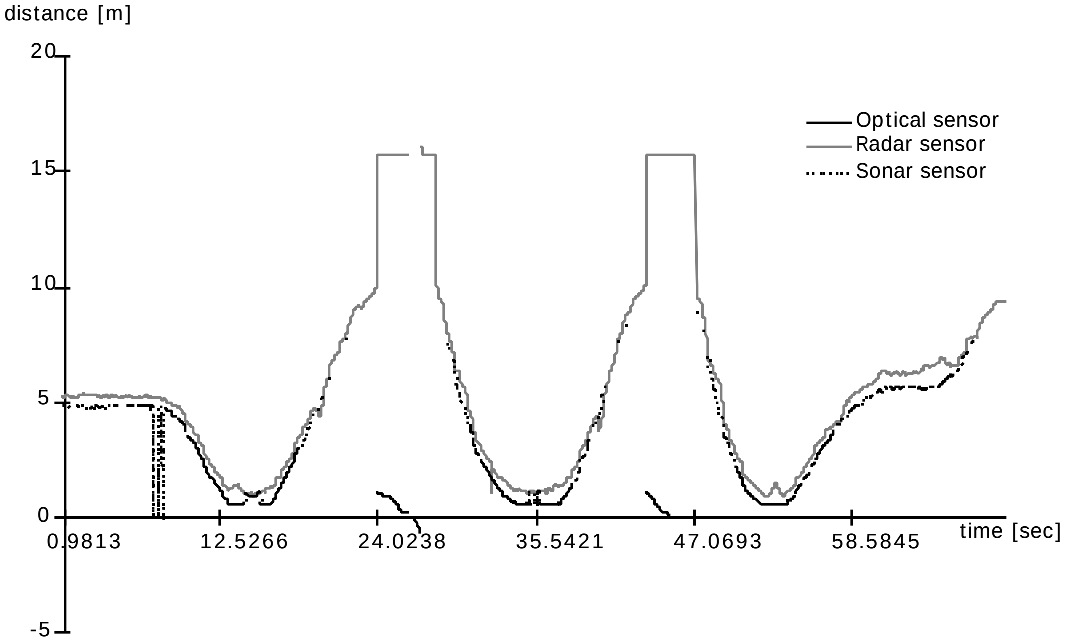


Fig. 40: Dynamic test of all three distance sensors while driving straight behind the lead car

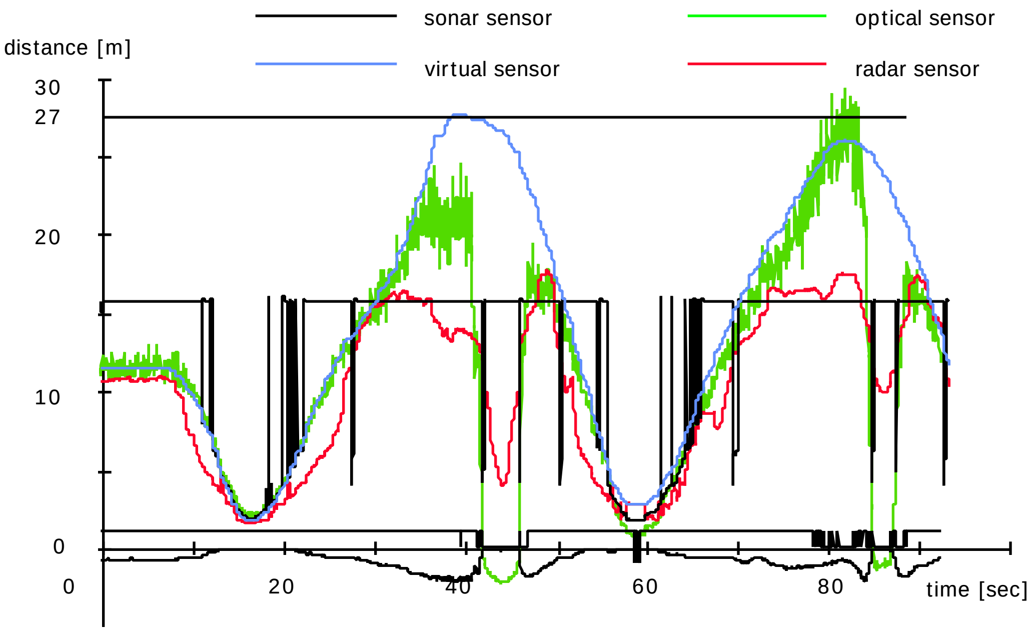


Fig. 41: Straight behind Fixed Target up to about 27m

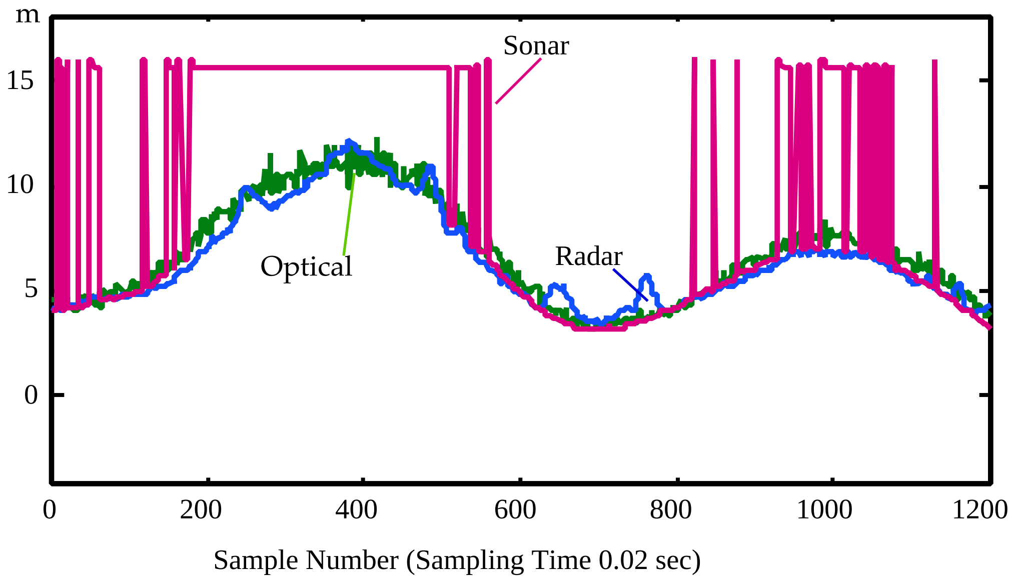


Fig. 42: Approaching a Fixed Target straight

5.2.2 Driving at an offset to the right and left

Fig.s 43-44 show the case for driving at an offset of about 1 meter to the right and to the left relative to the fixed lead car. The optical sensor has “good” readings over the entire range. The sensor confidence drops to zero at close range. However, the readings still seem to make sense. Apart from going into negative range which can be explained with wrong measurement of the length of the car which was subtracted from the distance readings to give bumper-to-bumper readings. It seems that in this case (driving at offset) the cameras get less intensity of light (due to the offset) at very close range - which allows the optical sensor to work as opposed to when the car was straight behind the lead car for the same range (in the latter case it experienced the beforementioned overflow problem). The offset readings are constant (+/- .25m).

The radar sensor gives readings over the entire range as well. However, all sensors seem to give different readings at closer range. The radar sensor renders larger values at close range. This is probably due to the fact that it measures something at the side of the car, for example on of the external rearview mirrors.

The sonar sensor is inaccurate beyond 3m range and has no good readings at all beyond 6m distance.

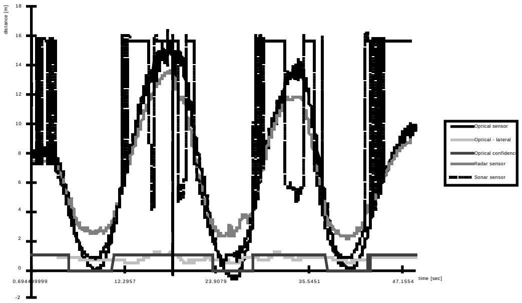


Fig. 43: Dynamic test of all three distance sensors while driving at an offset of 1m to the right and behind the lead car

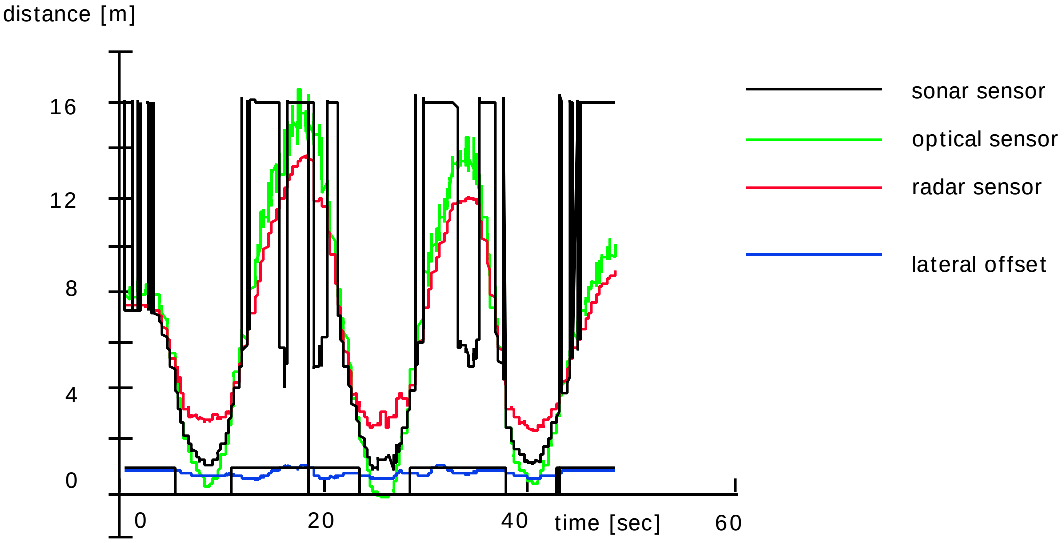


Fig. 44: Offset to Left (~1m)

5.2.3 Going through potholes and on uneven road

Fig.s 45-51 show a series of tests when going through dips, potholes or generally on rough road at varying speeds. When a larger pothole is encountered, all sensors seem to give wrong values: the optical sensor measures something further away while the radar and sonar sensors think the target is closer. For the set of smaller dips and swinging of the suspension after the large dip, the radar sensor behaves properly while the optical sensor experiences fluctuations. That coincides with our observation during static testing which produced evidence for high sensitivity with angular alignment of the source with the cameras. The sonar sensor does not start operation until very close to the target (about 5-6m). The radar sensor measures something that should be out of its range while at stand still. It should produce readings around 15m (its default value) but gives back readings of 24m.

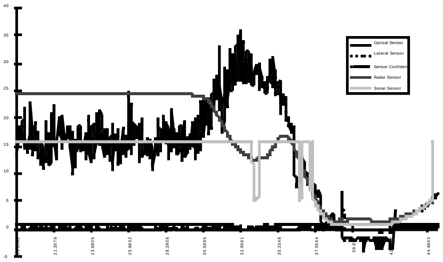


Fig. 45: Dynamic test of all three distance sensors while driving through a series of dips behind the lead car

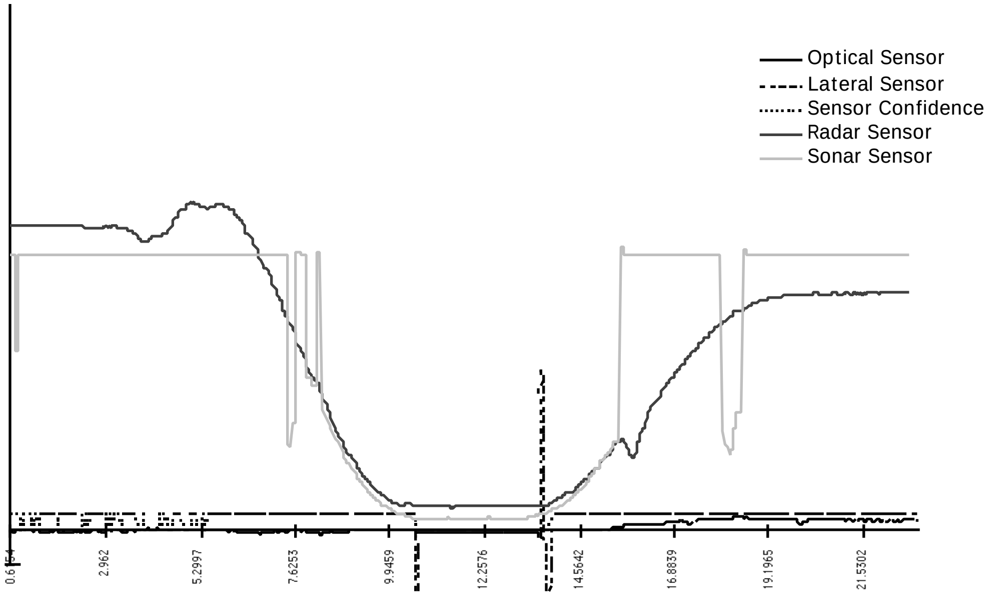


Fig. 46: Dynamic test of all three distance sensors while driving through a series of dips behind the lead car

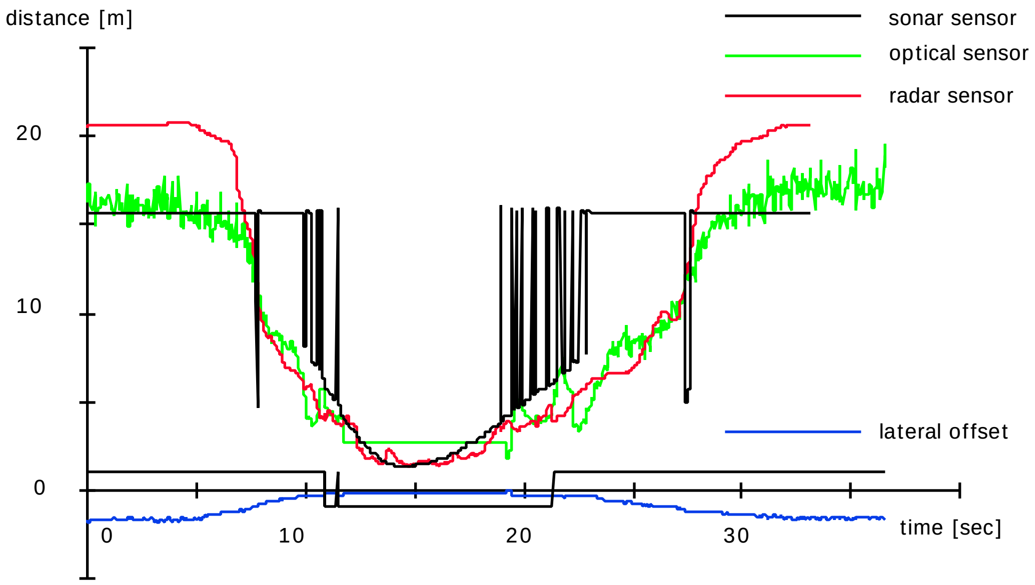


Fig. 47 Dip, slow (5km/ h)

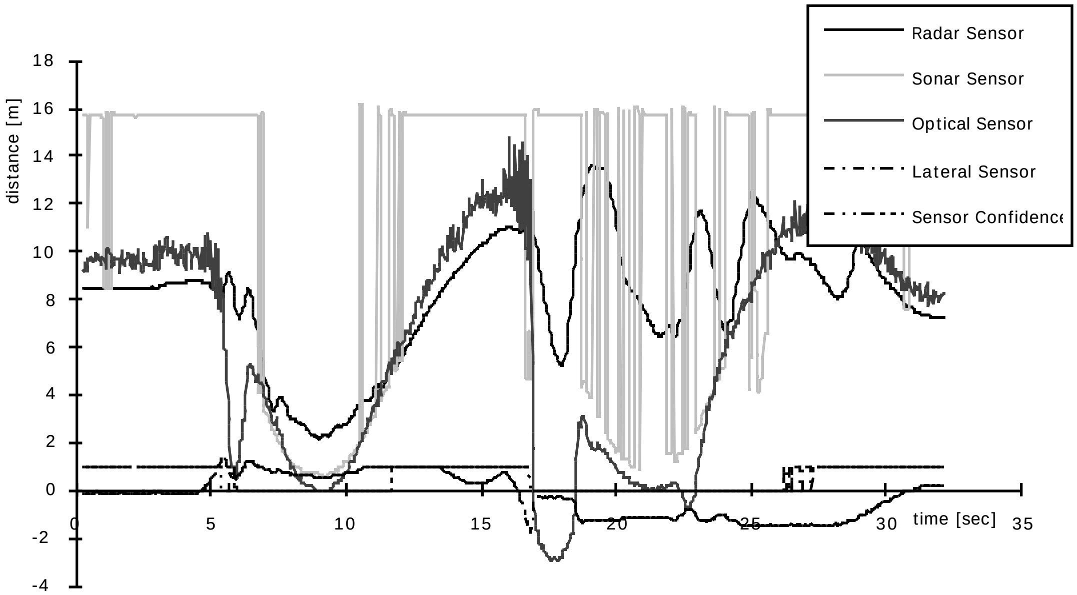


Fig. 48: Potholes

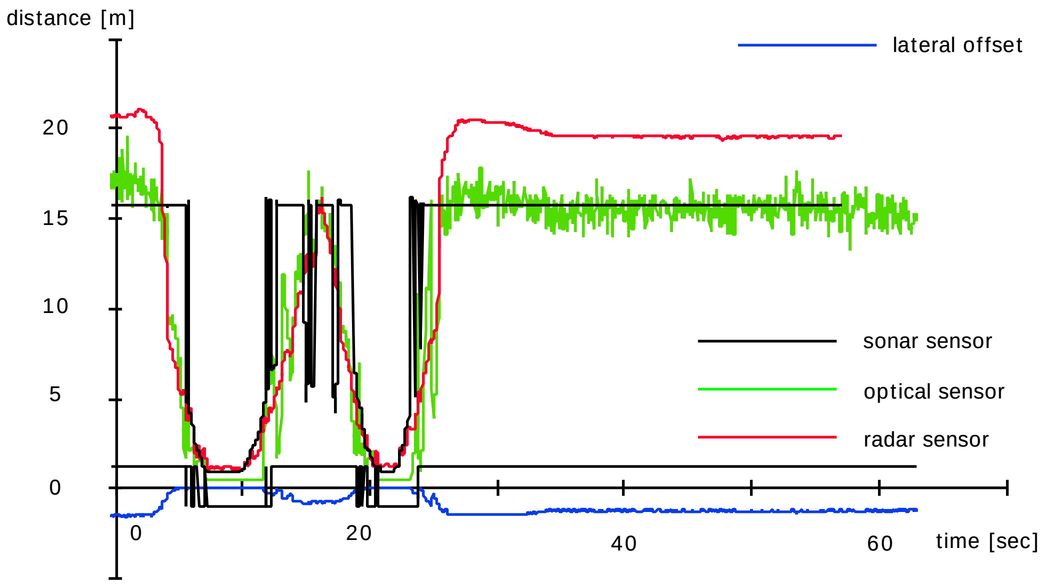


Fig. 49: Dip, fast (50 km/ h)

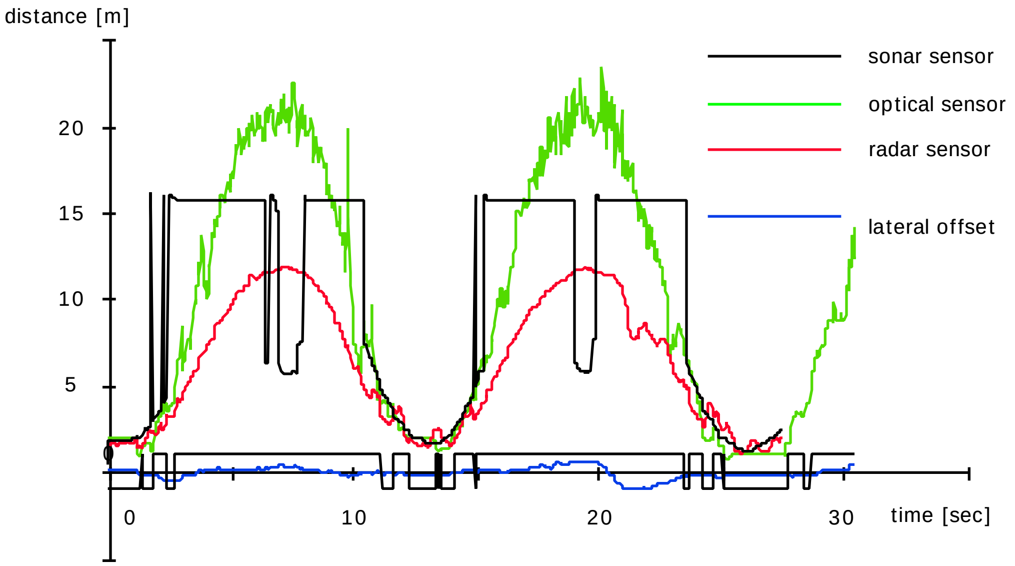


Fig.50: Rough Road (30 km/ h)

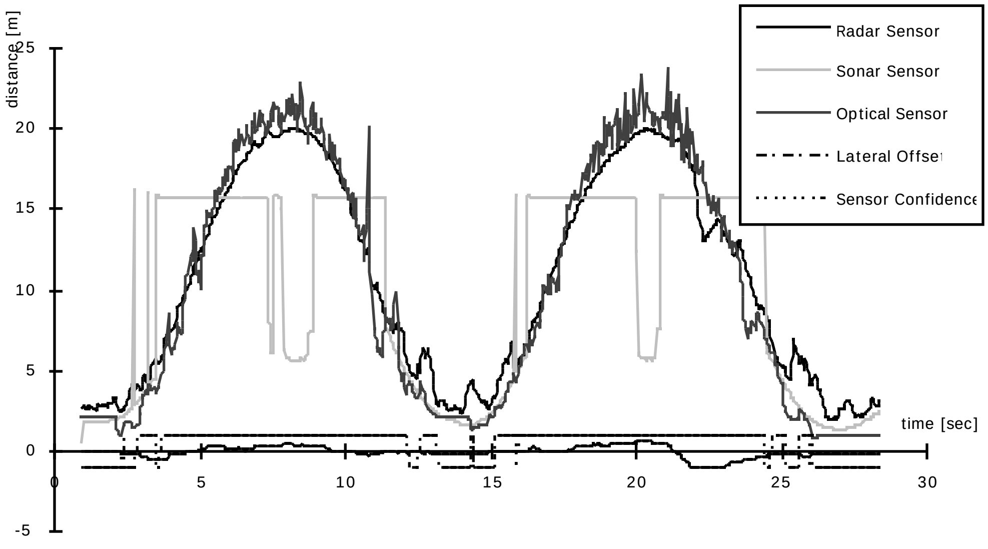


Fig. 51: Uneven Pavement

5.2.4 Inducing vibration through braking

During this experiment, the car was tilted through short braking action while driving in the second cycle behind the fixed lead car. It can be observed that the sensor readings are more fuzzy than when there is no braking (fig. 52).

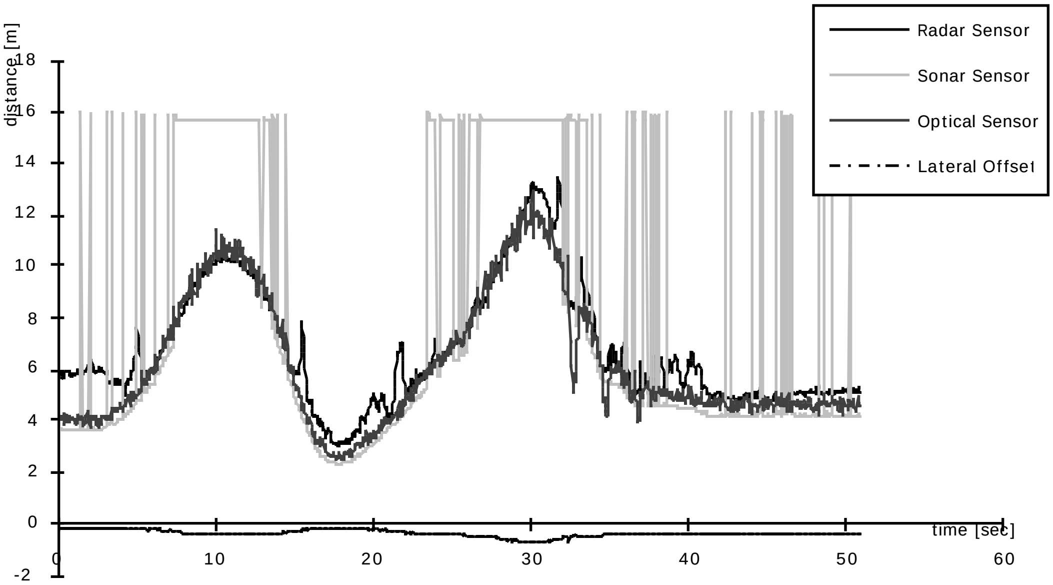


Fig. 52: Induced Vibrations Through Braking

5.2.5 Driving at an angle

Fig. 53 shows the result of driving at an angle to the lead vehicle. As can be seen, the sensors seem to be relatively unaffected by this condition. Radar and sonar sensor find a larger surface toeflect from and the optical sensor does not seem to be sensitive to the rotated source either.

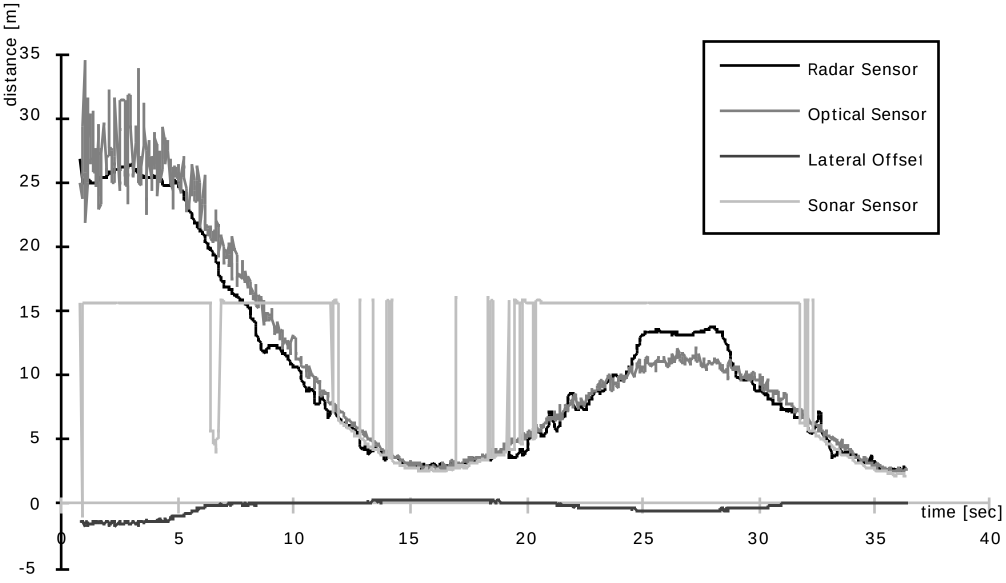


Fig. 53: Driving at an Angle of 20 degrees.

We are currently also looking at testing the sensor under more defined mode of vibration. A test apparatus for the vibration testing has been developed for this purpose. This will test the sensor when it undergoes vertical displacement to simulate potholes. The test parameters have been set to the frequency between 0.5 Hz -25 Hz and the amplitude at 30cm. Fig. 50 shows the schematic of the test stand. Testing will resume when the optical sensor becomes available.

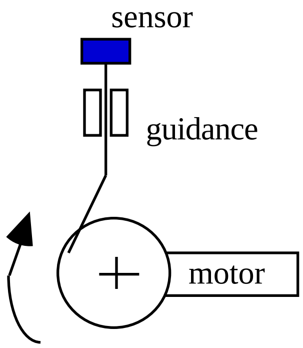


Fig. 54: Schematic of the vibration test stand

5.3 The Lead Vehicle Problem

Within the platoon, there are different jobs to do as far as the control algorithm for the longitudinal (and lateral) distance is concerned. One must differentiate between the lead vehicle and the follower vehicles. During normal operation, the follower vehicles have the task to maintain a constant distance to the preceding vehicle. This affects speed and acceleration control. Sensors, mounted primarily at the front of the vehicle, measure the distance to the next object. The next object is the preceding car unless another object is in between the two vehicles. At traveling velocities of 65 mph and a distance of 1-4 meters that does not seem to be too likely. However, faults can occur through objects which are picked up by the turbulences of the moving cars, as it occured during a recent test. During maneuvers such as lane change, merge, and split, the vehicle in question takes on the role of a lead vehicle. During follower action, several sensors are used to estimate the distance to the next vehicle. Currently, radar, sonar, and optical sensors are employed to perform this task. Some of these sensors can be used for small distances only such as the sonar sensor whose signal attenuates quickly and renders it useless beyond distances of 4 meters. The radar sensor faces the problem that the signal may be reflected by several surfaces such as the bumper, the trunk hood, etc. The optical sensor needs an active light source. Since no sensor can be relied upon alone, sensor validation and fusion needs to be performed to get the proper distance to the next vehicle.

The lead vehicle faces a more difficult problem: it has to find the distance to the next object and then decide whether this object is in its lane or an object outside its lane. Once the sensor detects an object, this object has to be identified to whether it is a platoon traveling ahead of the own platoon or a potentilly hazardous object in the lane such as a stranded vehicle, debris from an accident, an animal, etc. the latter case, maneuvers to avoid the obstacles have to be carried out. This could be decisions such as emergency braking or lane changing. In the former case, the platoon will try to establish and then maintain a safe distance to the platoon in front. Because of the high traveling speeds, long-range sensors need to be employed. However, there is an inherent problem involved with long-range sensing as the potential for misinterpretation is rather large. Objects not actually on the lane could be mistaken for objects on the lane. Objects on the lane could be overlooked because there is a stronger signal from another source. Roadside objects such as a street sign or house or railings or a car in an adjacent lane all could mislead the lead vehicle. The use of an intelligent sensor which can give qualitative information such as a vision sensor or an infrared sensor must be considered. A few graphs shall illustrate the problem outlined above.

- Fig. 55:
This shows how a signal can be reflected by an adjacent platoon. This platoon might be very close and it might travel at a similar velocity as the lead vehicle itself. Should relative velocity sensors be used, it might wrongly identify the adjacent platoon as an object in its lane.

- Fig. 56:
This Fig. shows an ambiguous situation which might occur when a preceding vehicle or platoon performs a lane change operation. The control algorithm will have difficulty to sort out when the object is in its lane and when not. Depending on the bandwidth of the signal, the sensor will or will not be able to “see” when the preceding car is in its lane or not. This is in particular true for a beam with narrow bandwidth and close preceding cars. Stereo sensors might help in this situation.

- Fig. 57:
In a curve, the sensor which previously might have recognized a preceding vehicle/platoon looses the object in a curve. Instead, the signal gets reflected by railings or other roadside elements. Information from roadside communication might help to recognize such a situation.

- Fig. 58:
Some sensors develop a pattern which can be described by lobes: The signal does not go out straight but rather in lobes that cover a certain region to the side. This further complicates identification of objects in longitudinal direction.

- Fig. 59:
If several objects are at the same distance, it will be difficult to identify the relevant one. Furthermore, a tracking system might follow the wrong target if the objects finally move apart. This situation will be further complicated when it occurs in curves.

- Fig. 60:
The signal might become corrupted by vertical influences such as dips, slopes, bridges, fog/rain/snow, vibrations, etc.

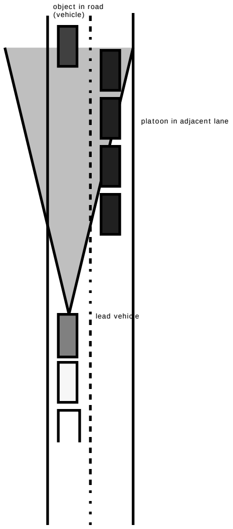


Fig.55: Sensor signal reflected by platoon in adjacent lane

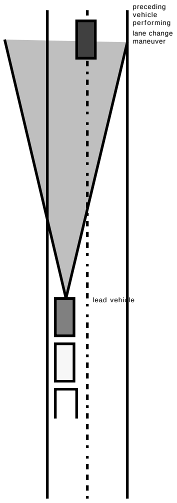


Fig.56: Sensor signal ambiguous

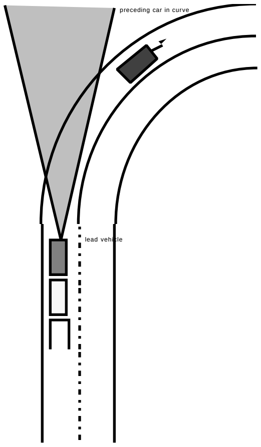


Fig. 57: Sensor looses object in curve and gets more reflections from roadside

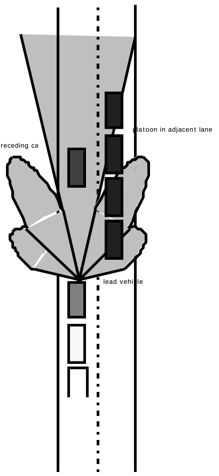


Fig.58: Lobes of sensor may result in ambiguous readings

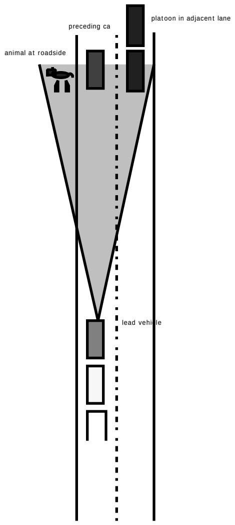


Fig.59: Ambiguity through several objects at same distance, two traveling at the same distance, one static

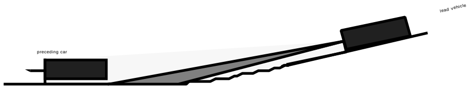


Fig.60: Dips, slopes, fog/rain/snow obstruct the sensor of the lead car

6.0 Summary

The research described in this project is concerned with sensor validation and sensor fusion as a part of a five module hierarchical architecture for supervisory control. The goal is to enhance reliability and safety of the IVHS system by taking into consideration the uncertainty of sensors and the system. Potential hazards are detected and feasible maneuvers are recommended within the upper modules of the architecture. This research focussed on the first two modules, namely sensor validation and sensor fusion and showed how the PDAF can solve the problem of validation and fusion.

7.0 Conclusions

This report describes a framework for validation of sensor data for the platoon-model IVHS system. We conclude that the role of sensor validation should not be limited to direct machine-level controller input. Within our framework, validated sensor readings are not only available for the machine level controllers but also serve as input to for a diagnostic module, a hazard analysis module, and a safety decision maker module. These integrated modules act as an intermediate supervisory controller which output at its top level recommendations in case of emergencies. Subsequent PATH projects address the last three modules of this hierarchical approach.

Acknowledgements

Funding for this project was made available by PATH grant MOU-132. We gratefully acknowledge the help of PATH engineers Pete Devlin, Seibum Choi, and Leon Chen.

References

Agogino, A. M. and Rege, A., "IDES: Influence Diagram Based Expert System", in *Mathematical Modeling in Science and Technology,* Proceedings of the Fifth ICMM, July 29-31, 1985, University of California, Berkeley (eds. X.J.R. Avula, G. Leitmann, C.D. Mote, Jr. and E.Y. Rodin), Pergamon Press, New York, 1986. Also published in Mathematical Modeling, Vol. 8, pp. 227-233, 1986.

Agogino, A.M., Srinivas, S., and Schneider, K.M., "Multiple Sensor Expert System for Diagnostic Reasoning, Monitoring and Control of Mechanical Systems", *Mechanical Systems and Signal Processing*, Vol. 2, No. 2, pp. 165-185, 1988.

Agogino, AM. and K. Ramamurthi, "Real Time Reasoning about Time Constraints and Model Precision in Complex Distributed Mechanical Systems," *Working Papers from the Symposium on AI and Limited Rationality* (AAAI Spring Symposium Series, Stanford University, March 28-30, 1989), pp. 1-5.

Agogino, A.M., M.L. Tseng and P. Jain, "Integrating Neural Networks with Influence Diagrams for Power Plant Monitoring and Diagnostics," (with M.L. Tseng and P. Jain), in *Neural Network Computing for the Electric Power Industry: Proceedings of the 1992 INNS Workshop* (International Neural Network Society), Lawrence Erlbaum Associates, Publishers, Hillsdale, NJ, 1992, pp. 213-216.

Alag, S., A.M. Agogino, and Hsueh, W., "A Methodology for Intelligent Sensor Validation, Fusion and Sensor Fault Detection for Dynamic Systems: Application to Power Plants - Part II," # 95-0303-3, Berkeley Expert Systems Lab., Univ. of Cal., Berkeley,1995.

Alag, S., K. Goebel, A. Agogino, " A Framework for Intelligent Sensor Validation, Sensor Fusion, and Supervisory Control of Automated Vehicles in IVHS", Proceedings of the ITS America 1995 conference, Washington, D.C. March 15-17, 1995a.

Alag, S., K. Goebel, A. Agogino, " A Methodology for Intelligent Sensor Validation and Fusion used in Tracking and Avoidance of Objects for Automated Vehicles", Proceedings of the ACC 1995 conference, Seattle, WA, June 21-23, 1995b.

Astrom, K. J., Eykhoff, P., "System Identification-A Survey", *Automatica,* vol. 7, pp. 123-162, 1971.

Ayache, N. and Faugeras, O. ,"Building, Registering, and Fusing Noisy Visual Maps", *Int. J. Robot. Res.* vol. 7, no. 6, 1988.

Bar-Shalom, Y. and T. E. Fortmann, *Tracking and Data Association*. Boston, MA: Academic, 1988.

Bar-Shalom, Y., editor. *Multitarget-Multisensor Tracking: Advanced Applications.* Artech House, Norwood, MA, 1990.

Bar-Shalom Y. and X. Li, *Estimation and Tracking: Principles, Techniques, and Software*. Boston, MA: Artech House, 1993.

Blackman, S. S., "Association and Fusion of Multiple Sensor Data". in Bar-Shalom, Y. (ed.): *Multi-Target Multi-Sensor Tracking*. Norwood, MA: Artech House, 1990.

Bradley, S.R. and A.M. Agogino, "Optimal Design as a Real Time AI Problem," *System Modeling and Optimization*, (ed., P. Kall; Lecture Notes in Control and Information Sciences 180), Springer-Verlag, 1990, pp. 629-638.

Duyar, A., Eldem, V. , Merrill, W. and Guo, T., "Fault Detection and Diagnosis in Propulsion Systems: A Fault Parameter Estimation Approach", *Journal of Guidance, Control, and Dynamics*, vol. 17, no. 1, 1994.

Duyar, A. and Merrill, W., “Fault Diagnosis for the Space Shuttle Main Engine”, *Journal of Guidance, Control, and Dynamics*, vol. 15, no. 2, 1992.

Fortmann,T.E., Y. Bar-Shalom, and M. Scheffe, “Sonar Tracking of Multiple Targets Using Joint Probabilistic Data Association”, *IEEE Journal of Oceanic Engineering,* vol. OE-8, July 1983, pp. 173-184.

Fleming, P.V., Kara-Baiti, C,, Keller, A.Z. “Application of Fuzzy Reasoning to Failure Mode and Effect Analysis”, Reliability and Safety of Processes and Manufacturing Systems, Proceedings of the 12th Annual Symposium of the Society of Reliability Engineers, Scandinavian Chapter, Tampere, Finlaand, Ed.: Malessen, Y., Rouhiainen, V. Barking, 1991.

Garg, V. and Hedrick, J.K., “Fault Detecion and Control in Automated Highway Systems,” *ASME WAM*, December, 1993.

Grewal, M. S. and Andrews, A. P., *Kalman Filtering: Theory and Practice*, Prentice Hall Information and System Sciences Series, New Jersey, 1993.

Hedrick, J.K. and Garg, V., “Failure Detection and Fault Tolerant Controller Design for Vehicle Control,” PATH Research Report draft no. 93-09, 1993.

Hitchcock, A., “Fault Tree Analysis of a First Example Automated Freeway,” PATH Research Report UCB-ITS-PRR-91-13, 1991.

Hitchcock, A., “Intelligent Vehicle-Highway System Safety:Problems of Requirement, Specifications and Hazard Analysis”, Highway Systems, Human Performance, and Safety, No. 1318, 1991b.

Hitchcock, A., “Methods of Analysis or IVHS Safety,” PATH Research Report UCB-ITS-PRR-92-14, 1992a.

Hitchcock, A., “Intelligent Vehicle/Highway System Safety: Specification and Hazard Analysis,” Transportation Research Board No.930435, 1993.

Hashemipour, H. R., Roy S., and Laub, A. J., “Decentralized structures for parallel Kalman filtering”, *IEEE Trans. AC-33 (1),* pp. 88-93, 1988.

Hsu, A., Eskafi, F., Sachs, S., and Varaiya, P., The Design of Platoon Maneuver Protocols for IVHS”, PATH Research Report, UCB-ITS-PRR-91-6, Berkeley, 1991.

Jazwinski, A. H., “Adaptive Filtering”, *Automatica*, vol 5, pp. 475-485.

Kim, Y-j, Y-j, Wood, W.H., and Agogino, A.M., “Signal Validation for Expert System Development”, *Proceedings of the 2nd International Forum on Expert Systems and Computer Simulations in Energy Engineering*, March 17-20, pp. 9-5-1 to 9-5-6, (Erlangen, Germany), 1992.

Kruse, P. and Clark, D., *Representing Uncertain Knowledge: An Artificial Approach*, Kluwer Academic Publishers, Boston, 1993.

Mehra, R. K., “On the Identification of Variances and Adaptive Kalman Filtering”, *IEEE Trans. Auto. Contr.*, AC-15, 1970, pp. 175-184.

Miller, A.C., M.M. Merkhofer, R.A. Howard, J.E. Matheson, "Development of Automated Aids for Decision Analysis", prepared for the Defense Advanced Research Projects Agency (DARPA), Menlo Park, California: SRI International, 1976.

Narendran, V. K., “Transition Maneuvers in Intelligent Vehicle Highway Systems”, doctoral dissertation *University of California, Berkeley*, 1994.

Olmsted, S. M., On Representing and Solving Decision Problems, Ph.D. Dissertation, Engineering-Economic Systems Department, Stanford University, 1984.

Paasch, R. K. and A.M. Agogino, “A Structural and Behavioral Reasoning System for Diagnosing Large-Scale Systems,” *IEEE Expert*, Vol. 8, No. 4, Aug. 1993, pp. 31-36.

Patwardhan, S., Tomizuka, M., and Kamei, E., “Robust Failure Detection in Lateral Control for IVHS,” *ACC*, p. 1768-1772, 1992a.

Patwardhan, S., Tomizuka, M., Zhang, W-B, and Devlin, P., “Theory and Experiments of Tire Blow-Out Effects and Hazard Reduction Control for Automated Vehicle Lateral Control Systems,”, *Proceedings of the International Symposium on Advanced Vehicle Control*, Tsukuba, Japan, Oct. 1994.

Pouliezos A. D. and G. S. Stavrakankis, *Real Time Fault Monitoring of Industrial Processes.* Kulwer Academic Publishers, Dordrecht, 1994.

Rauch, H. E., “Issues in Intelligent Fault Diagnosis and Control Reconfiguration”, Plenary Speech from *1994 IEEE, International Symposium on Intelligent Control,* Chicago, Aug. 25, 1993, written version, 1994.

Ramamurthi, K. and A.M. Agogino, "Real-Time Expert Systems for Fault Tolerant Supervisory Control", *ASME Transactions, Journal of Systems, Dynamics and Control.*, Vol. 115, June 1993, pp. 219-227.

Shachter, R.D, "Automating Probabilistic Inference", Presented at the Nineteenth Actuarial Research Conference -- 1984, "Credibility Theory and Bayesian Approximation Methods, University of California, Berkeley, 1984.

Sorsa, T. and Koivo, H. N., “Application of Artificial Neural Networks in Process Fault Detection”, *Automatica*, vol. 29, no. 4, pp. 843-849, 1993.

Specht D.F, " Probabilistic Neural Networks and the Polynomial Adaline as Complementary Techniques for Classification," *IEEE Transactions on Neural Networks*, v1, pp. 111-121, March1990.

Specht, R.F., "Intelligent Failure-Tolerant Control", IEEE Control Systems Magzine, June, 1991.

Stengel, , R.F. “Intelligent Failure-Tolerant Control”, *IEEE Control Systems Magazine*, June 1994.

Thomopoulos, S. C. A., “Sensor Integration and Data Fusion”, *Journal of Robotic Systems*. vol. 7, no. 3, 1990, pp. 337-372.

Varaiya, P. and Kurzhanski, A., eds. “Discrete Event Systems: Models and Applications”, vol.II ASA 103, *Lecture Notes in Control and Information Sciences*, Springer, 1988.

Varaiya, P., “Smart Cars on Smart Roads: Problems of Control”, PATH Technical Memorandum 91-5, UC Berkeley, 1991.

Varaiya, P., and Shladover, S., “Sketch of an IVHS Architecture,“ PATH Technical Report UCB-ITS-PRR-91-03, 1991.

Young, S. K. and Clarke, D. W. “Local Sensor Validation”, *Measurement and Control*, vol. 22, pp. 132, 1989.

Appendix

The results which can be found here (figs. 61-67) cover the static tests of the optical triangulation sensor. Tests were performed at distances ranging from 6meters to 65 meters, measured from the source to the sensor. 6 meters is equivalent to about .5 meters bumper to bumper. It must be noted that the results on display here still exhibit the overflow problem which was corrected by Qualimatrix at a later stage. Therefore, the results at close distance do not make too much sense since they are out of range readings. They are included here for completeness. Some experiments recorded readings which were outliers. They were not removed to give an accurate representation of the performance of the sensor. However, the distribution appears to be distorted, because it seems much narrower in the presence of the outliers. The results have to be read with this information in mind.

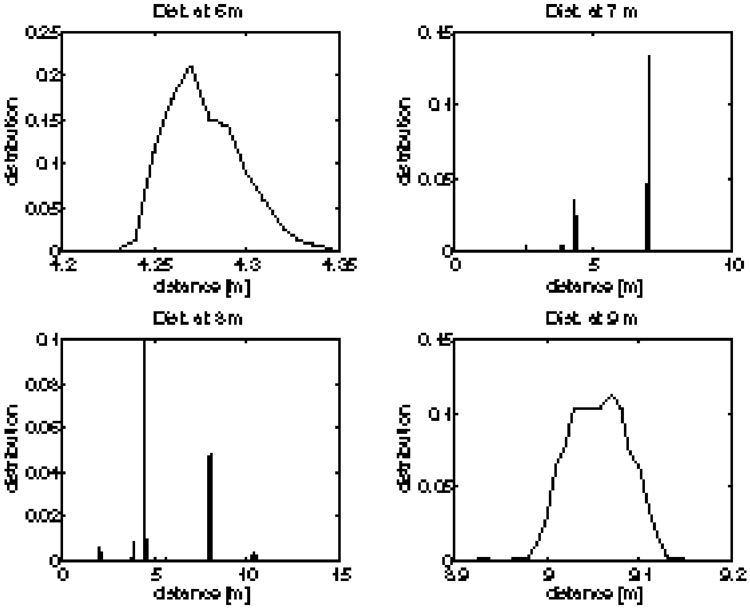


Fig. 61: Sensor output distribution from 6 to 9 meters

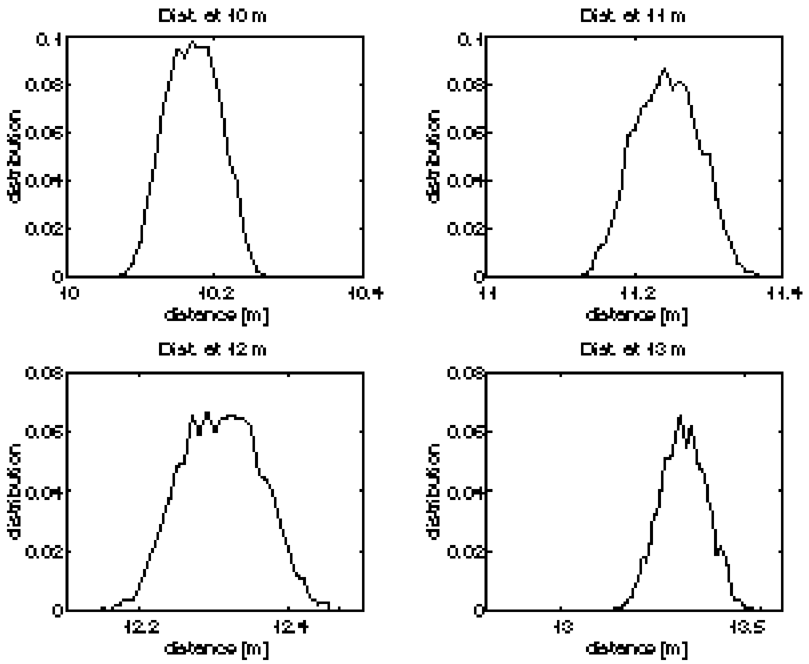


Fig. 62 Sensor output distribution from 10 to 13 meters

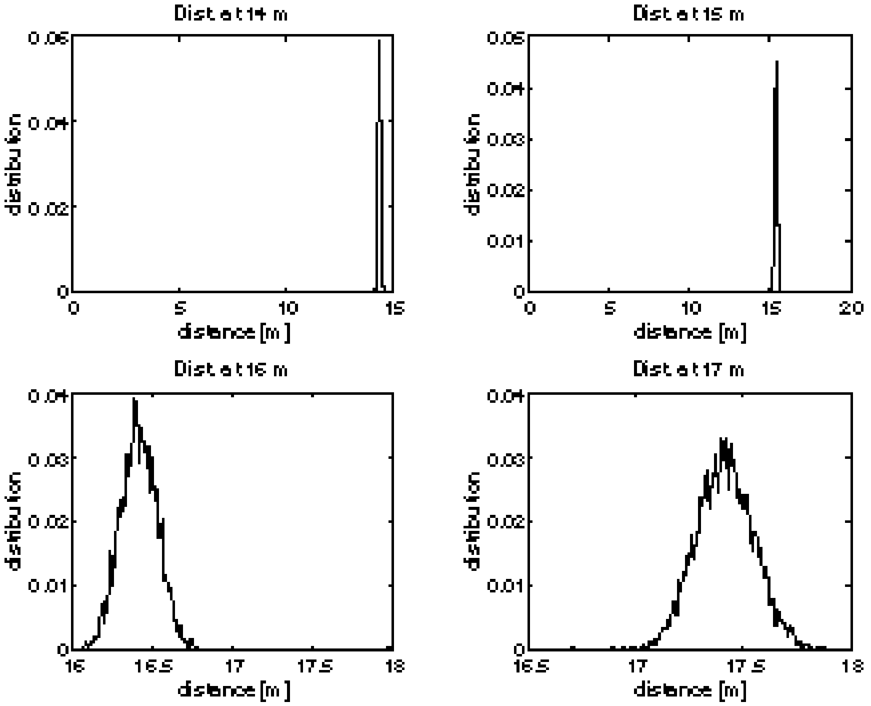


Fig. 63: Sensor output distribution from 14 to 17meters

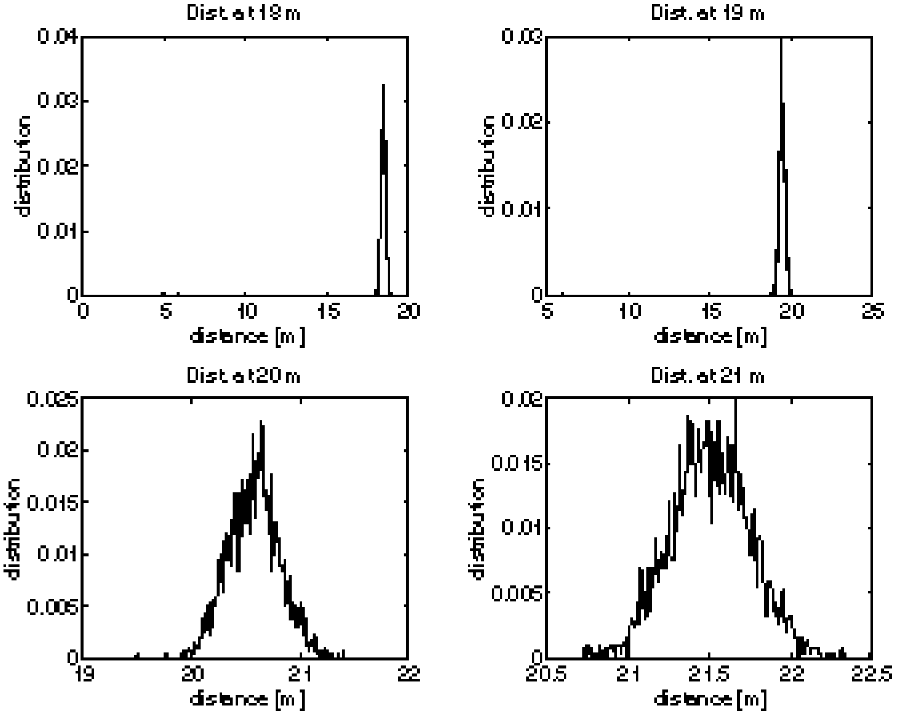


Fig. 64: Sensor output distribution from 18 to 21 meters

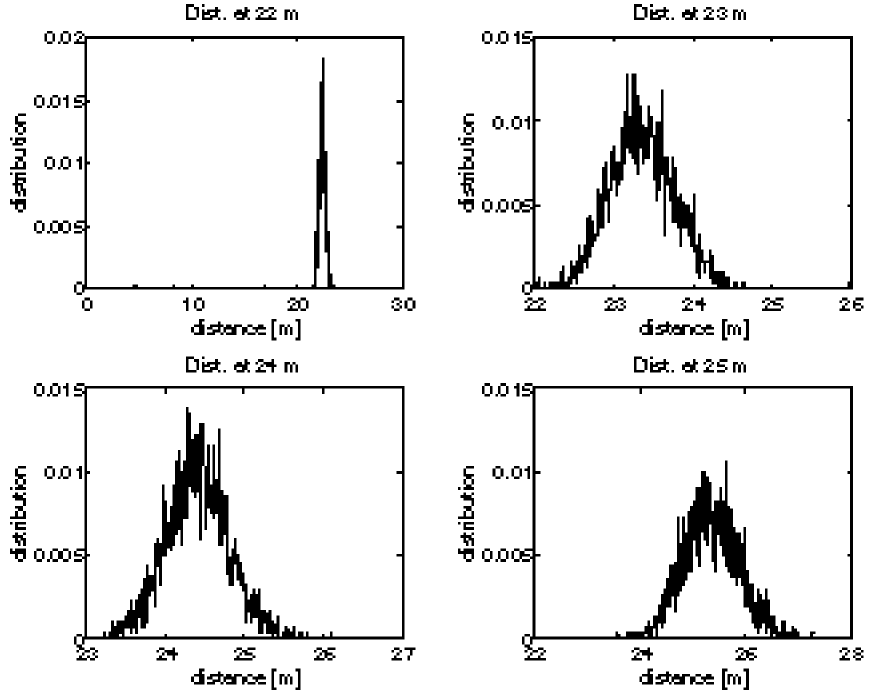


Fig. 65: Sensor output distribution from 22 to 25 meters

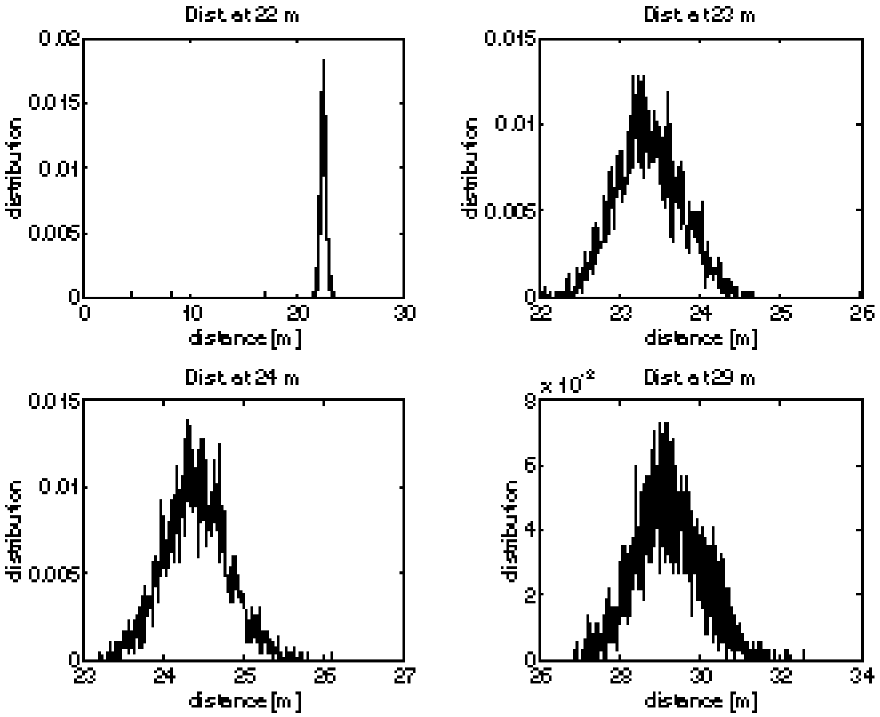


Fig. 66: Sensor output distribution from 22to 29 meters

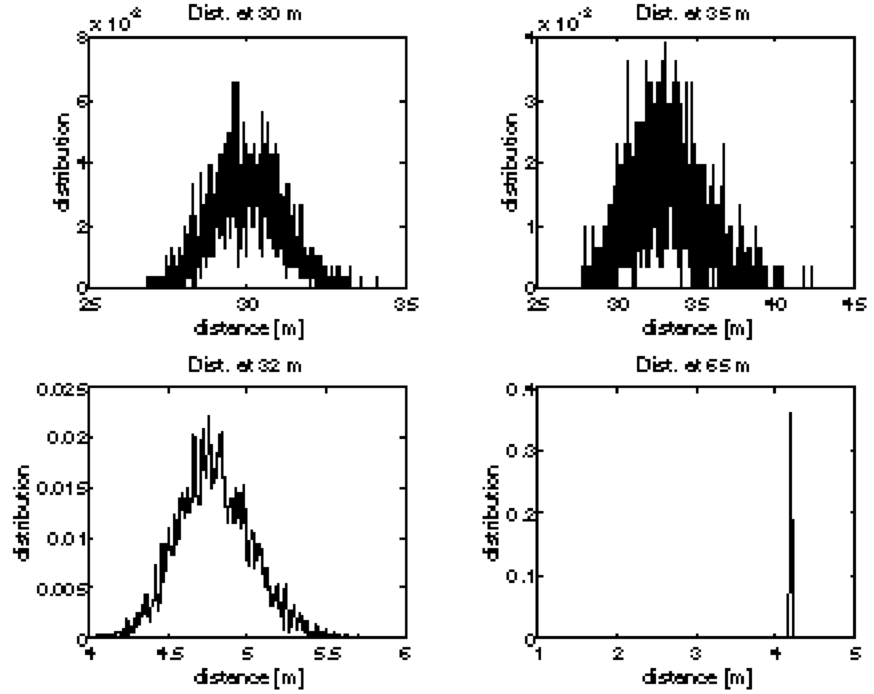


Fig. 67: Sensor output distribution from 30 to 65 meters