



Toward reducing failure risk in an integrated vehicle health maintenance system: A fuzzy multi-sensor data fusion Kalman filter approach for IVHMS

James A. Rodger *

Indiana University of Pennsylvania, MIS and Decision Sciences, Eberly College of Business & Information Technology, Indiana, PA 15705, USA

ARTICLE INFO

Keywords:

Multi-sensor data fusion
Failure risk
Product and process innovation
Fuzzy Kalman filter approach
Fault detection and isolation

ABSTRACT

This paper reports on a new integrated vehicle health maintenance system (IVHMS) based on fault detection and feedback. A fuzzy multi-sensor data fusion Kalman model was used to help reduce IVHMS failure risk. The IVHMS was tested, and sensors with and without faults were identified. The results demonstrate that multi-sensor data fusion based on fault detection and fuzzy Kalman feedback is an effective method of reducing risk in an IVHMS. Use of the fuzzy Kalman filter approach reduced the time needed to perform complex matrix manipulations to control higher order systems in the IVHMS. Moreover, the approach was able to capture the nonlinearity of engine operations under the influence of various anomalies.

© 2012 Elsevier Ltd. All rights reserved.

1. Introduction

This paper investigates a fuzzy multi-sensor data fusion Kalman filter model for reducing risk in an integrated vehicle health maintenance system (IVHMS). Such a maintenance system can be used to determine vehicle health by predicting system and component failures and measuring specific degradations of vehicle components. Effective and efficient platform health monitoring requires that data be readily available. Thus, data were integrated from sensors that monitored and predicted failures in high-payoff areas and captured information about vehicle use. Such information included recommendations for possible vehicle mounting locations and the positioning of sensors. This led to a definition of the data types and data parameters that were available from the working sensors and the data analysis that was necessary to determine diagnostics and planned prognostics on the vehicle systems and components measured. The ultimate goal was to design, build, and test a prototype system to integrate sensor fusion and sensor fusion algorithms to reduce engine failure rates.

Gear, engine, fuel, and electrical sensors were used in an IVHMS that could be viewed on a dashboard. The gear and engine subsystems combined to determine the engine speed in revolutions per minute. The fuel and electrical subsystems combined to determine the fuel economy in liters per hour. Through data fusion, the dashboard and the engine speed and fuel economy subsystems were integrated into an IVHMS. Therefore, if any sensor failed, the system was able to discriminate which subsystem, and which fault sensor, had failed. If two independent sensors in the two subsystems failed, then only the dashboard was available for fault detection, and the

efficiency of the sensors as well as the precision and usability of the IVHMS were greatly decreased.

2. Background information

2.1. IVHMS and vehicle maintenance

IVHMS can be used as an on-board data collection device to feed the multisensor data fusion fuzzy logic algorithms. The device can collect discrete sensor data, CAN J1939 and J1708 data, and transmit to the backend system via wireless communications. The device can sleep in a low power mode and wake from power up or CAN bias detection. The device is able to run on 12 V or 24 V vehicle power. The sensor rack is populated with signal conditioning modules, which convert the raw sensor voltages into 0–5 VDC signals to be read by the datalogger's analog inputs. The sensor rack is powered by the datalogger's internal power source and switched off during sleep mode. Data is sent from the on-board device to the backend servers via wireless communications to a local intermediate File Transfer Protocol (FTP) server. That server stores and forwards all data files from the on-board device via a secure Virtual Private Network (VPN) connection to backend servers. Several servers, each with a specific function post-process the data files and store the data for later review. The FTP server handles incoming data and the web portal interface and the parser server breaks down the data files into raw data for storage on the database server. Data can typically be stored in fault, event, probe, time-in-state, histogram, and pre/post fault buffer formats. Additionally, data can be backed up on a redundant server.

Generally, there are two structures used in multisensor data fusion systems: central fusion and distributed fusion. In a central fusion structure, all the information from the original sensors is

* Tel.: +1 724 357 5944.

E-mail address: jrodriger@iup.edu

sent to the same fusion center to generate a larger observing space. Because all the information is processed at the fusion center, there is a large amount of data transmission in this structure which may require fast processing. In distributed fusion structure, each sensor gives its own decision and the fusion is based on decisions rather than raw information, so the data transmission load is greatly reduced. Sensors in the distributed fusion structure are independent from each other and not required to be of the same type, so, even if some of the sensors fail, the system can still use the remaining sensors to detect the vehicle condition. This feature is important in vehicle health monitoring. The techniques successfully used in object identity fusion range from well known statistical-based algorithms such as classical inference and Bayesian methods to ad hoc methods such as neural networks. In many applications, parametric estimation represents the major fusion algorithms in a multisensor data fusion system. The four most applicable and mature multisensor data fusion algorithms are fuzzy logic, classical inference, Bayesian, and Dempster-Shafer.

For a nonmilitary application, such as monitoring a mechanical system, processing may be aimed at developing alternative predictions about the future conditions and health of a machine, based on anticipated machine utilization and load conditions. This level of processing would seek to estimate time remaining useful life for the machine in question. For applications such as health monitoring for machinery, a level of fusion involves attempting to predict the time of failure, evolution of failure conditions, consequences of operational capability, and uncertainties. Currently, the IVHMS unit can collect formatted time-stamped data from the vehicle data bus, and it can also collect data from other external sensors. Systems such as motor vehicles require monitoring beyond the ability of the human operator. Semi-automated monitoring is required to ensure that the system continues to operate properly. Data from multiple sensors is monitored to assess the health of the vehicle or systems of the vehicle. Observable data include acoustic emissions, vibrations, pressures, temperatures, liquid quality, acceleration, loads, etc.

The system configuration includes multiple sensors, a signal conditioning unit, a data logger, a data transmission method, and a data fusion platform for end user access to the collected data. A dashboard is used to synthesize the data into one visualization tool. Research identified potential signal conditioning modules and sensor-testing hardware products. The data acquisition, signal conditioning, and data communication products used have multiple signal conditioning modules that provide filtering; isolation; amplification of signals; and conversion of signals between sensors, actuators, and host data acquisition systems. An accelerometer input module provides excitation to piezoelectric sensors with built-in microelectronic amplifiers and isolates, filters, and amplifies the sensor output. The excitation current, signal gain, and filter high-pass and low-pass cutoff frequencies are configurable through a set of slide switchers. This integrated schematic is then used to identify faults in the system sensors and determine the resultant reduction in risk due to the application of a FKFA.

2.2. Fuzzy logic Kalman filters and review of multi-sensor data fusion applications

Fuzzy multi-sensor data fusion using a Kalman filter has received very little attention in the literature. Yet the greater the need to decrease failure risks, the more important this approach becomes. Models that assume fixed parameters likely ignore some acceptable solutions. For example, mechanics specifying a maximum engine “redline” speed of 9000 rpm may consider 9010 rpm acceptable for short periods of time. A typical filter set to ≤ 9000 rpm would ignore this option. In reality, an acceptable level of revolutions per minute may not adhere strictly to a certain

value (e.g., 9000 rpm). However, large deviations from acceptable revolutions per minute parameters will result in engine loss.

The remainder of this article is organized as follows. Section 2 briefly reviews data fusion and applications and fuzzy Kalman filters. Section 3 describes the model and equations. Sections 4 and 5 contain the results of the default model tested and conclusions, respectively.

Various state-of-the-art fusion techniques have been used with data obtained from physical sensors such as three-dimensional cameras, ultrasound, sonar, and infrared. For example, the output capability of a ball mill pulverizing system mainly represents the efficiency of the system. To measure output capability, researchers proposed a novel feature extraction method based on multi-sensor data fusion and implemented a FastICA to extract independent components from the field data, which included six sensor measurements (Wang, Zhu, Si, & Zang, 2008). However, sensor data alone are always uncertain to some extent because of noise and possible sensor failures (Ghahroudi & Fasih, 2007). Many approaches have been reported for multi-sensor data fusion under Gaussian noise (e.g., the minimum variance approach, the maximum kurtosis approach, the minimum kurtosis approach, and the minimum mean absolute error approach (Niu, Zhu, Gu, & Chu, 2009)), but there are few approaches for fusion under non-Gaussian noise. Data fusion algorithms have a wide range of applications in some fields, but with the increasing number of sensors in multi-sensor target tracking systems, algorithms that rely on conventional Kalman filters meet with heavy computational burden and poor robustness (Zhang & Wang, 2010).

In one network security situation awareness study, researchers adopted Snort and NetFlow as sensors to gather real network traffic and fused them using a multi-layer feed-forward neural network that could solve multi-class problems (Wang, Liu, Lai, & Liang, 2007). Multi-sensor data fusion has also been applied to a new software compensation method to counteract the effect of temperature on optical fiber displacement sensors. In this method, the outputs of different displacement sensors are processed using data fusion theory, and subsequently the effect of temperature is overcome. This practical method dispenses with the additional temperature sensor, making use of just two different outputs of optical fiber displacement sensors (Wei, Ma, Yang, & Liu, 2006). In another example, statistical learning theory was used to detect oil tube defects on support vector machines. Original information was obtained by multi-group vortex sensors and leakage magnetic sensors (Tian, Gao, & Li, 2006).

Using distributed multiple sensors to track objects is an important area of research in the fields of autonomous robotics, military applications, and mobile systems. Researchers have reviewed a number of computationally intelligent methods for developing robust tracking schemes through sensor data fusion (Smith & Singh, 2006). For example, a new data fusion method was developed to more precisely measure a vibration signal in rotary machine fault diagnosis. This method uses correlation to determine the weighted value. It requires no prior knowledge about sensors, and the weighted value of the sensor data can be determined based on the correlation of real-time data tested in the fusion process (Xuejun, Guangfu, & Dhillon, 2009).

Multi-sensor information fusion technology is widely used in various fields, particularly the military. To enhance the capability of ship chemical defense support in future sea warfare, a model of the naval ship chemical detection system was built based on multi-sensor information fusion. This model was able to obtain information on chemical agents, carry out feature extraction and selection of chemical agents through wavelet analysis, and then use neural networks to manage the multi-sensor data (Zhang, Song, Li, Yu, & Bao, 2009). Image fusion is a tool for interpreting multi-sensor, multi-temporal, and multi-resolution data. Multi-source

data vary in spectral, spatial, and temporal resolution, which necessitates the use of advanced analytical techniques (Uttam, Chiranjit, & Ramachandra, 2009). For example, a robot gripper contains many types of sensors (e.g., force sensors, proximity sensors, displacement sensors). However, because of measurement error and uncertainty, data from one sensor alone are insufficient to determine the state of connection. In order for the robot to grasp objects safely and reliably, researchers must use information fusion on the multi-sensor output data (Sun, Zhu, & Wu, 2009).

The existing research on image fusion focuses mainly on theory. However, a multi-sensor data fusion model for a universal, real-time fusion system based on a real-time operating system has been proposed. A VersaModular Eurocard bus and distributed parallel computer architecture was given, and the kernel component was a real-time operating system of the standard bus (Liu, Wang, & Cui, 2009). Data on temperature, humidity, ventilation, and other environmental parameters were obtained from multiple sensors and passed through a fuzzy filter. As a result of synthetic operation with a decision rule at the data fusion center, an accurate state parameter estimation of the environment was obtained and then used in the final decision-making process (Sun, Wang, Cao, He, & Yan, 2009; Sun, Zhu, et al., 2009).

In another application of multi-sensor data fusion, the effects of massive noise and uncertainty factoring in traffic data were studied. Because it was difficult to obtain accurate data from a single sensor, a three-layer structure was used to acquire traffic data based on multi-sensor data fusion. In the first layer, data from different data sources were integrated into a uniform data stream. In the second layer, a fuzzy neural network was used to predict the queue length and traffic time. In the third layer, a multi-phase fuzzy controller for intersection based on queue length was designed (Kong & Guo, 2006). Multi-sensor data fusion has also been used to improve the ability of naval ships in the sea combat. Target information obtained from multiple sensors on the ocean floor enabled ships to improve target locating precision and reduce target locating errors. However, the study demonstrated that it is impossible to eliminate locating errors completely. Because these errors greatly affect command decisions, more research in this area is essential (Yang, Cheng, Wei, & Lu, 2006). Another multi-sensor data fusion study was conducted to locate areas of air pollution around Oklahoma City. An interpolation method was used to fuse multi-spectrum and multi-sensor remotely sensed images, preserving appropriate textural features (Zia, DeBrunner, ChinnaSwamy, & DeBrunner, 2002). The US Naval Research Laboratory has developed an affordable, multi-sensor, real-time detection system for damage control and situational awareness. The system, called Volume Sensor, provides standoff identification of events within a space (e.g., flaming and smoldering fires, pipe ruptures, released gas) for US Navy vessels. Data fusion was used to integrate spectral sensors, acoustic sensors, and video image detection algorithms (Minor, et al., 2007).

A multi-sensor image fusion and interactive mining system based on neural models of color vision processing, learning, and pattern recognition was pioneered at MIT Lincoln Laboratory. The research initially focused on color-fused night vision (low-light visible and uncooled thermal imagery) but was later extended to multi-spectrum infrared and three-dimensional radar. The research also developed a proof-of-concept system for Ethernet Output, infrared, and synthetic aperture radar fusion and mining (Fay, Ivey, Bomberger, & Waxman, 2003). Another study presented a mechanism for providing differential data protection to publish/subscribe distributed systems, such as those used in peer-to-peer computing, grid environments, and so on. This mechanism, termed *security overlays*, incorporates credential-based communication channel creation, subscription, and extension (Widener, Schwan, & Bustamante, 2003). System robustness against individual sensor

failures is an important concern in multi-sensor networks, but unfortunately the complexity associated with using the remaining sensors to interpolate missing data from the failed sensor grows exponentially because of the “curse of dimensionality” (Dong & Pentland, 2006).

Multi-sensor fusion technology has also been used to diagnose faults in a vehicle transmission system. With the use of a hybrid fusion pattern based on artificial neural networks, the robustness of a diagnosing system improved greatly (Wu, Liu, & Sharma, 2007). The minimum entropy fusion approach for multi-sensor data fusion in non-Gaussian environments was investigated, and fused data in the form of the weighted sum of the multi-sensor outputs and the use of the varimax norm as the information measure were employed. Results of a numerical simulation showed the effectiveness of the approach (Zhou & Leung, 1997). In another work, a new approach to distributed sensor data fusion systems in multi-target tracking called *tessellated sensor data fusion* centered around a geographical partitioning (tessellation) of the data. A functional decomposition divided sensor data fusion into components that could be assigned to processing units, parallelizing the processing. The tessellation implicitly defined the set of tracks, potentially yielding correlations with the sensor plots (observations) in a tile (Storms, van Veelen, & Boasson, 2005).

Other researchers have proposed the use of a heuristic method for a resource-constrained project-scheduling problem with fuzzy activity times. This method was based on a priority rule for a parallel schedule generation scheme (Bhaskar, Pal, & Pal, 2011). Kao and Lin (2011) proposed that qualitative data be viewed as fuzzy numbers and used data envelopment analysis multipliers associated with decision-making units being evaluated to construct membership functions. Cluster analyses involve sets of heuristic algorithms, and *cluster analysis* is a generic name for a variety of procedures that can be used to group data into natural groupings (Hall, 1992). Factor analysis is used to form a correlation matrix of similarities among cases. Classical factor analysis is then performed on the $N \times N$ matrix, and the data are assigned to clusters based on their factor loadings (Skinner, 1979). Fuzzy clustering has been applied to many decision-making problems, from auditing to financial management to strategic portfolio management to multi-objective decision making (Ammar, Wright, & Selden, 2000; Tan & Hsieh, 2005; Yager, 1981).

3. The IVHMS

The following section presents several figures to assist the reader in gaining an understanding of the IVHMS used to record diagnostic performance data and parameters from external sensors and vehicle data busses. Fig. 1 illustrates that there are two vehicle data busses that could be monitored by the data logger; Controller Area Network (CAN) and the SAE standard J1708. The CAN data bus uses a SAE J1939 messaging protocol while the J1708 data bus uses a SAE J1708 messaging protocol. Examples of data parameters available using the J1939 protocol include: input shaft speed, transmission oil temperature, engine oil pressure, and fuel temperature. Examples of data parameters available using the J1708 protocol include: output shaft speed, fuel rate, engine speed, barometric pressure, and fuel level.

The architectural drawing, shown below in Fig. 2, displays a possible scenario for collecting, transmitting, storing, and displaying sensor data. Data is sent from the on-board device to the backend servers via wireless communications (or by manual retrieval if required) to a local intermediate File Transfer Protocol (FTP) server. That server can store and forward all data files from the on-board device via a secure Virtual Private Network (VPN) connection to backend servers. Several servers, each with a specific function can post-process the data files and store the data for later review. The

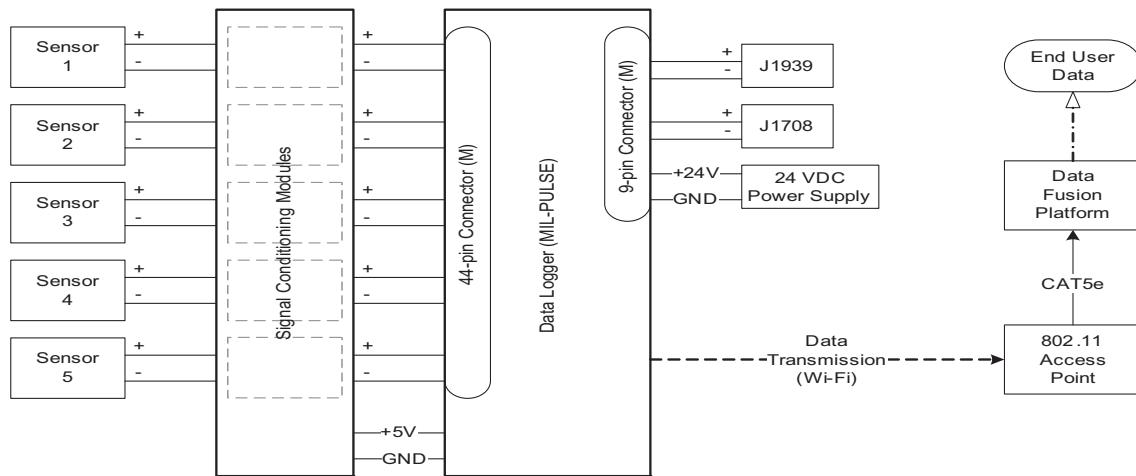


Fig. 1. High-level system prototype schematic. GND, ground; MIL-PULSE, Military Prognostic and Logistical Status Environment; VDC, volts direct current.

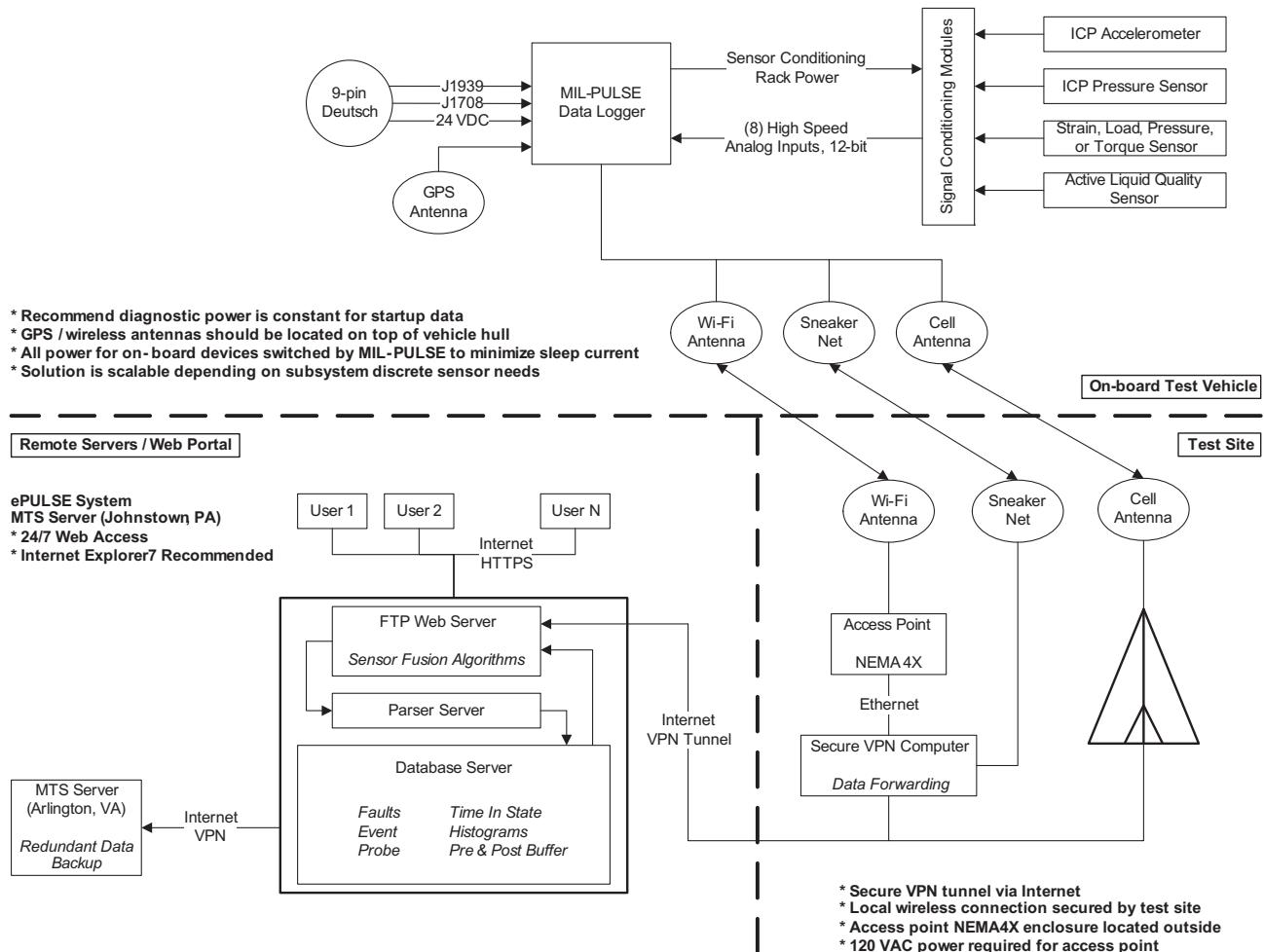


Fig. 2. Sensor fusion architectural drawing.

FTP server handles incoming data and the web portal interface and the parser server breaks down the data files into raw data for storage on the database server(s). Data can typically be stored in fault, event, probe, time-in-state, histogram, and pre/post fault buffer formats. Additionally, data can be backed up on a redundant server.

The architectural drawing, shown below in Fig. 3, illustrates how the sensor data was collected, transmitted, stored, and displayed.

Figs. 4–13 show how critical status and alerts were recorded as well as documenting the identity of input signals, HEMTT sensors, business rules, operation mode and alert groupings.

CriticalityStatusID	Definition	Notes	AlertSeverity
0	Normal	No Alert	0
1	Status Message	PTO engaged, starting, ect	0
2	Critical High	Critical High alert	2
3	Warn High	Warn High alert	1
4	Warn Low	Warn Low alert	1
5	Critical Low	Critical Low alert	2

Fig. 3. Criticality ID.

PGN	IN Tra	CAN_ID	SignalVarName	Tag	Sub	Type	rate_1	Indte_2	Indte_3	Indte_4	Indte_5	Indte_6	Indte_7	Indte_8	Ind	mx	b	Low	High	Units	owardToCs
84	65265 00	18FFF100	VehicleSpeed	VSO1	-1	0	2	1	-1	-1	-1	-1	-1	-1	-1	0.002427	0.000000	0.00	251.00	MPH	TRUE
91	61443 00	OCF00300	AccPosition	AP01	-1	0	1	-1	-1	-1	-1	-1	-1	-1	-1	0.400000	0.000000	0.00	100.00	%	TRUE
100	65263 00	18FEFF000	EngOilPress	EP01	-1	0	3	-1	-1	-1	-1	-1	-1	-1	-1	0.580151	0.000000	0.00	145.00	PSI	TRUE
102	65270 00	18FEFF600	BoostPress	TP01	-1	0	1	-1	-1	-1	-1	-1	-1	-1	-1	0.290075	0.000000	0.00	73.00	PSI	TRUE
105	65270 00	18FFF600	IntakeManifold1Temp	IT01	-1	0	2	-1	-1	-1	-1	-1	-1	-1	-1	1.800000	-40.000000	-40.00	410.00	F	TRUE
110	65262 00	18FEEE00	EngCoolantTemp	EC01	-1	0	0	-1	-1	-1	-1	-1	-1	-1	-1	1.800000	-40.000000	-40.00	410.00	F	TRUE
161	61442 03	OCF00203	InputShaftSpeed	IS01	-1	0	6	5	-1	-1	-1	-1	-1	-1	-1	0.125000	0.000000	0.00	8032.00	RPM	TRUE
174	65262 12	18FEEE12	FuelTemp	FT01	-1	0	1	-1	-1	-1	-1	-1	-1	-1	-1	1.800000	-40.000000	-40.00	410.00	F	TRUE
175	65262 00	18FEEE00	EngOil1Temp	OT01	-1	0	3	2	-1	-1	-1	-1	-1	-1	-1	0.056250	-459.400000	-460.00	3155.00	F	TRUE
177	65272 03	18FEFF803	TransOilTemp	TO01	-1	0	5	4	-1	-1	-1	-1	-1	-1	-1	0.056250	-459.400000	-460.00	3155.00	F	TRUE
183	65266 12	18FEEF212	FuelRate	FR01	-1	0	1	0	-1	-1	-1	-1	-1	-1	-1	0.189271	0.000000	0.00	845.00	GPH	TRUE
185	65266 12	18FEEF212	AvgFuelEconomy	FE01	-1	0	5	4	-1	-1	-1	-1	-1	-1	-1	4.233863	0.000000	0.00	78.00	MPG	TRUE
190	61444 00	OCF00400	EngineSpeed	ES01	-1	0	4	3	-1	-1	-1	-1	-1	-1	-1	0.125000	0.000000	0.00	8032.00	RPM	TRUE
191	61442 03	OCF00203	OutputShaftSpeed	OS01	-1	0	2	1	-1	-1	-1	-1	-1	-1	-1	0.125000	0.000000	0.00	8032.00	RPM	TRUE
260	65280 00	18FF00A0	CoolantLevel	CL01	-1	1	0	-1	-1	-1	-1	-1	-1	-1	-1	0.001000	0.000000	0.00	1.00	Boolean	FALSE
261	65281 A0	18FF01A0	AirFilterCond	FC01	-1	0	1	0	-1	-1	-1	-1	-1	-1	-1	0.001000	0.000000	0.00	20.00	PSI	FALSE
262	65282 A0	18FF02A0	EngineOilLevel	OL01	-1	1	0	-1	-1	-1	-1	-1	-1	-1	-1	0.001000	0.000000	0.00	1.00	Boolean	FALSE
263	65283 A0	18FF03A0	EngineOilAnalysis	OA01	-1	0	1	0	-1	-1	-1	-1	-1	-1	-1	0.001000	0.000000	0.00	100.00	%	FALSE
264	65284 A0	18FF04A0	FuelLevel	FL01	-1	0	1	0	-1	-1	-1	-1	-1	-1	-1	0.001000	0.000000	0.00	100.00	%	FALSE
265	65285 A0	18FF05A0	FuelFilterSep	FS01	-1	0	0	-1	-1	-1	-1	-1	-1	-1	-1	0.001000	0.000000	0.00	1.00	Boolean	FALSE
266	65286 A0	18FF06A0	FuelFilterCond	CS01	-1	0	0	-1	-1	-1	-1	-1	-1	-1	-1	0.001000	0.000000	0.00	1.00	Boolean	FALSE
267	65300 A8	18FF14A8	TransLevel	TL01	-1	0	0	-1	-1	-1	-1	-1	-1	-1	-1	0.001000	0.000000	0.00	1.00	Boolean	FALSE
268	65301 A8	18FF15A8	BrakeWear1	BW01	-1	0	0	-1	-1	-1	-1	-1	-1	-1	-1	0.001000	0.000000	0.00	1.00	Boolean	FALSE
269	65340 B2	18FF3CB2	Alternator1Voltage	AV01	-1	0	1	0	-1	-1	-1	-1	-1	-1	-1	0.001000	0.000000	0.00	36.00	V	FALSE
270	65341 B2	18FF3DB2	Alternator1Temp	AT01	-1	0	1	0	-1	-1	-1	-1	-1	-1	-1	0.001000	0.000000	-50.00	572.00	F	FALSE
271	65342 B2	18FF3EB2	Alternator1Current	AC01	-1	0	1	0	-1	-1	-1	-1	-1	-1	-1	0.001000	-30.000000	-30.00	30.00	A	FALSE
272	65343 B2	18FF3FB2	Starter1Voltage	SV01	-1	0	1	0	-1	-1	-1	-1	-1	-1	-1	0.001000	0.000000	0.00	36.00	V	FALSE
273	65344 B2	18FF40B2	Starter1Temp	ST01	-1	0	1	0	-1	-1	-1	-1	-1	-1	-1	0.001000	0.000000	-50.00	572.00	F	FALSE

Fig. 4. Processed input signals.

Signal ID	Signal Name	Signal Source	Sensor MFG	Sensor PN	Power Requirements	Output Range	Sensor Pin Out	SPN	PGN	Bytes	Length	Low	High	Units
S4	Engine Speed	CAN	J1939					190	61444	4-5	2 bytes	0.00	8032.00	RPM
S5	Accelerator Pedal Position 1	CAN	J1939					91	61443	2	1 byte	0.00	100.00	%
S22	Engine Coolant Temperature	CAN	J1939					110	65262	1	1 byte	-40.00	410.00	F
S23	Engine Oil Temperature 1	CAN	J1939					175	65262	3-4	2 bytes	-460.00	3155.00	F
S24	Engine Oil Pressure	CAN	J1939					100	65263	4	1 byte	0.00	145.00	PSI
S27	Boost Pressure	CAN	J1939					102	65270	2	1 byte	0.00	73.00	PSI
S28	Intake Manifold 1 Temperature	CAN	J1939					105	65270	3	1 byte	-40.00	410.00	F
S29	Coolant Level	ARL-DAQ	Oshkosh	2001470	+5v, GND	0-5v	A(Low), B(GND), C(+5v), D(Norm)	260	65280	1	1 byte	0.00	1.00	Boolean
S30	Air Filter condition	ARL-DAQ	Oshkosh	3246337	+5v	0-5v	A(Diff +), B(+5v), C(Diff -)	261	65281	1-2	2 bytes	0.00	20.00	PSI
S31	Engine Oil Level	ARL-DAQ	Oshkosh	2001470	+5v, GND	0-5v	A(Low), B(GND), C(+5v), D(Norm)	262	65282	1	1 byte	0.00	1.00	Boolean
S32	Engine Oil Analysis	ARL-CAN	Unknown					263	65283	1-2	2 bytes	0.00	100.00	%
S33	Fuel Temperature	CAN	J1939					174	65262	2	1 byte	-40.00	410.00	F
S34	Fuel Rate	CAN	J1939					183	65266	1-2	2 bytes	0.00	845.00	GPH
S35	Average Fuel Economy	CAN	J1939					185	65266	5-6	2 bytes	0.00	78.00	MPG
S36	Fuel Level	ARL-DAQ	HEMTT				O-Source	264	65284	1-2	2 bytes	0.00	100.00	%
S114	Fuel Filter Separator	ARL-DAQ	Racor					265	65285	1-1	1 byte	0.00	1.00	Boolean
S37	Fuel Filter Condition Switch	ARL-DAQ	Oshkosh	3290338	+5v	0-5v	A(High), B(+5v)	266	65286	1	1 byte	0.00	1.00	Boolean
S41	Current Gear	CAN	J1939					523	61445	4	1 byte	-125.00	125.00	Integer
S45	PTO State	CAN	J1939					976	65265	7.1	5 bits	0.00	1.00	Boolean
S49	Transmission Oil Temperature	CAN	J1939					177	65272	5-6	2 bytes	-460.00	3155.00	F
S50	Driveline Engaged	CAN	J1939					560	61442	1-1	2 bits	0.00	1.00	Boolean
S53	Output Shaft Speed	CAN	J1939					191	61442	2-3	2 bytes	0.00	8032.00	RPM
S54	Input Shaft Speed	CAN	J1939					161	61442	7-8	2 bytes	0.00	8032.00	RPM
S56	Transmission Level	ARL-DAQ	Oshkosh	2001470	+5v, GND	0-5v	A(Low), B(GND), C(+5v), D(Norm)	267	65300	1	1 byte	0.00	1.00	Boolean
S57	Wheel-Based Vehicle Speed	CAN	J1939					84	65265	2-3	2 bytes	0.00	251.00	MPH
S74	Front Brake Wear	PSU-ARL			+5v, GND	0-5v	A(Low), B(GND), C(+5v)	268	65301	1	1 byte	0.00	1.00	Boolean
S74	Rear Brake Wear	PSU-ARL			+5v, GND	0-5v	A(Low), B(GND), C(+5v)	268	65302	1	1 byte	0.00	1.00	Boolean
S77	Alternator Voltage	ARL-DAQ	PSU-ARL					269	65340	1-2	2 bytes	0.00	36.00	V
S78	Alternator Temperature	ARL-DAQ	Sensata	B5024R0695	+5v	0-36v	Thermistor	270	65341	1-2	2 bytes	-50.00	572.00	F
S79	Alternator Current	ARL-DAQ	LEM	DHAB S/18	+5v, GND	0-5v	A, B	271	65342	1-2	2 bytes	-350.00	350.00	A

Fig. 5. Signal ID.

4. The fuzzy Kalman filter: design methodology

4.1. Analysis and partition of the IVHMS system

Given the aforementioned applications of multi-sensor data fusion, it is likely that fuzzy control can be applied to vehicle maintenance, for which rules are established for such continuous variables as accelerator pedal position, actual engine (percent torque), gear ratio, boost pressure, current gear, driver's demand engine (percent torque), electrical potential, engine speed, fuel

rate, input shaft speed, fuel economy, output shaft speed, percent load at current speed, selected gear, and wheel-based vehicle speed. Factor analysis places these 15 variables into four clusters (i.e., gear, engine, fuel, and electrical; see Table 1).

4.2. Defining the input and output membership functions

Characteristics of each of these variables have values within a certain range (e.g., fuel economy may be described as low or high). Each of these states can be expanded to fully specify the variable,

SubSystem	User Classification		Sensor			CAN ID								
	Signal Source	Operator	MFG	PN	Signal Name	Power Req	Output Range	Sensor Pin Out	Notes	SPN	PGN	Bytes	Length	Source Units
Engine System (Source Address 0x00)														
CAN	X	X	\$4	J1939	Engine Speed				J1939	190	61444	4-5	2 bytes	RPM
CAN	X	X	\$5	J1939	Accelerator Pedal Position 1				J1939	91	61443	2	1 byte	%
CAN	X	X	\$22	J1939	Engine Coolant Temperature				J1939	110	65262	1	1 byte	C
CAN	X	X	\$28	J1939	Engine Oil Temperature 1				J1939	175	65262	8-4	2 bytes	C
CAN	X	X	\$24	J1939	Engine Oil Pressure				J1939	100	65263	4	1 byte	kPa
CAN	X	X	\$27	J1939	Boost Pressure				J1939	102	65270	2	1 byte	kPa
CAN	X	X	\$28	J1939	Intake Manifold 1 Temperature				J1939	105	65270	3	1 byte	C
ARL-DAQ	X	X	\$29	Oshkosh	2001470	Coolant Level	+5v, GND	0-5v	A(Low), B(GND), C(+5v), D(Norm)	260	65280	1	1 byte	Boolean
ARL-DAQ	X	X	\$30	Oshkosh	3246357	Air Filter condition	+5v	0-5v	A(Diff+), B(+5v), C(Diff-)	261	65281	1-2	2 bytes	kPa
ARL-DAQ	X	X	\$31	Oshkosh	2001470	Engine Oil Level	+5v, GND	0-5v	A(Low), B(GND), C(+5v), D(Norm)	262	65282	1	1 byte	Boolean
ARL-CAN	X	X	\$32	Unknown		Engine Oil Analysis				263	65283	1-2	2 bytes	%
Fuel System (Source Address 0x12)														
CAN	X	X	\$33	J1939	Fuel Temperature				J1939	174	65262	2	1 byte	C
CAN	X	X	\$34	J1939	Fuel Rate				J1939	183	65266	1-2	2 bytes	L/H
CAN	X	X	\$35	J1939	Average Fuel Economy				J1939	185	65266	5-6	2 bytes	L/100KM
ARL-DAQ	X	X	\$36	HEMTT		Fuel Level			0-Source	264	65284	1-2	2 bytes	%
ARL-DAQ	X	X	\$114	Racor		Fuel Filter Separator				265	65285	1	1 byte	Boolean
ARL-DAQ	X	X	\$37	Oshkosh	3290388	Fuel Filter Condition Switch	+5v	0-5v	A(High), B(+5v)	266	65286	1	1 byte	Boolean
Transmission System (Source Address 0x03)														
CAN	X	X	\$41	J1939	Current Gear				J1939	523	61445	4	1 byte	Integer
CAN	X	X	\$45	J1939	PTO State				J1939	976	65265	7-1	5 bits	Boolean
CAN	X	X	\$49	J1939	Transmission Oil Temperature				J1939	177	65272	5-6	2 bytes	C

Fig. 6. HEMTT sensors.

	Group ID	Alert Name	Signal ID	OperatingModeID	CriticalityStatusID	InputSensedItem	InputSignalVarName	SensorTag	Threshold	ThresholdType	Units
A1	G1	Engine Speed Alert	\$4	M4	2	Engine Speed	EngineSpeed	E501	3000	>>	rpm
A2	G1	Engine Speed Alert	\$4	M4	4	Engine Speed	EngineSpeed	E501	450	<<	rpm
A3	G2	Engine Coolant Temperature Alert	\$22	M2	2	Engine Coolant Temperature	EngCoolantTemp	EC01	220	>>	F
A4	G2	Engine Coolant Temperature Alert	\$22	M2	3	Engine Coolant Temperature	EngCoolantTemp	EC01	210	>>	F
A5	G2	Engine Coolant Temperature Alert	\$22	M2	5	Engine Coolant Temperature	EngCoolantTemp	EC01	-50	<<	F
A6	G3	Engine Oil Temperature Alert	\$23	M2	2	Engine Oil Temperature	EngOilITemp	OT01	260	>>	F
A7	G3	Engine Oil Temperature Alert	\$28	M2	3	Engine Oil Temperature	EngOilITemp	OT01	250	>>	F
A8	G3	Engine Oil Temperature Alert	\$23	M2	5	Engine Oil Temperature	EngOilITemp	OT01	-50	<<	F
A9	G4	Engine Oil Pressure Alert	\$24	M4	3	Engine Oil Pressure	EngOilPress	EP01	100	>>	PSI
A10	G4	Engine Oil Pressure Alert	\$24	M4	4	Engine Oil Pressure	EngOilPress	EP01	15	<<	PSI
A11	G4	Engine Oil Pressure Alert	\$24	M4	5	Engine Oil Pressure	EngOilPress	EP01	5	<<	PSI
A12	G5	Engine Boost Pressure Alert	\$27	M4	3	Engine Boost Pressure	BoostPress	TP01	30	>>	PSI
A13	G5	Engine Boost Pressure Alert	\$27	M8	5	Engine Boost Pressure	BoostPress	TP01	10	<<	PSI
A14	G6	Engine Intake Manifold Temperature Alert	\$28	M4	2	Engine Intake Manifold Temperature	IntakeManifoldTemp	IT01	300	>>	F
A15	G7	Engine Coolant Level Alert	\$29	M0	5	Engine Coolant Level	CoolLevel	CL01	Boolean	equals	
A17	G8	Engine Air Filter Restriction Alert	\$30	M9	2	Engine Air Filter Restriction	AirFilterCond	FC01	8	>>	PSI
A18	G8	Engine Air Filter Restriction Alert	\$30	M8	5	Engine Air Filter Restriction	AirFilterCond	FC01	0	equals	PSI
A19	G9	Engine Oil Level Alert	\$31	M0	5	Engine Oil Level	EngOilLevel	OL01	Boolean	equals	
A20	G10	Engine oil quality Alert	\$32	M0	1	Engine oil quality	EngOilAnalysis	OA01	0	<<	%
A21	G11	Fuel Temperature Alert	\$33	M0	4	Fuel Temperature	FuelTemp	FT01	0	<<	F
A22	G12	Fuel Level Alert	\$36	M2	4	Fuel Level	FuelLevel	FL01	10	<<	%
A23	G13	Fuel Filter Restriction Alert	\$37	M4	2	Fuel Filter Restriction	FuelFilterCond	CS01	Boolean	equals	
A24	G14	Transmission Oil Temperature Alert	\$49	M2	2	Transmission Oil Temperature	TransOilTemp	TO01	260	>>	F
A25	G14	Transmission Oil Temperature Alert	\$49	M2	3	Transmission Oil Temperature	TransOilTemp	TO01	250	>>	F
A26	G14	Transmission Oil Temperature Alert	\$49	M2	5	Transmission Oil Temperature	TransOilTemp	TO01	-50	<<	F
A27	G15	Transmission Oil Level Alert	\$56	M0	5	Transmission Oil Level	TransLevel	TL01	Boolean	equals	
A28	G16	Wheel based vehicle speed Alert	\$57	M2	2	Wheel based vehicle speed	VehicleSpeed	VS01	65	>>	MPH
A29	G17	Front Break Wear Alert	\$74	M0	4	Front Break Wear	BreakWear1	BW01	Boolean	equals	
A30	G17	Rear Break Wear Alert	\$74	M0	4	Rear Break Wear	BreakWear2	BW02	Boolean	equals	
A31	G18	Alternator Voltage Alert	\$77	M4	2	Alternator Voltage	AlternatorVoltage	AV01	80	>>	V

Fig. 7. Alerts and business rules.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
SignalSource	Signal ID	Sensor Source	Signal Name	SPN	PGN	Bytes	Length	Signal Low	Signal High	Units	id	tag	mx	b
ARL-CAN	225	MACS-DISPLAY MDL	Cab Illumination	1487	53248	1	1 byte	0	100	%	1000	CIO1	0.4	0
ARL-CAN	226	MACS-BODYCTRL	High Beam Head Light	2347	65089	1.7	2 bytes	0	3	Boolean	1001	HBL1	1	0
ARL-CAN	227	MACS-BODYCTRL	Left Turn Signal	2367	65089	2.7	2 bytes	0	3	Boolean	1002	LTS1	1	0
ARL-CAN	228	MACS-BODYCTRL	Right Turn Signal	2369	65089	2.5	2 bytes	0	3	Boolean	1003	RTS1	1	0
ARL-CAN	229	MACS-DISPLAY MDL	Tractor Rear High Mounted W	511	65505	1.3	2 bytes	0	3	Boolean	1004	TRL1	1	0
ARL-CAN	230	MACS-DISPLAY MDL	Tractor Clearance Light	512	65505	1.5	2 bytes	0	3	Boolean	1005	TCL1	1	0
ARL-CAN	231	MACS-DISPLAY MDL	Jacobs On/Off	500	65500	1.1	2 bytes	0	3	Boolean	1006	JB01	1	0
ARL-CAN	232	MACS-DISPLAY MDL	Jacobs High/Low	501	65501	1.3	2 bytes	0	3	Boolean	1007	JHL1	1	0
ARL-CAN	233	MACS-BODYCTRL	8x8 Drive	502	65502	1.1	2 bytes	0	3	Boolean	1014	EIG1	1	0
ARL-CAN	234	MACS-BODYCTRL	Park Brake	503	65503	1.1	2 bytes	0	3	Boolean	1015	PB01	1	0
ARL-CAN	235	MACS-BODYCTRL	Diff Lock	504	65504	1.1	2 bytes	0	3	Boolean	1016	DFL1	1	0
ARL-CAN	236	MACS-DISPLAY MDL	Dome Light	505	65505	1.1	2 bytes	0	3	Boolean	1008	DOL1	1	0
ARL-CAN	237	MACS-DISPLAY MDL	AC Power	506	65506	1.1	2 bytes	0	3	Boolean	1009	ACP1	1	0
ARL-CAN	239	MACS-DISPLAY MDL	Diag Capture	508	65508	1.1	2 bytes	0	3	Boolean	1011	DCC1	1	0
ARL-CAN	240	MACS-DISPLAY MDL	Commander Display Heartbeat	509	65509	1.1	2 bytes	0	3	Boolean	1012	RDH1	1	0
ARL-CAN	241	MACS-DISPLAY MDL	Driver Display Heartbeat	510	65510	1.1	2 bytes	0	3	Boolean	1013	LDH1	1	0
ARL-CAN	242	MACS-DISPLAY MDL	Driver Display Mode	513	65511	1.1	4 bytes	0	15	Boolean	1017	LDMC	1	0
ARL-CAN	243	MACS-DISPLAY MDL	Driver Display Cycle	514	65511	1.5	2 bytes	0	3	Boolean	1018	LDMC	1	0
ARL-CAN	238	MACS-DISPLAY MDL	Commander Display Mode	507	65507	1.1	4 bytes	0	15	Boolean	1010	RDMC	1	0
ARL-CAN	244	MACS-DISPLAY MDL	Commander Display Cycle	515	65507	1.5	2 bytes	0	3	Boolean	1019	LDMC	1	0

Fig. 8. Input/output signals.

	Definition	Mode Preceder	Notes	Signal Var Name	Signal ID	Tag	Threshold Type	Threshold	Units	Signal Var Name	Signal ID
M0	Master Switch On		11 System powered up	AlternatorVoltage	S77	AV01	>>	1	V		
M1	Master Switch On, Engine Off		10 Engine speed >0 RPM	AlternatorVoltage	S77	AV01	>>	1	V	FuelLevel	S36
M2	Master Switch On, Engine On		9 Engine Speed <300 RPM	EngineSpeed	S4	ES01	equals	0	RPM		
M3	Master Switch On, Engine Start		8 Engine Speed >300 RPM	EngineSpeed	S4	ES01	<<	300	RPM		
M4	Master Switch On, Engine On, Engine Running		7 Engine Speed >300 RPM	EngineSpeed	S4	ES01	>>	300	RPM	PTO_State	S45
M5	Master Switch On, Engine On, Engine Running, PTO Engaged		6 Engine Speed >300 RPM, PTO engaged	EngineSpeed	S4	ES01	>>	300	RPM	EngCoolantTemp	S22
M6	Master Switch On, Engine On, Engine Running, Operating Temp		5 Engine Speed >300 RPM, Coolant temp >160F	EngineSpeed	S4	ES01	>>	800	RPM	PTO_State	S45
M7	Master Switch On, Engine On, Engine Running, Operating Temp, PTO Engaged		4 Engine Speed >300 RPM, PTO engaged, Coolant temp >160F	EngineSpeed	S4	ES01	>>	1500	RPM		
M8	Master Switch On, Engine On, Engine Running, High Engine Speed		3 Engine Speed >1500 RPM	EngineSpeed	S4	ES01	>>	1000	RPM		
M9	Master Switch On, Engine On, Engine Running, Low Engine Speed		2 Engine Speed <1000 RPM	EngineSpeed	S4	ES01	<<	1000	RPM		
M10	Master Switch On, Engine On, Engine Running, Vehicle Moving		1 Vehicle Speed > 5 MPH	VehicleSpeed	S57	VS01	>>	5	MPH		

Note: Node precedence is evaluated against the "Group ID". So for Group ID X where both Operating Mode IDs 5 and 7 are active, Operating Mode 7 would take precedence over Operating Mode 5.

Fig. 9. Operation mode ID.

	Drivetrain	Hydraulic	Electrical	Alternator	Battery	Starter	Ultra Cap
EngineSpeedAlert1	TransmissionOilTemperatureAlert1	HydraulicReservoirLevelAlert	AlternatorVoltageAlert1	AlternatorVoltageAlert1	BatteryVoltage1Alert1	StarterVoltageAlert	UltracapVoltageAlert1
EngineSpeedAlert2	TransmissionOilTemperatureAlert2	PTOHydraulicPressureAlert1	AlternatorVoltageAlert2	AlternatorVoltageAlert2	BatteryVoltage1Alert2	StarterTemperatureAlert	UltracapVoltageAlert2
EngineCoolantTemperatureAlert1	TransmissionOilTemperatureAlert3	PTOHydraulicPressureAlert2	AlternatorVoltageAlert3	AlternatorTemperatureAlert	BatteryVoltage2Alert1		UltracapTemperatureAlert1
EngineCoolantTemperatureAlert2	TransmissionOilLevelAlert	PTOHydraulicPressureAlert3	AlternatorTemperatureAlert	AlternatorCurrentAlert1	BatteryVoltage2Alert2		UltracapTemperatureAlert2
EngineCoolantTemperatureAlert3	WheelbasevehicleSpeedAlert	ManifoldHydraulicPressureAlert1	AlternatorCurrentAlert1	AlternatorCurrentAlert2	BatteryVoltage3Alert1		UltracapCurrentAlert
EngineOilTemperatureAlert1	FrontBreakWearAlert	ManifoldHydraulicPressureAlert2	AlternatorCurrentAlert2	AlternatorCurrentAlert2	BatteryVoltage3Alert2		
EngineOilTemperatureAlert2	RearbreakWearAlert	ManifoldHydraulicPressureAlert3	BatteryVoltage4Alert1		BatteryVoltage4Alert1		
EngineOilTemperatureAlert3	AirBrakePressureChannel1Alert2	CraneHydraulicPressureAlert	BatteryVoltage4Alert2		BatteryVoltage4Alert2		
EngineOilPressureAlert1	AirBrakePressureChannel1Alert2	CraneHydraulicPressureAlert1	BatteryVoltage4Alert1		BatteryPackVoltageAlert1		
EngineOilPressureAlert2	AirBrakePressureChannel1Alert1	CraneHydraulicPressureAlert2	BatteryVoltage2Alert2		BatteryPackVoltageAlert2		
EngineOilPressureAlert3	AirBrakePressureChannel2Alert2	MainWInchHydraulicPressureAlert1	BatteryVoltage3Alert1		BatteryTemperature1Alert1		
EngineBoostPressureAlert1		MainWInchHydraulicPressureAlert2	BatteryVoltage3Alert2		BatteryTemperature1Alert2		
EngineBoostPressureAlert2		MainWInchHydraulicPressureAlert3	BatteryVoltage4Alert1		BatteryTemperature2Alert1		
EngineIntakeManifoldTemperatureAlert		HydraulicSupplyFilterDifferentialPressureAlert1	BatteryVoltage4Alert2		BatteryTemperature2Alert2		
EngineCoolantLevelAlert		HydraulicSupplyFilterDifferentialPressureAlert2	BatteryPackVoltageAlert1		BatteryTemperature3Alert1		
EngineAirFilterRestrictionAlert1		HydraulicSupplyFilterDifferentialPressureAlert3	BatteryPackVoltageAlert2		BatteryTemperature3Alert2		
EngineAirFilterRestrictionAlert2		HydraulicReturnFilterDifferentialPressureAlert1	BatteryTemperature1Alert1		BatteryTemperature4Alert1		
EngineOilLevelAlert		HydraulicReturnFilterDifferentialPressureAlert2	BatteryTemperature1Alert2		BatteryTemperature4Alert2		
EngineOilQualityAlert		HydraulicReturnFilterDifferentialPressureAlert3	BatteryTemperature2Alert1		BatteryCurrent1Alert1		
FuelTemperatureAlert		SelfRecoveryWinchHydraulicPressureAlert1	BatteryTemperature2Alert2		BatteryCurrent2Alert1		
FuelLevelAlert		SelfRecoveryWinchHydraulicPressureAlert2	BatteryTemperature3Alert1		BatteryCurrent3Alert1		
FuelFilterRestrictionAlert		SelfRecoveryWinchHydraulicPressureAlert3	BatteryTemperature3Alert2		BatteryCurrent4Alert1		
FuelFilterSeparatorAlert			BatteryTemperature4Alert1		BatteryPackCurrentAlert		

Fig. 10. Alert groupings.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
Tx	Rx	Description	PGN	SPN	Byte Offset	Bit Offset	Number of Bits	Units Per Bit	Units Offset	CAN Bus Units (SI)	Source Units	Scale	Translate	Units				
Mission Adaptable Control System HEMTT A2																		
Engine																		
Engine Speed																		
61444 190 3 0 16 1/8 0.0 RPM																		
61443 91 1 0 8 4/10 0.0 %																		
65262 110 0 0 8 1 -40.0 C																		
65262 175 2 0 16 3125/100000 -459.4 C																		
65263 100 3 0 8 4 0.0 kPa																		
65270 102 1 0 8 2 0.0 kPa																		
65270 105 2 0 8 2 -40.0 C																		
65280 260 0 0 8 1 0.0 0 - OK,1 - CHECK																		
65281 261 0 0 16 1 0.0 kPa																		
65282 262 0 0 8 1 0.0 0 - OK,1 - CHECK																		
65283 263 0 0 16 1 0.0 %																		
Fuel																		
Fuel Temperature																		
65262 174 1 0 8 1 -40.0 C																		
65266 183 0 0 16 1/20 0.0 L/H																		
65266 183 4 0 16 1/512 0.0 km/L																		
Fuel Level																		
65284 264 0 0 16 1 0.0 %																		
Fuel Filter Separator																		
65285 265 0 0 8 1/20 0.0 0 - OK,1 - CHECK																		
Fuel Filter Condition Switch																		
Transmission																		
Current Gear																		
61445 523 3 0 8 1 0.0 120-5R,121-4R,122-3R,123-2R,124-1R,125-N,126-1F,127-2F,128-3F,1																		
65265 976 6 0 5 1 0.0 0 - Off/Disabled, 01 - Hold, 02 - Remote Hold, 03 - Standby, 04 -																		
Transmission Oil Temperature																		
65270 177 4 0 16 1 0.0 0 - Disengaged, 01 - Engaged, 02 - Error, 03 - Not Available																		
Driveline Engaged																		
61442 560 0 0 2 1 0.0 RPM																		
Outset Shaft Speed																		
61442 191 1 0 16 1/8 0.0 RPM																		

Fig. 11. HEMTT data signals.

because the change from one state to another is not precisely defined. For example, fuel economy may be expanded to low, medium, or high, and a fuel consumption of 11 km/L may be considered medium or high depending on whether the vehicle is a truck or a car. These variables are subject to different interpretations by different experts at different points in time. Fuzzy logic allows the researcher to write control statements to accommodate this variability. For example: "IF gear ratio is high AND engine rpm is high, THEN fuel consumption is high." Following this method, fuzzy rules can replace mathematical rules, and decisions can more easily be made about how to use multi-sensor data fusion for optimal performance given the outputs of the sensors (Klein, 2004). The three basic elements of a fuzzy system are fuzzy sets, membership functions, and production rules. In this example, fuel economy is a fuzzy set with a membership function between 1 and 0, where

0 means that the variable is not part of a set and 1 means that it is completely part of a set. A variable may be a member of more than one set, and production rules in the form of "IF-THEN" logical statements represent human knowledge.

In the present study, the fuzzy Kalman filter approach (FKFA) was used to reduce the time needed to perform complex matrix manipulations to control higher order systems in the vehicle maintenance model. Moreover, the FKFA was able to capture the nonlinearity of engine operations under the influence of various anomalies. For instance, when an engine experiences component failures, the control system adjusts the actuator positions based on the sensor feedback measurements in order to meet its objective (e.g., to maintain engine speed at the desired value). Thus, the engine moves to a new operating condition, which may be a significant deviation from the nominal condition. Piecewise linear

Drivetrain		Wheel-Based Vehicle Speed		65265	84	1	0	16	1/256	0.0	KPH		621/1000	0	MPH
Brakes															
	Front Brake Wear		65301	268	0	0	8		1	0		0 - OK,1 - BAD			0 - OK,1 - BAD
	Rear Brake Wear		65302	468	0	0	8		1	0		0 - OK,1 - BAD			0 - OK,1 - BAD
Electrical															
	Alternator Voltage		65340	269	0	0	16		1/1000	0		V			V
	Alternator Temperature		65341	270	0	0	16		1/100	0		F			F
	Alternator Current		65342	271	0	0	16		1/100	0		A			A
	Starter Voltage		65343	272	0	0	16		1/1000	0		V			V
	Starter Temperature		65344	273	0	0	16		1/1000	0		F			F
	Battery Voltage #1		65320	274	0	0	16		1/1000	0		V			V
	Battery Voltage #2		65323	275	0	0	16		1/1000	0		V			V
	Battery Voltage #3		65330	276	0	0	16		1/1000	0		V			V
	Battery Voltage #4		65333	277	0	0	16		1/1000	0		V			V
	Battery Temperature #1		65321	278	0	0	16		1/100	0		F			F
	Battery Temperature #2		65324	279	0	0	16		1/100	0		F			F
	Battery Temperature #3		65331	280	0	0	16		1/100	0		F			F
	Battery Temperature #4		65334	281	0	0	16		1/100	0		F			F
	Battery Current #1		65322	282	0	0	16		1/100	0		A			A
	Battery Current #2		65325	283	0	0	16		1/100	0		A			A
	Battery Current #3		65332	284	0	0	16		1/100	0		A			A
	Battery Current #4		65335	285	0	0	16		1/100	0		A			A
	Ultracap Voltage		65400	286	0	0	16		1/1000	0		V			V
	Ultracap Temperature		65401	287	0	0	16		1/100	0		F			F
	Ultracap Current		65402	288	0	0	16		1/100	0		A			A
	Ignition Sense		65309	289	0	0	8		1	0		0 - OFF,1 - ON			0 - OFF,1 - ON

Fig. 12. HEMTT data signals continued.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	b	Signal Var Name																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																	
SPN	PGN	Bytes	Length	Signal Low	Signal High	Units	Id	tag	Can ID (hex)	Sub	Type	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15	Byte 16	Byte 17	Byte 18	Byte 19	Byte 20	Byte 21	Byte 22	Byte 23	Byte 24	Byte 25	Byte 26	Byte 27	Byte 28	Byte 29	Byte 30	Byte 31	Byte 32	Byte 33	Byte 34	Byte 35	Byte 36	Byte 37	Byte 38	Byte 39	Byte 40	Byte 41	Byte 42	Byte 43	Byte 44	Byte 45	Byte 46	Byte 47	Byte 48	Byte 49	Byte 50	Byte 51	Byte 52	Byte 53	Byte 54	Byte 55	Byte 56	Byte 57	Byte 58	Byte 59	Byte 60	Byte 61	Byte 62	Byte 63	Byte 64	Byte 65	Byte 66	Byte 67	Byte 68	Byte 69	Byte 70	Byte 71	Byte 72	Byte 73	Byte 74	Byte 75	Byte 76	Byte 77	Byte 78	Byte 79	Byte 80	Byte 81	Byte 82	Byte 83	Byte 84	Byte 85	Byte 86	Byte 87	Byte 88	Byte 89	Byte 90	Byte 91	Byte 92	Byte 93	Byte 94	Byte 95	Byte 96	Byte 97	Byte 98	Byte 99	Byte 100	Byte 101	Byte 102	Byte 103	Byte 104	Byte 105	Byte 106	Byte 107	Byte 108	Byte 109	Byte 110	Byte 111	Byte 112	Byte 113	Byte 114	Byte 115	Byte 116	Byte 117	Byte 118	Byte 119	Byte 120	Byte 121	Byte 122	Byte 123	Byte 124	Byte 125	Byte 126	Byte 127	Byte 128	Byte 129	Byte 130	Byte 131	Byte 132	Byte 133	Byte 134	Byte 135	Byte 136	Byte 137	Byte 138	Byte 139	Byte 140	Byte 141	Byte 142	Byte 143	Byte 144	Byte 145	Byte 146	Byte 147	Byte 148	Byte 149	Byte 150	Byte 151	Byte 152	Byte 153	Byte 154	Byte 155	Byte 156	Byte 157	Byte 158	Byte 159	Byte 160	Byte 161	Byte 162	Byte 163	Byte 164	Byte 165	Byte 166	Byte 167	Byte 168	Byte 169	Byte 170	Byte 171	Byte 172	Byte 173	Byte 174	Byte 175	Byte 176	Byte 177	Byte 178	Byte 179	Byte 180	Byte 181	Byte 182	Byte 183	Byte 184	Byte 185	Byte 186	Byte 187	Byte 188	Byte 189	Byte 190	Byte 191	Byte 192	Byte 193	Byte 194	Byte 195	Byte 196	Byte 197	Byte 198	Byte 199	Byte 200	Byte 201	Byte 202	Byte 203	Byte 204	Byte 205	Byte 206	Byte 207	Byte 208	Byte 209	Byte 210	Byte 211	Byte 212	Byte 213	Byte 214	Byte 215	Byte 216	Byte 217	Byte 218	Byte 219	Byte 220	Byte 221	Byte 222	Byte 223	Byte 224	Byte 225	Byte 226	Byte 227	Byte 228	Byte 229	Byte 230	Byte 231	Byte 232	Byte 233	Byte 234	Byte 235	Byte 236	Byte 237	Byte 238	Byte 239	Byte 240	Byte 241	Byte 242	Byte 243	Byte 244	Byte 245	Byte 246	Byte 247	Byte 248	Byte 249	Byte 250	Byte 251	Byte 252	Byte 253	Byte 254	Byte 255	Byte 256	Byte 257	Byte 258	Byte 259	Byte 260	Byte 261	Byte 262	Byte 263	Byte 264	Byte 265	Byte 266	Byte 267	Byte 268	Byte 269	Byte 270	Byte 271	Byte 272	Byte 273	Byte 274	Byte 275	Byte 276	Byte 277	Byte 278	Byte 279	Byte 280	Byte 281	Byte 282	Byte 283	Byte 284	Byte 285	Byte 286	Byte 287	Byte 288	Byte 289	Byte 290	Byte 291	Byte 292	Byte 293	Byte 294	Byte 295	Byte 296	Byte 297	Byte 298	Byte 299	Byte 300	Byte 301	Byte 302	Byte 303	Byte 304	Byte 305	Byte 306	Byte 307	Byte 308	Byte 309	Byte 310	Byte 311	Byte 312	Byte 313	Byte 314	Byte 315	Byte 316	Byte 317	Byte 318	Byte 319	Byte 320	Byte 321	Byte 322	Byte 323	Byte 324	Byte 325	Byte 326	Byte 327	Byte 328	Byte 329	Byte 330	Byte 331	Byte 332	Byte 333	Byte 334	Byte 335	Byte 336	Byte 337	Byte 338	Byte 339	Byte 340	Byte 341	Byte 342	Byte 343	Byte 344	Byte 345	Byte 346	Byte 347	Byte 348	Byte 349	Byte 350	Byte 351	Byte 352	Byte 353	Byte 354	Byte 355	Byte 356	Byte 357	Byte 358	Byte 359	Byte 360	Byte 361	Byte 362	Byte 363	Byte 364	Byte 365	Byte 366	Byte 367	Byte 368	Byte 369	Byte 370	Byte 371	Byte 372	Byte 373	Byte 374	Byte 375	Byte 376	Byte 377	Byte 378	Byte 379	Byte 380	Byte 381	Byte 382	Byte 383	Byte 384	Byte 385	Byte 386	Byte 387	Byte 388	Byte 389	Byte 390	Byte 391	Byte 392	Byte 393	Byte 394	Byte 395	Byte 396	Byte 397	Byte 398	Byte 399	Byte 400	Byte 401	Byte 402	Byte 403	Byte 404	Byte 405	Byte 406	Byte 407	Byte 408	Byte 409	Byte 410	Byte 411	Byte 412	Byte 413	Byte 414	Byte 415	Byte 416	Byte 417	Byte 418	Byte 419	Byte 420	Byte 421	Byte 422	Byte 423	Byte 424	Byte 425	Byte 426	Byte 427	Byte 428	Byte 429	Byte 430	Byte 431	Byte 432	Byte 433	Byte 434	Byte 435	Byte 436	Byte 437	Byte 438	Byte 439	Byte 440	Byte 441	Byte 442	Byte 443	Byte 444	Byte 445	Byte 446	Byte 447	Byte 448	Byte 449	Byte 450	Byte 451	Byte 452	Byte 453	Byte 454	Byte 455	Byte 456	Byte 457	Byte 458	Byte 459	Byte 460	Byte 461	Byte 462	Byte 463	Byte 464	Byte 465	Byte 466	Byte 467	Byte 468	Byte 469	Byte 470	Byte 471	Byte 472	Byte 473	Byte 474	Byte 475	Byte 476	Byte 477	Byte 478	Byte 479	Byte 480	Byte 481	Byte 482	Byte 483	Byte 484	Byte 485	Byte 486	Byte 487	Byte 488	Byte 489	Byte 490	Byte 491	Byte 492	Byte 493	Byte 494	Byte 495	Byte 496	Byte 497	Byte 498	Byte 499	Byte 500	Byte 501	Byte 502	Byte 503	Byte 504	Byte 505	Byte 506	Byte 507	Byte 508	Byte 509	Byte 510	Byte 511	Byte 512	Byte 513	Byte 514	Byte 515	Byte 516	Byte 517	Byte 518	Byte 519	Byte 520	Byte 521	Byte 522	Byte 523	Byte 524	Byte 525	Byte 526	Byte 527	Byte 528	Byte 529	Byte 530	Byte 531	Byte 532	Byte 533	Byte 534	Byte 535	Byte 536	Byte 537	Byte 538	Byte 539	Byte 540	Byte 541	Byte 542	Byte 543	Byte 544	Byte 545	Byte 546	Byte 547	Byte 548	Byte 549	Byte 550	Byte 551	Byte 552	Byte 553	Byte 554	Byte 555	Byte 556	Byte 557	Byte 558	Byte 559	Byte 560	Byte 561	Byte 562	Byte 563	Byte 564	Byte 565	Byte 566	Byte 567	Byte 568	Byte 569	Byte 570	Byte 571	Byte 572	Byte 573	Byte 574	Byte 575	Byte 576	Byte 577	Byte 578	Byte 579	Byte 580	Byte 581	Byte 582	Byte 583	Byte 584	Byte 585	Byte 586	Byte 587	Byte 588	Byte 589	Byte 590	Byte 591	Byte 592	Byte 593	Byte 594	Byte 595	Byte 596	Byte 597	Byte 598	Byte 599	Byte 600	Byte 601	Byte 602	Byte 603	Byte 604	Byte 605	Byte 606	Byte 607	Byte 608	Byte 609	Byte 610	Byte 611	Byte 612	Byte 613	Byte 614	Byte 615	Byte 616	Byte 617	Byte 618	Byte 619	Byte 620	Byte 621	Byte 622	Byte 623	Byte 624	Byte 625	Byte 626	Byte 627	Byte 628	Byte 629	Byte 630	Byte 631	Byte 632	Byte 633	Byte 634	Byte 635	Byte 636	Byte 637	Byte 638	Byte 639	Byte 640	Byte 641	Byte 642	Byte 643	Byte 644	Byte 645	Byte 646	Byte 647	Byte 648	Byte 649	Byte 650	Byte 651	Byte 652	Byte 653	Byte 654	Byte 655	Byte 656	Byte 657	Byte 658	Byte 659	Byte 660	Byte 661	Byte 662	Byte 663	Byte 664	Byte 665	Byte 666	Byte 667	Byte 668	Byte 669	Byte 670	Byte 671	Byte 672	Byte 673	Byte 674	Byte 675	Byte 676	Byte 677	Byte 678	Byte 679	Byte 680	Byte 681	Byte 682	Byte 683	Byte 684	Byte 685	Byte 686	Byte 687	Byte 688	Byte 689	Byte 690	Byte 691	Byte 692	Byte 693	Byte 694	Byte 695	Byte 696	Byte 697	Byte 698	Byte 699	Byte 700	Byte 701	Byte 702	Byte 703	Byte 704	Byte 705	Byte 706	Byte 707	Byte 708	Byte 709	Byte 710	Byte 711	Byte 712	Byte 713	Byte 714	Byte 715	Byte 716	Byte 717	Byte 718	Byte 719	Byte 720	Byte 721	Byte 722	Byte 723	Byte 724	Byte 725	Byte 726	Byte 727	Byte 728	Byte 729	Byte 730	Byte 731	Byte 732	Byte 733	Byte 734	Byte 735	Byte 736	Byte 737	Byte 738	Byte 739	Byte 740	Byte 741	Byte 742	Byte 743	Byte 744	Byte 745	Byte 746	Byte 747	Byte 748	Byte 749	Byte 750	Byte 751	Byte 752	Byte 753	Byte 754	Byte 755	Byte 756	Byte 757	Byte 758	Byte 759	Byte 760	Byte 761	Byte 762	Byte 763	Byte 764	Byte 765	Byte 766	Byte 767	Byte 768	Byte 769	Byte 770	Byte 771	Byte 772	Byte 773	Byte 774	Byte 775	Byte 776	Byte 777	Byte 778	Byte 779	Byte 780	Byte 781	Byte 782	Byte 783	Byte 784	Byte 785	Byte 786	Byte 787	Byte 788	Byte 789	Byte 790	Byte 791	Byte 792	Byte 793	Byte 794	Byte 795	Byte 796	Byte 797	Byte 798	Byte 799	Byte 800	

the vehicle health measure to calculate the average innovation vector \bar{Z}_k , which is then used in the fuzzy state correlator. The health risk measure is used to correlate new data with previously identified risky vehicle health data. The correlation is performed using the fuel rate difference and engine speed difference membership functions, as shown in Fig. 14.

An example of the production rules that determine whether a return i falls within the fuel rate and engine speed validation gate is the following: IF (fuel_rate_is_low) AND (engine_speed_is_high), THEN (degree_of_risk_is_medium). This scenario may occur if the vehicle is traveling at a moderate speed with little cargo. The complete set of production rules needed to associate new data with risk is shown in Table 2.

Once the data have been associated with previously identified vehicles, a risk membership function can be used to find subsequent values. The next step is to develop a fuzzy state correlator that will calculate the correction \mathbf{C}_k that will update the state estimate for the fuel rate and engine speed of the vehicle at time k as follows:

$$\hat{\chi}_{k/k} = \hat{\chi}_{k-1/k-1} + \mathbf{G}_k \mathbf{C}_k. \quad (3)$$

The fault detection of the subsystem is implemented by χ^2 detection combined with hardware redundancy. χ^2 detection is implemented as follows. Let $\gamma(k)$ be the residual of the system and $U(k)$ be the square difference of $\gamma(k)$. Then the fault function is defined as

$$\lambda(k) = \gamma^T(k) U^{-1}(k) \gamma(k), \quad (4)$$

and the rule is defined as

$$\lambda(k) > T_D - \text{Fault}, \quad (5)$$

$$\lambda(k) \leq T_D - \text{No Fault},$$

Table 2
Fuzzy association memory rules for degree of risky vehicle health.

Engine speed (rpm)	Fuel rate (L/h)		
	Low	Medium	High
Low	LOW	LOW	MEDIUM
Medium	LOW	MEDIUM	HIGH
High	MEDIUM	HIGH	HIGH

where $\lambda(k)$ refers to the χ^2 distribution (i.e., $\lambda(k) \sim \chi^2$ and T_D is determined by experience).

This method is used to detect faults in the subsystems. According to rule 1, if the subsystem made up of the gear and engine sensors fails, then the engine sensor may be at fault. If the subsystem made up of the fuel and electrical sensors fails, then the fuel sensor may be at fault. If the subsystem made up of the gear and electrical sensors fails and the subsystem made up of the gear and engine sensors fails, then the gear sensor may be at fault. If the subsystem made up of the gear and electrical sensors and the subsystem made up of the fuel and electrical sensors fails, then the electrical sensor may be at fault. If the three subsystems fail, then the dashboard may be at fault. The output of the fuel and electrical sensors is the input for fuzzy Kalman filter 1, the output of the gear and electrical sensors is the input for fuzzy Kalman filter 2, and the output of the gear and engine sensors is the input for fuzzy Kalman filter 3.

If a sensor fails, then the data should be discarded, the other sensors should be combined into a new configuration, and a new rule (rule 2) should be used to fuse the data. Rule 2 is as follows: If the engine sensor fails, then the gear and electrical, and the fuel and electrical, will make up the two subsystems separately in conjunction with the dashboard. If the fuel sensor fails, then the gear

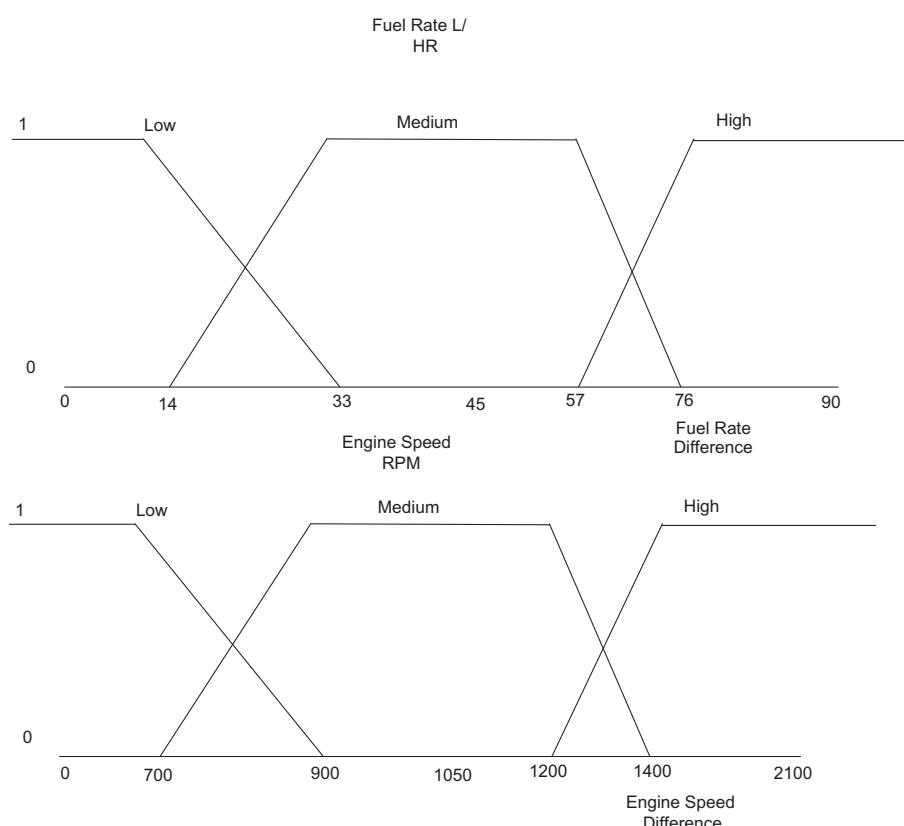


Fig. 14. Fuel rate difference and engine speed difference membership. L/HR = liters per hour; RPM = revolutions per minute.

and electrical, and the gear and engine, will make up the two subsystems separately in conjunction with the dashboard. If the gear sensor fails, then the fuel and engine, and the fuel and electrical, will make up the two subsystems separately in conjunction with the dashboard. If the electrical sensor fails, then the gear and engine, and the fuel and engine, will make up the two subsystems separately in conjunction with the dashboard. If the dashboard fails, then the gear and engine, and the fuel and electrical, will make up the two subsystems separately as the IVHMS. After the IVHMS is reconstructed, data from all of the subsystems are fused to obtain the IVHMS parameters, which are part of the fault-tolerant algorithm. The data fusion method uses dynamic detection and fault-tolerant feedback as follows. In the IVHMS, if the sensors do not fail, then the output Z of the system is

$$Z = \beta_1 Z_1 + \beta_2 Z_2 + \beta_3 Z_3, \quad (6)$$

where Z_1 , Z_2 , and Z_3 are the separate outputs of the three subsystems of the IVHMS before sensor failure and β_1 , β_2 , and β_3 are the weights of the separate subsystems where $\beta_1 + \beta_2 + \beta_3 = 1$ according to the information conservation principle. If any subsystem is at fault at any time, then the output Z of the system will be reconstructed as follows:

$$Z = \alpha_1 Z_{n1} + \alpha_2 Z_{n2} + \cdots + \alpha_i Z_{ni}, \quad (7)$$

where Z_{n1} , Z_{n2} , ..., Z_{ni} are the separate outputs of all of the subsystems after reconstruction and $\alpha_1, \alpha_2, \dots, \alpha_l = 1$ because of the information conservation principle. The weights are related to the precision, reliability, and usability of each subsystem under normal conditions and are determined by the optimal weight distribution principle. For simplicity and based on reasonability, we assume that $\beta_1 = \beta_2 = \beta_3 = 1/3$ and that $\alpha_1 = \alpha_{n2} = \cdots = \alpha_l = 1/l$, such that the weights will be adjusted if the system fails.

A critical aspect in designing a fuzzy logic Kalman filter making it robust to off-nominal engine behavior caused by degradation (Kobayashi, Simon, & Litt, 2005; Lu, Chen, & Hamilton, 2000). Component degradation is a natural phenomenon that occurs with all engines as a result of normal usage. The level of component degradation worsens gradually with time, and consequently engine performance begins to deviate from the nominal level. Component faults also result in off-nominal engine performance, but these occur abruptly because of anomalies such as damage from foreign or domestic objects, such as sand and dirt. As discussed earlier, sensors and actuators may also exhibit anomalous behavior due to bias and drift. If a filter is not robust to component degradation and anomalies, incorrect estimation may lead mission-critical elements (e.g., the control system, the maintenance crew) to take

incorrect action. The general linear Kalman filter involves a nonlinear engine model that is linearized at various operating points. The nonlinear engine model is represented by the following equations:

$$\begin{aligned} x &= f(x, h, u_{cmd}, e), \\ y &= g_y(x, h, u_{cmd}, e) + v, \\ z &= g_z(x, h, u_{cmd}, e), \end{aligned} \quad (8)$$

where x , h , u , and e represent the vectors of state variables, vehicle health parameters, command and control inputs, and environmental parameters, respectively. A health parameter such as revolutions per minute indicates the “health” of the engine component. For any given input value, the nonlinear functions f , g_y , and g_z generate the vectors of state derivatives x , sensor outputs y , and non-measurable performance parameters z , respectively. Sensor output is varied via vector noise v . The following state-space equations are obtained by linearizing the engine model:

$$\begin{aligned} x &= A(x - x_{ss}) + L(h - h_{ref}) + B(u_{cmd} - u_{ss}), \\ y - y_{ss} &= C_y(x - x_{ss}) + M(h - h_{ref}) + D_y(u_{cmd} - u_{ss}) + v, \\ z - z_{ss} &= C_z(x - x_{ss}) + N(h - h_{ref}) + D_z(u_{cmd} - u_{ss}), \end{aligned} \quad (9)$$

where A , B , C_y , C_z , D_y , D_z , L , M , and N are the state-space matrices with the appropriate dimensions. The subscript ss refers to the steady-state point at which the engine model is linearized, and the vector h_{ref} refers to a reference health condition. Thus, as the engine model deviates from normal operating conditions such as optimal revolutions per minute, these health parameter variations cause the state derivatives, sensor outputs, and performance parameters to deviate from their normal condition values. Therefore, the estimation capability of the FKFA adapts to engine health parameter deviations by estimating the health parameter vector in addition to the state variable vector. If all of the health parameters can be accurately estimated, then the state variables, sensor outputs, and performance parameters can also be accurately estimated regardless of the deviations that occur in the engine health. However, this assumes that the number of available sensors must be at least equal to the number of engine health parameters in order to estimate all of the health parameters. The FKFA algorithm is used to accurately estimate and correct the engine output despite deviations in the health parameter sensors.

Our experiments revolve around two parameters: fuel consumption in liters per hour (0–90) and engine speed in revolutions per minute (0–2100). We assume that the vehicle electrical potential is 13.93 V, fuel consumption is 4 L/h, engine speed is 1200 rpm, and the gear ratio is 8.93%. The data are simulated, and for simplicity we assume that errors include only random noise.

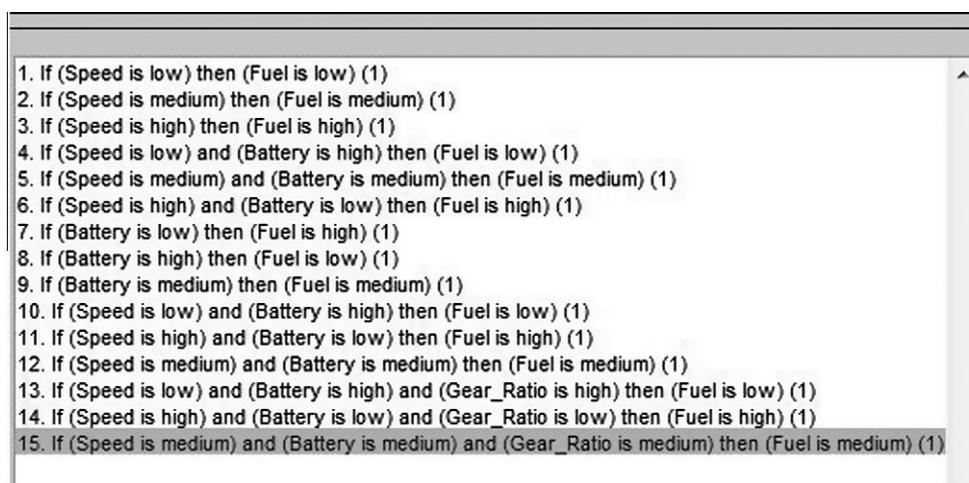


Fig. 15. Rules generated.

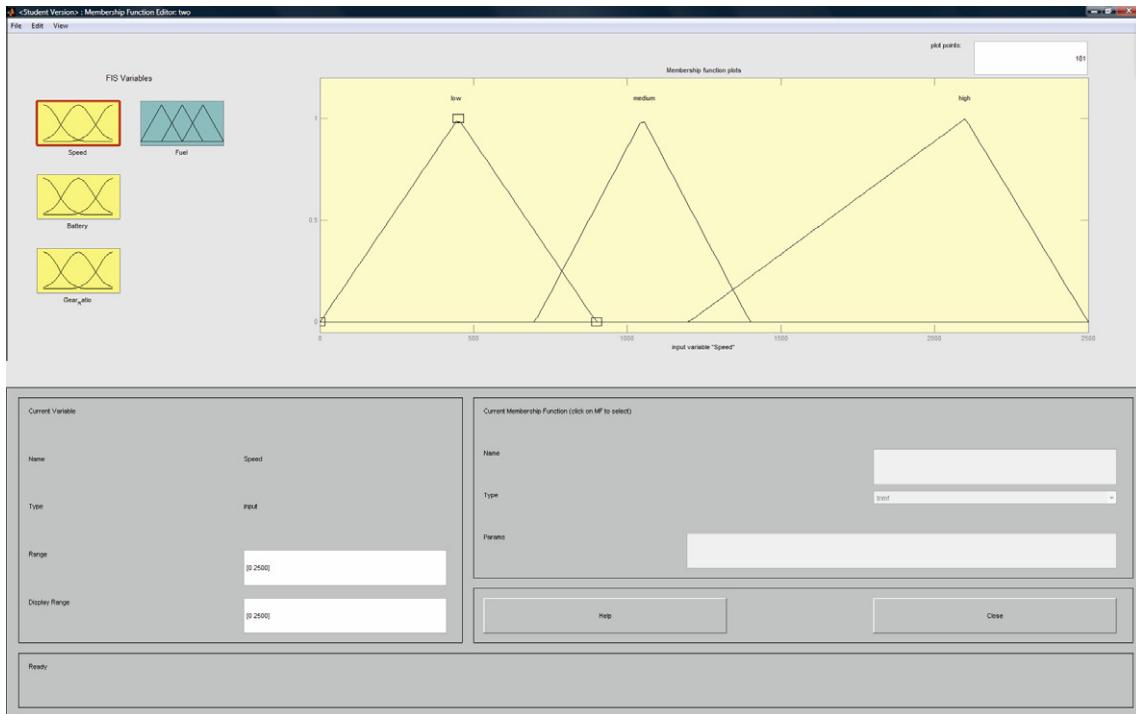


Fig. 16. Speed input.

The simulated faults included in our experiments are block fault, constant gain fault, and constant bias fault, which are described as follows. Let $y_{i\text{out}}$ be the actual output of the i th sensor and y_{in} be the normal output when the i th sensor is working normally. The block fault mode of the i th sensor is

$$y_{i\text{out}}(t) = \alpha_i, \quad (10)$$

where α_i is a constant. The constant gain fault mode of the i th sensor is

$$y_{i\text{out}}(t) = \alpha_i \beta_i y_{in}(t), \quad (11)$$

where β_i is the proportional coefficient of the change constant. The constant bias fault mode of the i th sensor is

$$y_{i\text{out}}(t) = y_{in}(t) + \Delta_i, \quad (12)$$

where Δ_i is a constant. It is assumed that the faults are simulated and that the amplitude is random.

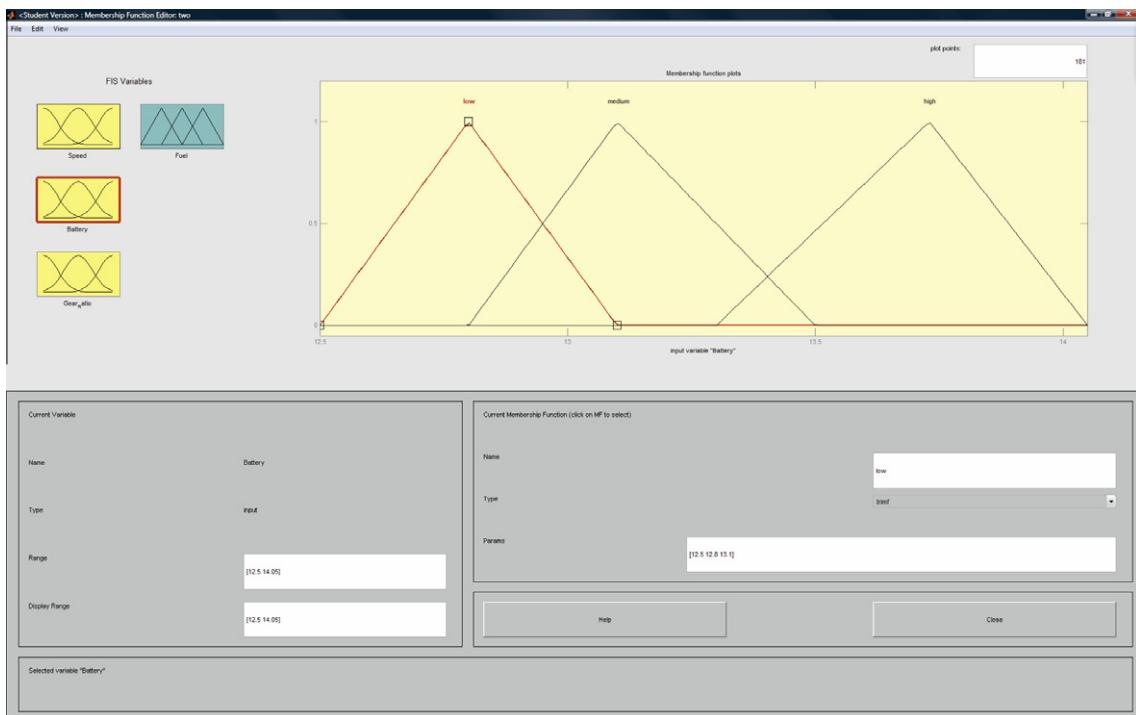


Fig. 17. Battery input.

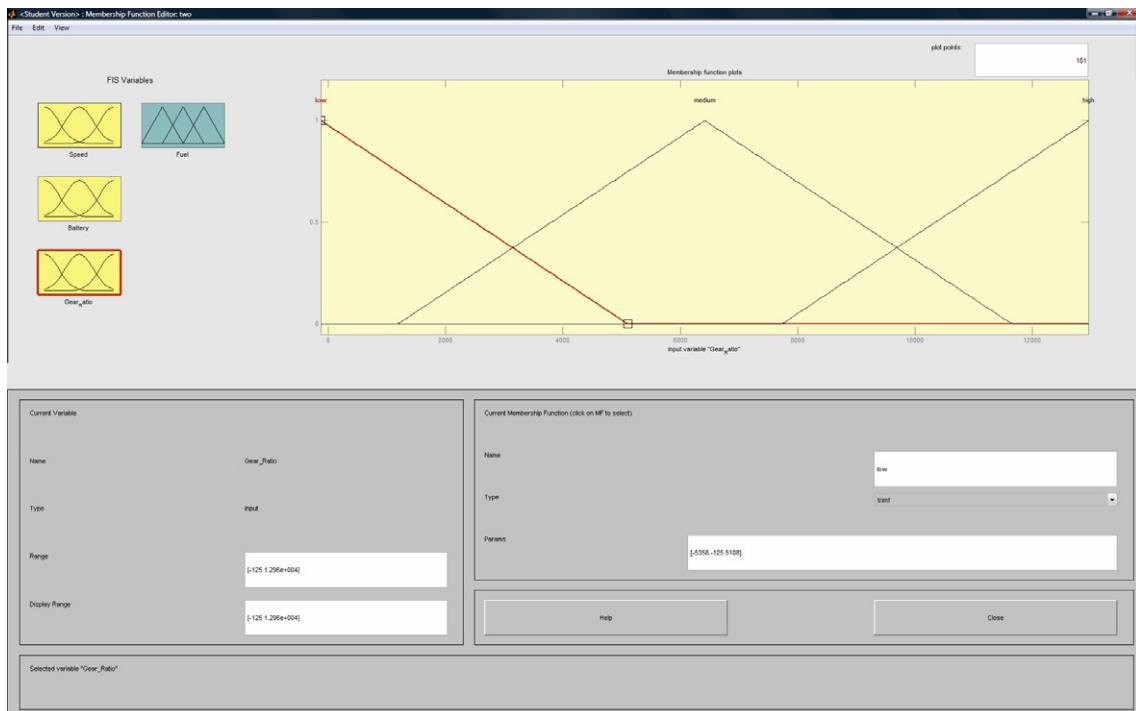


Fig. 18. Gear ratio input.

4.3. Development of the rules

The following figures show the rules input and output membership functions (MFs) are illustrated in Figs. 15–19. Fig. 15 shows the 15 rules generated. Figs. 16–18 show the inputs of speed, battery and gear ratio. Fig. 19 provides the fuel consumption output.

4.4. Output control surfaces

Fig. 20 demonstrates the 2D surface map for engine speed and fuel while Fig. 21 illustrates the interaction of rpm = 400, voltage = 14.05, gear = 1400 and fuel output of 16.0218.

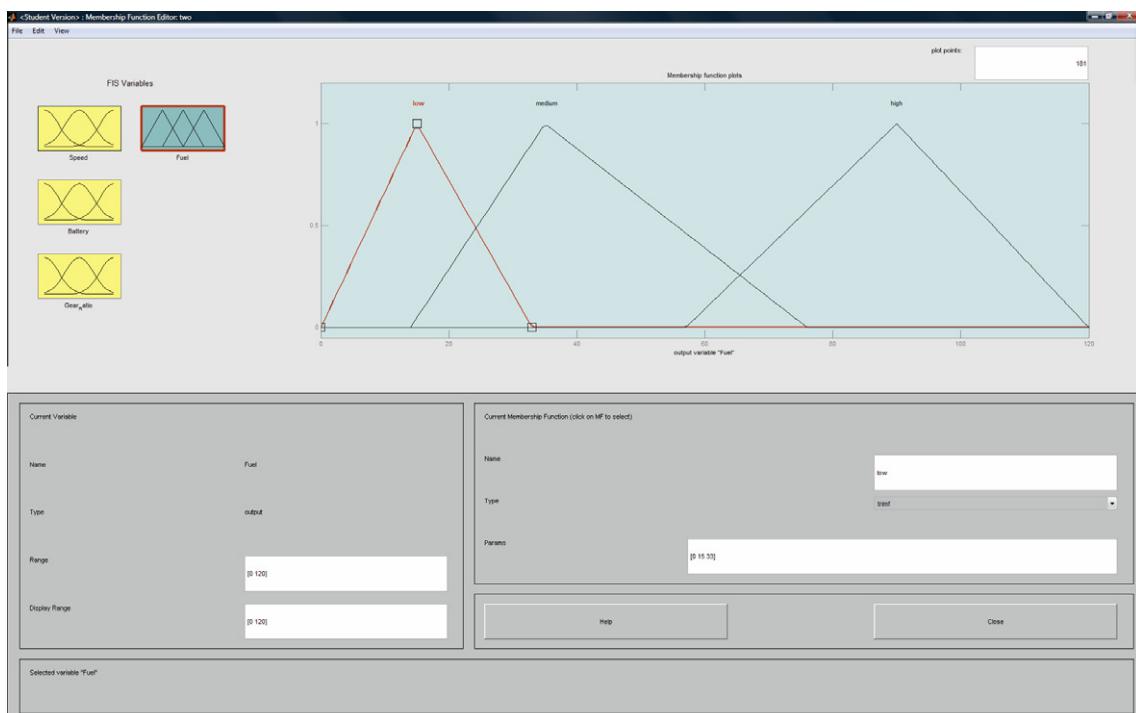


Fig. 19. Fuel consumption output.

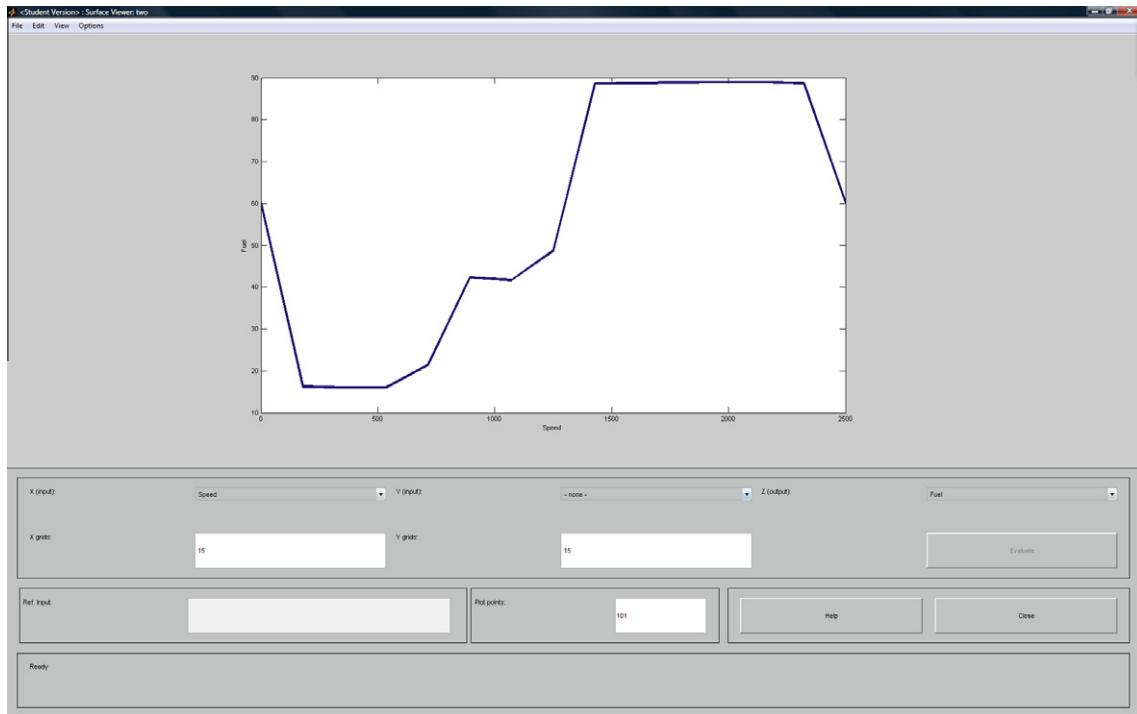


Fig. 20. Surface map 2D engine speed and fuel.

5. FDI integration and experimental methodology

5.1. FDI integration into IVHMS

Fault detection and isolation (FDI) is a subfield of control engineering concerned with monitoring a system, identifying when a fault has occurred, and pinpointing the type of fault and its location. Two approaches can be distinguished. The first is direct pattern recognition of sensor readings that indicate a fault. The second analyzes the discrepancy between the sensor readings

and expected values derived from some model. In the latter case, a fault is detected if the discrepancy or residual exceeds a certain threshold. It is then the task of fault isolation to categorize the type of fault and its location in the machinery.

5.2. Experimental methodology

FDI techniques can be broadly classified into two categories: model-based FDI and signal processing-based FDI. The FDI model may be a mathematical or knowledge-based model of the system.

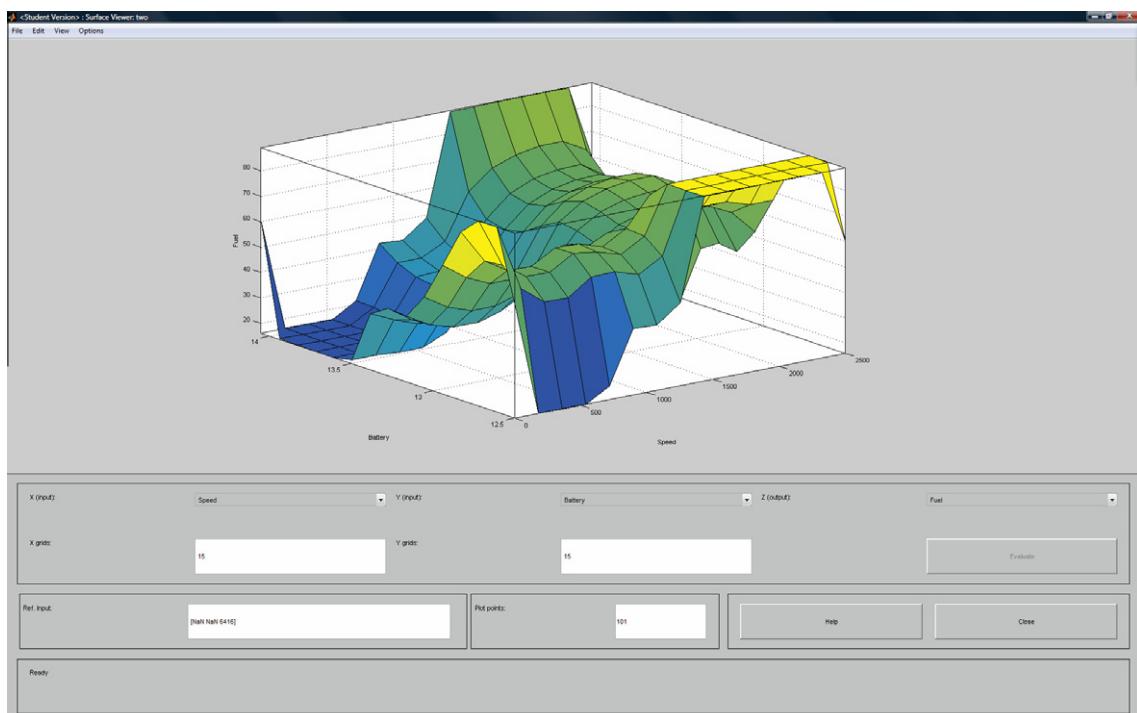


Fig. 21. Interaction of rpm, voltage, gear and fuel.

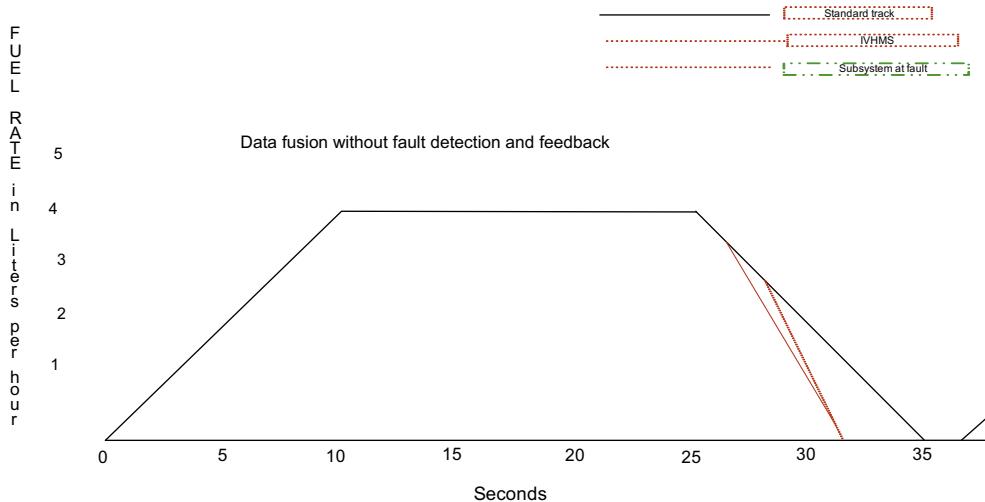


Fig. 22. Data fusion without fault detection and feedback.

In model-based FDI, a model of the system is used to determine the occurrence of fault (Wang & Liang, 2008). Some model-based FDI techniques include the observer-based approach, the parity-space approach, and parameter identification-based methods. One example of a model-based FDI technique is the truth table and state chart used for an aircraft elevator reactive controller. The truth table defines how the controller reacts to detected faults, and the state chart defines how the controller switches between the different modes of operation of each actuator (passive, active, standby, off, and isolated). For example, if a fault is detected in the hydraulic system, then the truth table sends an event to the state chart that the left inner actuator should be turned off. One of the benefits of this model-based FDI technique is that this reactive controller can also be connected to a continuous-time model of the actuator hydraulics, allowing for the study of switching transients. In signal processing-based FDI, some mathematical or statistical operations are performed on the measurements, or some neural network is trained to use measurements to extract information about the fault. A good example of signal processing-based FDI is time domain reflectometry, in which a signal is sent down a cable or electrical line and the reflected signal is compared mathematically to the original signal to identify any faults. For

instance, spread spectrum time domain reflectometry involves sending a spread spectrum signal down a wire line to detect wire faults.

6. Experimental results and discussions

6.1. Test results

We used FDI to improve inefficiencies in our engine model. An IVHMS with four sensors and a dashboard was designed and a fuzzy Kalman filter applied. The IVHMS includes three subsystems: the gear and engine, the fuel and electrical, and the gear and electrical. By using fault-tolerant algorithms, the IVHMS can determine whether there is a faulty sensor.

In the first example, for the constant bias fault of the fuel, we verify that the fault toleration method is valid by performing the following experiment. First we define the three parameters F_i , F_o , and F_c . We determine that if the subsystem made up of the fuel and electrical sensors is at fault, then $F_i = 1$. If the subsystem made up of the gear and electrical sensors is at fault, then $F_o = 1$. If the subsystem made up of the gear and engine sensors is at fault, then $F_c = 1$. From the 25th to the 35th seconds, the fuel sensor is at the

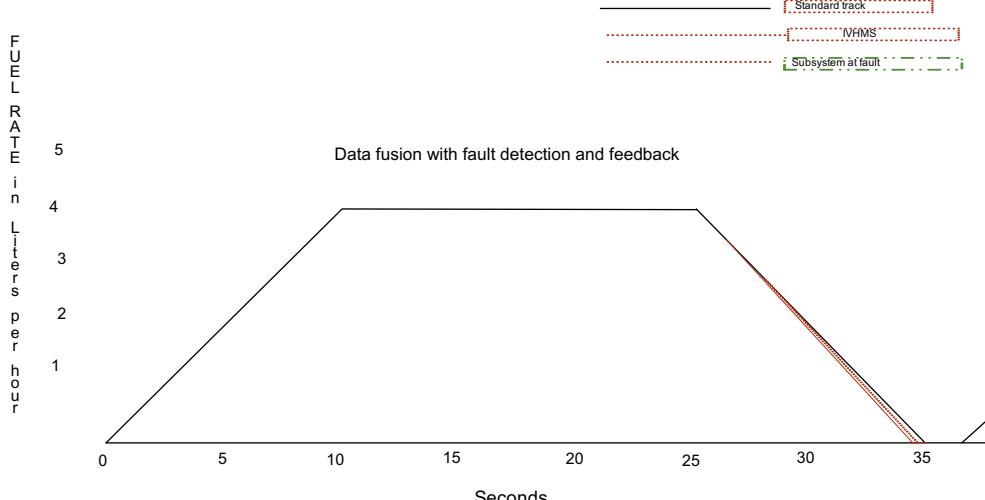


Fig. 23. Data fusion with fault detection and feedback.

constant fault bias of 4 L/h. F_i , F_o , and F_c are 1, 1, and 0, respectively, and in accord with rule 1 the fault of the fuel sensor is identified exactly as predicted. A comparison is then made between the outputs of the IVHMS subsystems with and without faults. If the IVHMS does not tolerate the fault when it is at the constant fault bias of the fuel sensor, then the outputs of the IVHMS and the subsystem with the fault are adjusted accordingly. The fuel adjustments are illustrated in Figs. 22 and 23.

7. Conclusions, discussion, and future research

In this paper, we explored a new multi-sensor data fusion method based on fault detection and FKFA feedback for an IVHMS. Kalman filters were applied to a dashboard to create an IVHMS model of a real engine whose performance will deviate from the nominal baseline over its lifetime of operation. The FKFA was designed to adjust its performance as the engine subsystems adjusted. The FKFA attributed any component performance deviations due to degradation and/or faults to combined engine subsystems to accurately estimate off-nominal vehicle health.

The FKFA was evaluated using a fuzzy logic simulation of a large commercial engine model. The stability and estimation accuracy of the FKFA were evaluated at multiple operating points using component degradation, actuator biases, and component faults. The FKFA exhibited excellent stability. Despite the fact that the FKFA was designed at specific operating points, the algorithm was stable throughout the vehicle maintenance profile. The FKFA was also able to estimate performance parameters at off-nominal health conditions with fairly good accuracy. Parameters such as fuel consumption and engine speed were estimated with high accuracy and subsystem failure was compensated for and adjusted to the standard track. These results demonstrate that multi-sensor data fusion based on fault detection and fuzzy Kalman feedback is an effective method of reducing risk in an IVHMS.

We have also demonstrated that cluster analysis can be used to find natural groupings of data. Specifically, factor analysis was used to assign multi-sensor data to clusters based on factor loadings. Following the five basic steps required for a cluster algorithm, we selected the sample data, defined a set of variables to measure, computed the similarities among the entities, used factor analysis to create similar groups, and validated the resulting cluster solution. This process allowed for the separation of data into identifiable groups that fit logically with our research model.

The natural groupings consisted of four clusters: (1) gear, consisting of current gear (.942), output shaft speed in revolutions per minute (.942), selected gear (.939), wheel-based vehicle speed in miles per hour (.943), and an inverse relationship with actual gear ratio (-.605); (2) engine, consisting of accelerator pedal position (.860), boost pressure (.851), driver's demand engine (percent torque) measured in miles (.886), and engine speed in revolutions per minute (.681); (3) fuel economy, consisting of fuel rate in liters per hour (.719), speed (.877), actual engine (percent torque) (.793), and an inverse relationship with instantaneous fuel economy in kilometers per liter (-.836); and (4) electrical potential (.945). Input shaft speed (.671) loaded on both the gear and engine factors. These results were consistent with the major multi-sensor data infusion inputs from the drive train, engine, carburetor, and electrical systems.

Furthermore, we developed a risk membership function that could be used to find subsequent optimal engine speed and fuel economy. We demonstrated that the IVHMS and the subsystem at fault were corrected and greatly improved through the use of fault-tolerant methods and that an updated IVHMS could be constructed. The IVHMS was able to use simulation and verification to identify the sensors without fault and fuse their data to improve system performance. It can therefore be concluded that multi-sensor

data fusion based on fault detection and feedback is an effective method of reducing risk in an IVHMS.

Finally, in Fig. 21, we tested our fuzzy logic filter and compared the results to actual outputs from the IVHMS. We found that a RPM of 400, 14.05 V and a gear ratio of 15,000 lead to a 16.0218 L/h fuel consumption. Similarly 1200 rpm with 13.2 V and 7000 gear ratio lead to 41.9993 L/h consumption. 2400 rpm with 12.4 V and 125 gear ratio lead to 88.667 L/h rate of fuel consumed.

References

- Ammar, S., Wright, R., & Selden, S. (2000). Ranking state financial management: A multilevel fuzzy rule-based system. *Decision Sciences*, 31, 449–481.
- Bhaskar, T., Pal, M., & Pal, A. (2011). A heuristic method for RCPSP with fuzzy activity times. *European Journal of Operational Research*, 208, 57–66.
- Dong, W., & Pentland, A. (2006). Multi-sensor data fusion using the influence model. In *International workshop on wearable and implantable body sensor networks* (pp. 72–75). Washington, DC: IEEE Computer Society.
- Fay, D., Ivey, R., Bomberger, N., & Waxman, A. (2003). Multisensor and spectral image fusion and mining: From neural systems to applications. In *32nd Applied imagery pattern recognition workshop* (p. 11). Washington, DC: IEEE Computer Society.
- Ghahroudi, M., & Fasih, A. (2007). A hybrid method in driver and multisensor data fusion, using a fuzzy logic supervisor for vehicle intelligence. In *2007 International conference on sensor technologies and applications* (pp. 393–398). Valencia, Spain: IEEE Xplore.
- Hall, D. L. (1992). *Mathematical techniques in multisensor data fusion*. Norwood, MA: Artech House.
- Kao, C., & Lin, P. (2011). Qualitative factors in data envelopment analysis: A fuzzy number approach. *European Journal of Operational Research*, 211, 586–593.
- Klein, L. A. (2004). *Sensor and data fusion: A tool for information assessment and decision making*. Bellingham, WA: SPIE Press.
- Kobayashi, T., Simon, D., & Litt, J. (2005). *Application of a constant gain extended Kalman filter for in-flight estimation of aircraft engine performance parameters*. Turbo Expo, American Society of Mechanical Engineers, Reno, Nevada, June 6–9.
- Kong, Y., & Guo, S. (2006). Urban traffic controller using fuzzy neural network and multisensors data fusion. In *IEEE international conference on information acquisition* (pp. 404–409). Veihai, China: IEEE Explore.
- Liu, L., Wang, W., & Cui, J. (2009). A new scheme for multisensor image fusion system. *Fifth International Conference on Information Assurance and Security*, 2, 509–512.
- Lu, Y., Chen, T. Q., & Hamilton, B. (2000). A fuzzy system for automotive fault diagnosis: Fast rule generation and self-tuning. *IEEE Transactions on Vehicular Technology*, 49, 651–660.
- Minor, C., Johnson, K., Rose-Pehrsson, S., Owrutsky, J., Wales, S., Steinhurst, D., et al. (2007). Data fusion with a multisensor system for damage control and situational awareness. In *2007 IEEE conference on advanced video and signal based surveillance* (pp. 313–317). London: IEEE Computer Society.
- Niu, W., Zhu, J., Gu, W., Chu, J. (2009). Four statistical approaches for multisensor data fusion under non-Gaussian noise. In *2009 IITA international conference on control, automation and systems engineering* (pp. 27–30). Zhangjiajie, China: IEEE Computer Society.
- Skinner, H. (1979). Dimensions and clusters: A hybrid approach to classification. *Applied Psychological Measurement*, 3, 327–341.
- Smith, D., & Singh, S. (2006). Approaches to multisensor data fusion in target tracking: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 18, 1696–1710.
- Storms, P., van Velen, J., & Boasson, E. (2005). A process distribution approach for multisensor data fusion systems based on geographical dataspace partitioning. *IEEE Transactions on Parallel and Distributed Systems*, 16, 14–23.
- Sun, H., Wang, W., Cao, Y., He, S., & Yan, X. (2009). Application of fuzzy data fusion in multi-sensor environment monitor, in. *International Conference on Computational Intelligence and Security*, 2, 550–553.
- Sun, H., Zhu, H., & Wu, T. (2009). RBF network based feature-level data fusion for robotic multi-sensor gripper. *International Joint Conference on Computational Sciences and Optimization*, 1, 719–722.
- Tan, L. C. B., & Hsieh, P. J. (2005). Application of the fuzzy weighted average in strategic portfolio management. *Decision Sciences*, 36, 489–511.
- Tian, J., Gao, M., & Li, K. (2006). The detection system of oil tube defect based on multisensor data fusion by classify support vector machine. *First International Conference on Innovative Computing, Information and Control*, 3, 182–185.
- Uttam, K., Chiranjit, M., & Ramachandra, T. (2009). Fusion of multisensor data: Review and comparative analysis. *WRI Global Congress on Intelligent Systems*, 2, 418–422.
- Wang, J., & Liang, K. (2008). Multi-sensor data fusion based on fault detection and feedback for integrated navigation systems. In *2008 International symposium on intelligent information technology application workshops* (pp. 232–235). Shanghai, China: IEEE Computer Society.
- Wang, H., Liu, X., Lai, J., & Liang, Y. (2007). Network security situation awareness based on heterogeneous multi-sensor data fusion and neural network. In *Second international multi-symposiums on computer and computational sciences* (pp. 352–359). Iowa City, IA: University of Iowa.

- Wang, J., Zhu, W., Si, G., & Zang, Y. (2008). Feature extraction in Ball mill pulverizing system based on multisensory data fusion. *CSSE '08 Proceedings of the International Conference on Computer Science and Software Engineering*, 1, 573–576.
- Wei, Z., Ma, Z., Yang, H., & Liu, M. (2006). Research on the temperature compensation of optical fiber displacement sensor based on multisensor data fusion. *Sixth International Conference on Intelligent Systems Design and Applications*, 2, 562–566.
- Widener, P., Schwan, K., & Bustamante, F. (2003). Differential data protection for dynamic distributed applications. In *19th Annual computer security applications conference* (pp. 396–405). Las Vegas, NV: IEEE Computer Society.
- Wu, X., Liu, S., & Sharma, D. (2007). The application of multi-sensors fusion in vehicle transmission system fault diagnosis. *Third International Conference on Natural Computation*, 2, 728–731.
- Xuejun, L., Guangfu, B., & Dhillon, B. (2009). A new method of multi-sensor vibration signals data fusion based on correlation function. *WRI World Congress on Computer Science and Information Engineering*, 6, 170–174.
- Yager, R. (1981). Concepts, theory, and techniques: A new methodology for ordinal multiobjective decisions based on fuzzy sets. *Decision Sciences*, 12, 589–600.
- Yang, L., Cheng, Y., Wei, H., & Lu, J. (2006). Error analysis of multi-sensor data fusion system for target detection on the ocean surface. In *2006 IEEE international conference on information acquisition* (pp. 415–419). Veihai, China.
- Zhang, M., Song, H.Lv.S., Li, Y., Yu, X., & Bao, J. (2009). Research on the multi-sensors information fusion technique based on the neural networks and its application. In *2009 Second international workshop on knowledge discovery and data mining* (pp. 93–96). Moscow, Russia: IEEE Computer Society.
- Zhang, C., & Wang, H. (2010). Decentralized multi-sensor data fusion algorithm using information filter. *International Conference on Measuring Technology and Mechatronics Automation*, 1(2010), 890–893.
- Zhou, Y., & Leung, H. (1997) Minimum entropy approach for multisensor data fusion. In *1997 IEEE signal processing workshop on higher-order statistics* (p. 336). Washington, DC: IEEE Computer Society.
- Zia, A., DeBrunner, V., Chinnaswamy, A., & DeBrunner, L. (2002). Multi-resolution and multi-sensor data fusion for remote sensing in detecting air pollution. In *Fifth IEEE southwest symposium on image analysis and interpretation* (p. 9). Washington, DC: IEEE Computer Society.