

An Integrated Diagnostic System for Marine Diesel Engines

Jason Burkholder and Christopher Wiles

Barron Associates, Inc.

1410 Sachem Place, Suite 202, Charlottesville, VA, USA, 22901

*Ryan Roecker

Southwest Research Institute

6220 Culebra Road, San Antonio, TX, USA, 78238

Key Words: Diagnostics, Prognostics, Fault Detection

ABSTRACT

Many military and commercial entities are currently funding the development of Unmanned Surface Vehicles (USVs) that could potentially perform vital functions, such as harbor security patrols and mine reconnaissance. A primary obstacle that must be overcome in order for USVs to provide valuable service is the lack of maintenance personnel on board the vessel. It is essential to develop an automated system for marine diesel engines that performs diagnostics and prognostics, *viz.*, an integrated diagnostic system (IDS) for marine diesel engines. Barron Associates and Southwest Research Institute acquired a marine diesel engine, installed the engine in a test facility, paired the engine with a computer-controlled dynamometer, and instrumented the engine with an extensive sensor suite and data acquisition system. The sensor suite included thermocouples, pressure transducers, accelerometers, and several automotive-type sensors. The research team constructed models of normal engine behavior and developed algorithms to reliably diagnose and prognose a wide range of realistic seeded faults.

INTRODUCTION

Barron Associates and Southwest Research Institute (SwRI) conducted a research effort that included the development of a marine diesel engine dynamic gas path model, the application and demonstration of fault detection, isolation, and prognostication algorithms for the gas path actuator and sensors, and the presentation of algorithms for monitoring the engine mechanical hardware. The research team acquired a marine diesel engine, installed the engine in a test facility at SwRI, paired the engine with a computer-controlled dynamometer, and instrumented the engine with an extensive sensor suite and data acquisition system. The sensor suite included thermocouples, pressure transducers, accelerometers, and several automotive-type sensors. The research team constructed models of normal engine behavior with the engine operating along a realistic “propeller curve” in the torque vs. speed plane. The gas path models, based on first principles of combustion thermodynamics, accurately predict engine behavior, including air flow, fuel flow, compressor performance, turbocharger performance, and cooling system performance. The mechanical models capture the vibration signatures correlated with the known firing order.

The research team analyzed the engine data and designed and tested the health monitoring algorithms using MATLAB and Simulink. The algorithms were then implemented in a deployable, ruggedized engine control unit (ECU) for real-time, online testing with the diesel engine. The real-time prototype reliably detected

representative failures without false alarms. The false alarm rate and missed detection rate were estimated analytically based on many hours of engine operating data. A successful “virtual” sea trial was conducted wherein the dynamometer was controlled to impart a seaway-like load disturbance on the engine.

Although developed for marine diesel engines used for propulsion, the underlying IDMS technology is readily applied to both marine and terrestrial diesel engines for vehicle propulsion, electrical power generation, and other applications. Work is ongoing applying IDS technologies to land-based diesel generator sets. A conceptual block diagram of the IDS is shown in Figure 1.

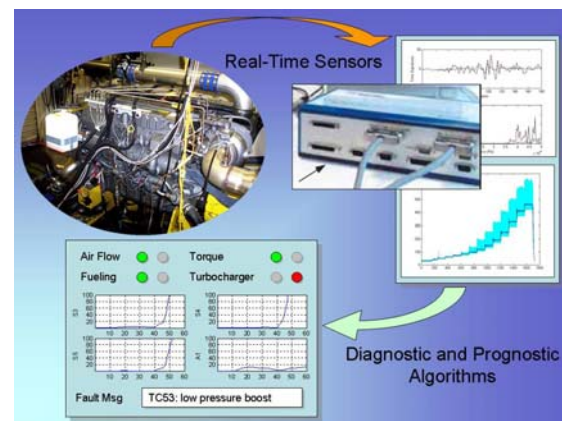


Figure 1: IDS Engine Monitoring Concept

EXPERIMENTAL SETUP

The target engine was a Yanmar Marine 6LY2A-STP. SwRI mounted the engine in a test cell and installed all components to complete the auxiliary systems necessary for engine operation. On-highway DF-2 diesel fuel and weight 10W40 oil were used. An eddy-current dynamometer was connected to the engine crankshaft to provide variable loading of the engine. A means to accurately control the throttle of the engine was implemented using a Woodward throttle actuator. This implementation provided drive-by-wire capability for accurate computer-based control.

The engine was instrumented with additional non-stock sensors and integrated with a high-performance data acquisition system. Instrumentation included pressure transducers, thermocouples, accelerometers and additional automotive-type sensors. Six pressure transducers were acquired and mounted on the engine to collect pressure data. The instrumented engine is shown in Figure 2.

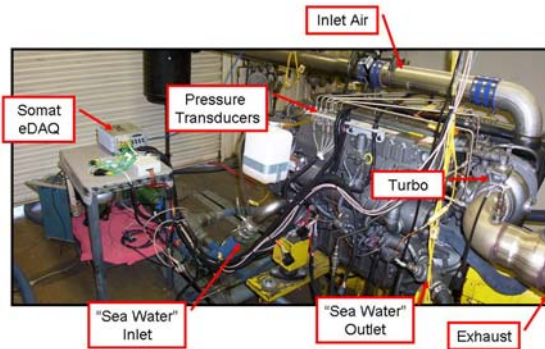


Figure 2: Engine with Instrumentation

The engine was operated in two primary modes for data acquisition to capture both steady-state operation and transient dynamics for construction of the engine model required by the diagnostic and prognostic system. The first mode of operation is the “Propeller Curve” mode. The propeller curve (a.k.a. prop curve) of an engine is defined by the speed and torque of the engine’s maximum output operating point. Using this point, an exponential curve is defined that is commonly used to select a propeller for an engine/ship combination. The second test mode is the “Six Mode Test”, wherein the engine is moved automatically between six steady-state operating points at fixed times. The operating points encompass a wide range of operating speeds and loads.

Engine Model Design

A mean-value approach is used to model the engine, where the steady-state operation of the engine is considered [1,2]. Thus, the effects of events that occur on a timescale of less than one engine cycle are lumped together and modeled on a per-cycle basis. Data were collected while the engine was operated at a variety of speeds and loads, encompassing the engine’s normal operating regime. Using the known model structure, model parameters capturing the important features of the

engine subsystems were estimated. Figure 3 shows the modeled engine subsystems and their interconnections.

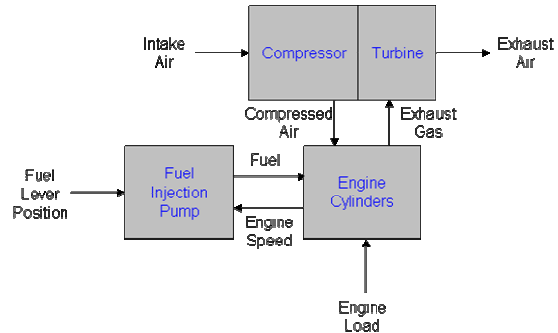


Figure 3: Engine Model Subsystems

The engine cylinder subsystem models the operation of all six engine cylinders lumped together as a single combustion chamber, with the effects of the combustions averaged over one engine cycle (two revolutions of the crankshaft). The engine cylinders accept air from the intake manifold through the intake valves and fuel from the injection pump through the fuel injector nozzles. The combustions produce torque on the crankshaft, exhaust gases, and waste heat. The products of combustion are ejected in gaseous form through the exhaust valves into the exhaust manifold. The waste heat from combustion exits by convection in the exhaust gases and also is conducted through the cylinder walls and engine block to the engine oil and cooling water.

The mass flow rate of air into the cylinders is calculated from other measured quantities by considering the operation of the engine cylinders as a pumping device. Under ideal conditions, the mass of air entering the engine cylinders per cycle is given by the engine displacement volume and the operating conditions which influence the density of the air entering the cylinders. The per-cycle air mass is converted to a mass flow rate by multiplying by the engine speed and a conversion factor representing the operation of one engine cycle over two engine revolutions. The primary factors producing a deviation from ideal conditions, the finite intake valve orifice area and the finite opening time of the intake valve, are captured with a multiplier, denoted “volumetric efficiency,” applied to the ideal mass air flow rate.

The ratio of air mass to fuel mass in the cylinders at the time of combustion is calculated using the chemical equation for combustion and performing a mass balance of the reactants and products of the combustion chemical reaction. Assuming a complete combustion of the fuel and negligible nitrogen oxide formation, the amount of oxygen remaining in the combustion products is inversely related to the amount of fuel combusted. Using the law of partial pressures for ideal gases, a measurement of the exhaust oxygen concentration is used to determine the mass ratio of air to fuel present prior to combustion. The combustion chemical equation expresses the number of moles of air available to the reaction as the sum of the amount needed to completely

oxidize the fuel (a_{th}) and the excess air available (a_{ex}).

In addition, the number of moles of water vapor present in the air is expressed as (v). The number of moles of oxygen remaining in the exhaust is equal to the total moles of gas in the exhaust times the percent of gas that is oxygen ($O_{2,meas}$). The two pertinent equations follow.

$$C_n H_m + (a_{th} + a_{ex})(O_2 + 3.76N_2) + vH_2O \rightarrow nCO_2 + \left(\frac{m}{2}\right)H_2O + a_{ex}O_2 + (a_{th} + a_{ex})3.76N_2$$

$$a_{ex} = O_{2,meas} \left(n + \frac{m}{2} + a_{ex} + 3.76(a_{th} + a_{ex}) + v \right)$$

The fuel supply subsystem models the combined effects of the fuel filter, fuel pump, governor, fuel injector lines, and injector nozzles. The amount of fuel delivered per cycle is dictated by the pressure developed by the high pressure plunger and the length of time high pressure fuel is available to the delivery valve. The governor assembly uses the control lever position and engine speed to control the position of the control sleeve, which determines the end of fuel delivery timing by depressurizing the fuel in the high pressure plunger. The timing device adjusts the cycle timing of the beginning of fuel delivery based on the engine speed and pressure of the fuel developed in the low pressure cavity by the feed pump and regulating valve. The fuel flow rate delivered by the injection pump is modeled using a second order polynomial in two inputs:

$$\dot{m}_f = \alpha_1 + \alpha_2 \cdot l + \alpha_3 \cdot N + \alpha_4 \cdot l^2 + \alpha_5 \cdot N^2$$

which expresses the mass fuel flow rate, \dot{m}_f , as a

function of engine speed, N , and control lever position, l .

The turbocompressor subsystem uses some of the residual heat energy of the exhaust gases to increase the pressure of the intake manifold to provide greater air mass flow to the engine cylinders. The expansion of the exhaust gases drives a turbine, which mechanically drives a compressor that boosts the pressure of the air in the intake system, thus increasing the air density. The compression of the intake air is modeled as an isentropic process, with the efficiency calculated as the enthalpy gain of the gas not due to compression.

The combination of these subsystem models in a single model for the entire engine yields the following results comparing values predicted by the model to values calculated from engine measurements. Note that measured values approach modeled values in steady-state operation due to the low bandwidth of some sensors (e.g. temperature thermocouples).

Additional engine modeling was conducted using three sets of accelerometer measurements to characterize the unique engine vibration caused by each cylinder firing. The modeling block uses the known firing sequence of the engine to find the unique vibration waveforms due to each cylinder firing. The firing sequence for this engine is 1-5-3-6-2-4.

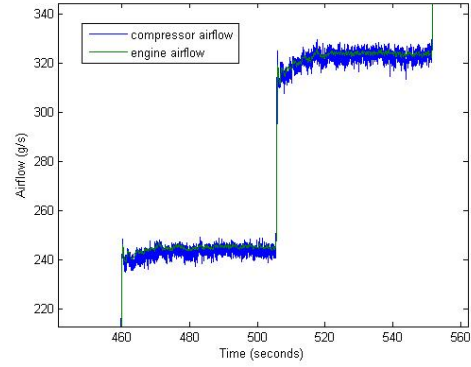


Figure 4: Measured and Predicted Airflow

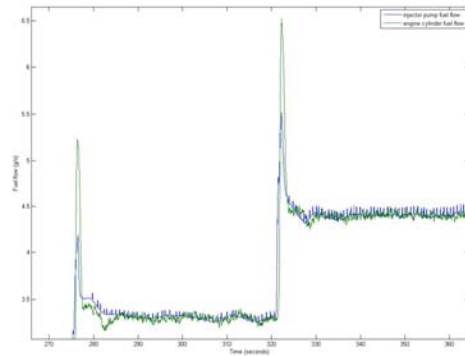


Figure 5: Measured and Predicted Fuel Flow

The accelerometer data contain both high-frequency and low-frequency content. Prior to each firing, there is a brief high-frequency signal. A high-pass filter is applied to each set of accelerometer data and the high-pass data are used to find a complete set of six firings (i.e., one for each cylinder). This is done by searching the middle section of the high-pass data according to the known firing order (i.e., searching for one and two in the 'front' array; searching for three and four in the 'middle' array; searching for five and six in the 'back' array) to determine the most likely sequence placement. A second set of data sequences is constructed by applying a low-pass filter to the accelerometer data. Segments of low-pass data are taken around the times of the six firings (that were found by processing the high-pass data). These segments are used as the characteristic waveforms for each of the six firings. The algorithm then steps from the middle section forward through the low-pass data, according to the known firing sequence, and searches over a window of candidate firing times to find the most likely matching waveform. For example, when the algorithm is searching for a waveform for a cylinder five firing, it uses an average of the cylinder one, five, and three firings from the previous cycle then adds the estimated cycle duration to yield the base search index. The algorithm correlates the characteristic waveform for cylinder five with the low-pass 'back' data at different leads and lags about the base search index

and picks the best waveform match. This process is repeated for each cylinder according to the firing order, and then it is repeated working from the middle section backward through the data. The result is a set of vibration waveforms corresponding to each cylinder firing.

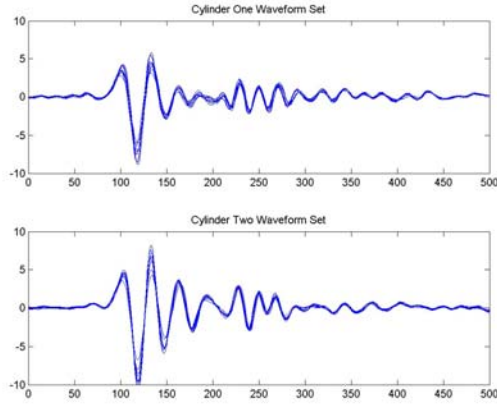


Figure 5: Overlaid Vibration Waveforms; Cylinders 1&2

Diagnostic Algorithms

The diagnostic algorithms are designed to produce results which quantify the agreement of the engine model to the data collected by the sensors and parameters calculated from these data. These FDI observer “residuals” are examined for departures from their baselines, which may be indicative of a fault. The residuals are analyzed statistically for changes, resulting in decision statistics signifying the current health of the engine with respect to the model. Making the assumption of a decorrelated Gaussian residual sequence readily lends itself to a straightforward multivariate extension of the standard generalized likelihood ratio (GLR) formulation for a univariate Gaussian sequence susceptible to an unknown mean shift [3]. The multivariate GLR extension yields a single detection statistic for each FDI observer. The GLR test is sensitive to faults of unknown onset time and unknown severity. The window-limited GLR is suitable for online, real-time implementation.

Diagnostic Performance

The distributions of the decision statistics are analyzed in the unfaulted cases to determine an optimum threshold that will yield a satisfactorily small false-alarm rate. The threshold is then applied to the different faulted cases to determine the probability of detection and evaluate the overall performance of the SCD algorithm. For example, consider the detection statistics arising from the airflow model for the unfaulted case shown in Figure 6. The small false alarm rate is distributed over the wide range of engine operating states.

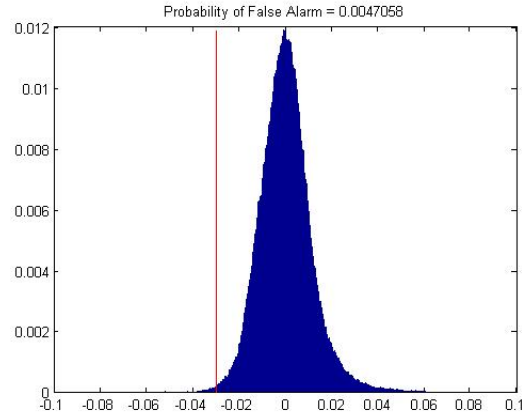


Figure 6: Airflow Model Detection Statistics (No Fault)

The airflow model reliably detects faults in the air delivery path, such as intake manifold leaks and air inlet restrictions. The research team induced intake manifold leaks of varying severity. The probability of detection for four test cases is shown in Figure 7.

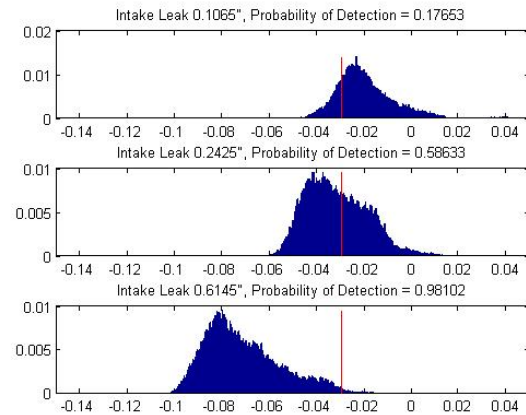


Figure 7: Airflow Detection Statistics (Faulted)

The fuel flow decision statistic is formed by the difference of the measured fuel flow through the engine cylinders, calculated by analysis of the gases in the exhaust manifold, and the expected fuel flow of the fuel supply system, calculated by using the fuel injector flow characteristics. An error caused by greater fuel flow through the fuel supply system than through the engine cylinders, such as a fuel leak or fuel actuator failure, is indicated by an increase in the mean of this decision statistic. An error caused by less fuel flow through the fuel injectors than expected based on the fuel delivery characteristics is indicated by a negative shift in the decision statistic mean. Figures 8 and 9 show the distributions of the fuel flow decision statistic for unfaulted, steady-state cases and for a fuel leak fault, respectively. The probability of detection of the fuel flow leak fault was 99.8 % at a false alarm probability of less than 0.8 %.

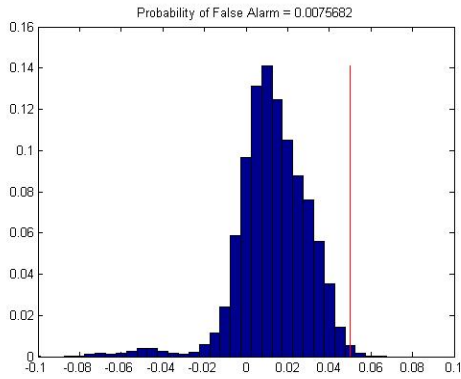


Figure 8: Fuel Flow Detection Statistics (No Fault)

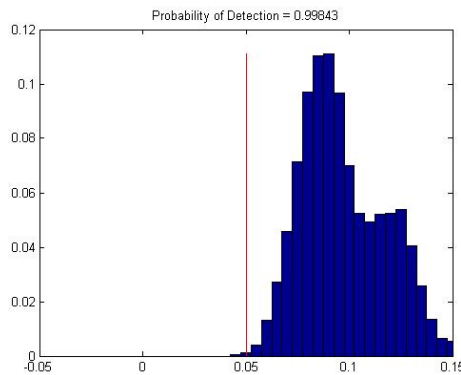


Figure 9: Fuel Flow Detection Statistics (Fuel Leak)

Within the accelerometer modeling block, a simple diagnostic test is implemented to assess whether one or more cylinders is not producing the expected output. The algorithm constructs the sets of waveforms for each cylinder and uses them to build a set of detection measures that are compared to a threshold. Consider the case of n waveforms found for cylinder k . The detection measure for cylinder k is constructed by first determining the standard deviation for each waveform, yielding n power (neglecting the mean) measures. The power estimate for cylinder k is then determined by taking the mean of the n standard deviation estimates. At this point, there are six power estimates, one for each cylinder. The detection measure is determined by dividing each power estimate by the mean (taken over all six cylinders) power. If the power estimates are equal, then the detection measure for each cylinder would equal a value of one. The detection measures are compared to a pre-set threshold. If a measure is less than the threshold, then a corresponding fault flag is set to one and is output from the modeling block; otherwise, all the output flags remain zero. In performing the fault analysis, waveforms were compared in time-domain, in frequency-domain, and according to variations in the firing intervals.

For detection thresholds of 0.5 and below, there are no false alarms and the accelerometer modeling block correctly detects the fuel leak fault, identifying the fault

in cylinder three. No other faults are detected. The algorithm's detection threshold is set as 0.3 to allow some margin in the implementation. As more data are made available (and a wider range of faults is considered), the detection threshold should be adjusted appropriately. Figure 10 shows the distributions of each cylinder's decision statistic during unfaulted engine operation. Figure 11 shows the distributions of each cylinder's decision statistic where a fuel flow fault has caused cylinder 3 to produce less power than normal due to lean misfires.

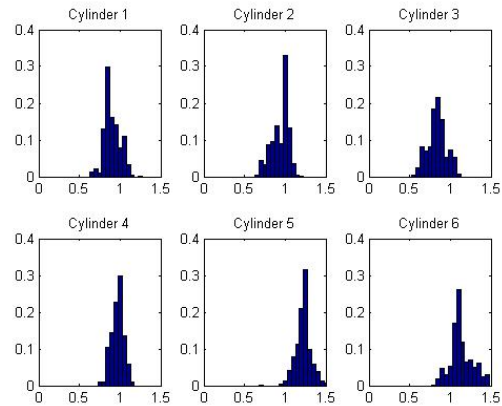


Figure 10: Accelerometer Detection Statistics (Unfaulted)

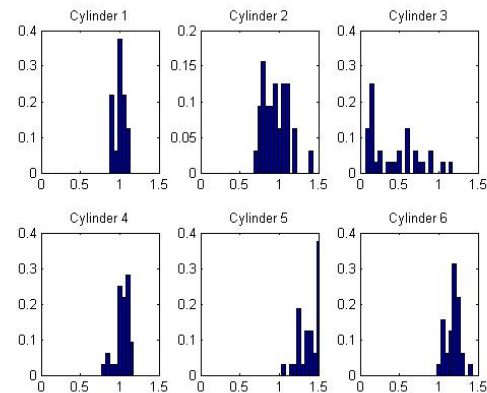


Figure 11: Accelerometer Detection Statistics (Fuel Leak)

Prognostic Algorithms and Performance

The detection statistics generated by the diagnostic algorithms can be used in prognostic algorithms to determine the Remaining Useful Life (RUL) of either the component or the system as a whole. Barron Associates developed an RUL prediction algorithm based on polynomial fitting of fault detection statistics. The algorithm was applied to fault detection statistics for the air intake leak and its performance was assessed across all of the faulted and unfaulted data collected.

Barron Associates' prognostics method provides fault prognostication that is superior to simple usage

monitoring and far less costly and much more practical in most cases than trending or model-based approaches, which require failure progression models that are often expensive and impractical to obtain. Although usage or trending information can be integrated into the prognostics module if it is available, it is not required.

Barron Associates' fault detection and isolation techniques use statistical change detection methods that provide a single detection statistic for each FDI observer and a threshold is set such that a detection statistic exceeding the threshold is taken to be indicative of a fault. The end of useful life will be defined as the time a detection statistic first exceeds the fault detection threshold. This definition is reasonable in the case of sensor failures as this is the point after which the failed sensor input should not be used by the control system or monitoring system (i.e., the end of the sensor's useful life). In the case of actuator failures, it is the point at which the control or monitoring system should schedule maintenance and, possibly, limit the actuation range (i.e., the end of the actuator's useful life). In the case of plant failures, such as a mechanical failure, this definition is meaningful in that it is the point at which maintenance is required prior to the next mission. Pursuing this interpretation, prognostication of RUL is possible. Based on the critical threshold and the detection statistic history, a time-series model to predict RUL (i.e., time until the detection statistic crosses the threshold) may be constructed.

The RUL prediction algorithm developed under this effort takes a one-dimensional series of fault detection statistics (y) as an input. The algorithm operates on a sliding data window of length M . In practice, the algorithm simply maintains a storage array of the previous M samples $Y = [y(0), y(1), \dots, y(M-1)]^T$. At each time, the prediction algorithm determines the best N th-order polynomial fit, in the least-square sense, for the data array over an arbitrary basis $X = [x(0), x(1), \dots, x(M-1)]^T$. The polynomial fit is then used to extrapolate the data over some pre-defined look-ahead window. The predicted data is compared to the fault threshold to determine the RUL. A polynomial fit is a simple predictive technique that performs well for the fault detection statistics generated under this effort. Another candidate predictive algorithm that was considered was the Wiener filter, which has somewhat greater computational complexity.

Because the RUL prediction algorithm was developed to run as part of a real-time application, the implementation of the polynomial fit algorithm needed to be computationally efficient, minimizing the number of computations required at each time step. Fortunately, the least-squares solution can be subdivided such that the bulk of the computations are performed on initialization and the number of calculations that must be performed on each iteration is minimal. Data are input to the algorithm at a static sampling interval and stored in a first-in, first-out storage array of length M . Since the length of the data window and the spacing of the samples are fixed, an arbitrary basis (i.e., time axis) may be

selected for use at every iteration of the algorithm.

In order to assess the performance of the RUL prediction model, the algorithm was applied over all of the applicable faulted and unfaulted data collected, using the fault detection statistics for the Airflow Model (i.e., in this case for the detection of the air intake leak). The model was constructed to be applied on an engine operating on the prop curve at steady-state, for mid-range engine speeds (between 850 rpm and 2900 rpm). The prognostic subsystem is intended for application to slow onset failures. Parameters such as the sampling rate and the size and resolution of the look-ahead window are defined by the user, providing the user the freedom to adjust the model for the particular application.

Figure 12 shows the prognostic results for an example air leak fault case. The top plot shows the fault detection statistics input to the RUL prediction subsystem. The fault threshold is shown as a dashed line. The middle plot shows the estimated RUL. The RUL estimate is not calculated until the RUL falls within the 5-second look-ahead window. The bottom plot shows the estimated fault time, which is simply calculated from the RUL estimate. It can be seen that the RUL is correctly and promptly (once within the look-ahead window) estimated.

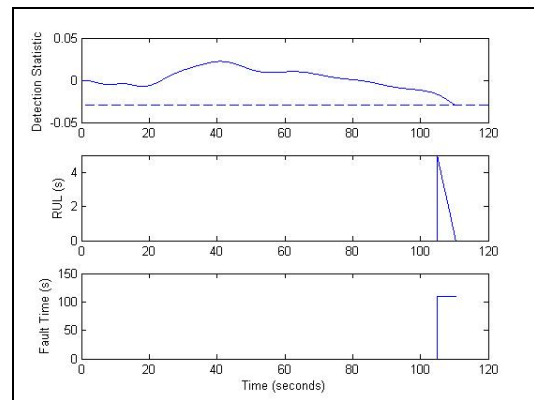


Figure 12: Prognostic Results (Intake Manifold Leak)

The RUL prediction subsystem was also applied to a case with a gradual onset air inlet restriction. In this case, the look-ahead window was extended to 10 seconds (note that these are accelerated onset faults – the air inlet restriction could, in practice, be due to a gradually clogging air filter). Figure 13 shows the prognostic results for the example air inlet restriction fault case. The impending fault is accurately and promptly predicted.

In summary, over this data set, the RUL prediction subsystem performed well. Larger air intake leaks are detected without false alarms. In practice, the prognostic algorithm will be applied to gradual onset failures, with data collected over longer periods at slower rates, as opposed to the discrete failures and brief data captures inherent to the data set built and used in this effort. A more expansive data set will be needed to fully assess the prediction and false alarm performance of the prognostic algorithm.

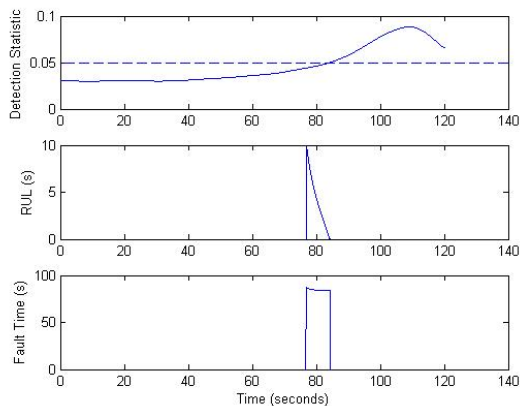


Figure 13: Prognostic Results (Air Inlet Restriction)

Prototype Implementation and Demonstration

Barron Associates has implemented the finalized diagnostic and prognostic algorithms in a deployable, ruggedized engine control unit (ECU). The ECU is able to receive accelerometer data and other sensor values (e.g., temperatures and pressures) from signal conditioning hardware and calculates the diagnostic and prognostic quantities necessary for determining engine health.

The diagnostic and prognostic modules prototyped with Simulink and Stateflow move seamlessly from development into production using code generated from Simulink models by Real-Time Workshop [4]. The Simulink blockset and real-time compiler allow the health monitoring algorithms, finalized in off-line testing in the MATLAB and Simulink environment, to be downloaded to the ECU using a straightforward programming procedure. This ability reduces development time and facilitates debugging and troubleshooting. The chosen implementation platform was the dSpace MicroAutobox [5].

Using The Mathworks's Real-Time Workshop and Embedded Coder, code is produced that is structured to be called by a timer interrupt in a real-time operating system. As a proof-of-concept demonstration of code deployment, a Windows DLL was created from the diagnostic models using Real-Time Workshop. An application was created which invoked function calls into the DLL to execute the algorithms in near-real-time on a desktop PC. This demonstration allowed for the refinement of the code generation parameters and specification of the test and debug variables necessary for verification of the hardware implementation. Figure 14 shows a screen capture of the application output.

A simple GUI for test and debugging the MicroAutobox implementation was built using the dSPACE ControlDesk software to display signals from the ECU in real-time. ControlDesk supports numerical displays, dial gauges, plots, indicator lamps, and other intuitive display controls. A display of the sensor data was prepared, including inputs, outputs, and intermediate variables. Variables are also logged to an external PC in

MATLAB data file format. Figure 15 shows a screen capture of the MicroAutobox display. At the point in testing when this photograph was taken, a fuel flow fault has been detected.

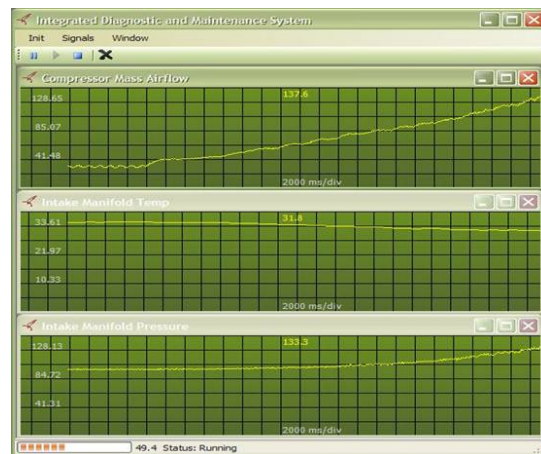


Figure 14: Windows DLL Screenshot

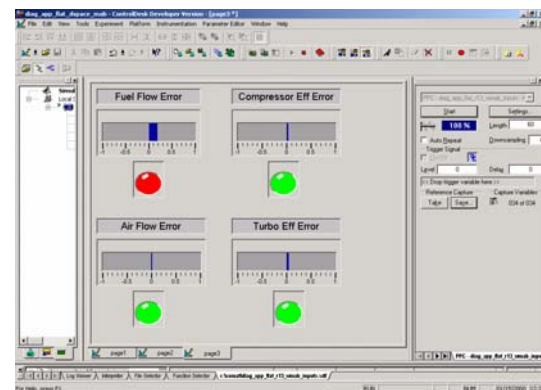


Figure 15: MicroAutobox Display Screenshot

Several tests were successfully conducted to verify real-time diagnostic and prognostic performance. In all cases, the prototype IDS performed as predicted in the off-line tests used to develop the algorithms.

Laboratory “Sea Trial” Test

It is clear that a diesel-powered marine propulsion system will be subject to seaway-induced load variations when underway. The culmination of the research effort was a set of experiments designed to emulate “at-sea” conditions in a laboratory environment. The dynamometer employed in the laboratory at SwRI allows the programming of time-varying loads that follow a pre-defined relationship (e.g. propeller curve relationship) and can be user-defined. This ability provides a mechanism for inducing load disturbances on the engine in the laboratory which are characteristic of actual disturbances that could be encountered at sea.

Generation of realistic wave forces is subject to the operating parameters of the vessel. Likely areas of deployment and weather conditions under which it will

be deployed impact the wave models used to generate random wave patterns. The dimensions of the vehicle, preferred encounter angle with respect to the waves, and hydrodynamic properties of the hull and propeller determine the translation of the wave disturbances into an actual load on the engine.

Waves are created by the interaction of persistent wind across the sea and hydrostatic and gravitational forces on the water. The fetch, or distance across the sea which the wind is blowing, wind speed, commonly measured at a standard height above the water, and time over which the wind has been blowing affect the statistics of the amplitude and frequency of the waves encountered on the water. The wave spectrum describes the distribution of wave amplitudes as a function of wave frequency. Several well-known formulations of wave spectra are the Pierson-Moskowitz, Modified Pierson-Moskowitz, Bretschneider, Ochi, and Joint North Sea Wave Project (JONSWAP) spectra. Many of these include parameters that account for the observed wave height, wind speed, and the development stage of the sea in time due to the length of time the wind has been blowing. Only the JONSWAP spectrum addresses the parameter of fetch, while the other models assume unlimited fetch – a condition only valid on the open ocean. It was deemed prudent to assume a finite fetch and use the JONSWAP model to generate the wave spectrum. In order to translate the wave force into loading torque on the engine, a representative ship's hull and propeller hydrodynamics were approximated [6,7].

A sea state of 3 was used as a representative case to determine wind speed and wave height. A fetch of 10 nautical miles was chosen as representative of littoral waters. According to the World Meteorological Organization, a sea state code of 3 corresponds to wave heights classified as "slight," with observed wave heights ranging from 0.5 to 1.25 meters, while a Beaufort number of 3 corresponds to a gentle breeze, with wind speeds of 8 to 11 knots.

The sum of the seaway-induced load disturbance time sequence and the nominal load from the propeller load curve was presented to the dynamometer. The dynamometer was stimulated at 1 second intervals by the automated laboratory configuration, while a constant speed command was presented to the engine. A 15 minute test was performed, corresponding to the file size limit for signal capture.

The distributions of the diagnostics subsystem detection statistics during the test all correctly result in no errors detected despite the wave-like disturbance on the engine load. The prognostics algorithm also does not produce a RUL estimate, as the statistics do not indicate a tendency toward gradual failure. The results of this simulated sea trial demonstrate the robustness of the diagnostic and prognostic algorithms, and indicate a high likelihood of success for a fielded system.

CONCLUSIONS

The research team constructed models of normal engine behavior with the engine operating along a realistic "propeller curve" in the torque vs. speed plane. The gas path models, based on first principles of combustion thermodynamics, accurately predict engine behavior, including air flow, fuel flow, compressor performance, turbocharger performance, and cooling system performance. In addition, a wide range of combustion degradation and asymmetries can be detected using vibration data.

The research team analyzed the engine data and designed and tested the health monitoring algorithms using MATLAB and Simulink. The algorithms were then implemented in a deployable, ruggedized engine control unit (ECU) for real-time, online testing with the diesel engine. The real-time prototype reliably detected representative failures without false alarms. The false alarm rate and missed detection rate were estimated analytically based on many hours of engine operating data. A successful "virtual" sea trial was conducted wherein the load dynamometer was controlled to impart a seaway-like load disturbance on the engine.

Although developed for marine diesel engines used for propulsion, the underlying technology is readily applied to both marine and terrestrial diesel engines for vehicle propulsion, electrical power generation, and other applications.

Acknowledgements

The authors thank the U.S. Office of Naval Research for supporting this work.

REFERENCES

- [1] Heywood, J., *Internal Combustion Engine Fundamentals*. New York: McGraw-Hill, 1988.
- [2] Guzzella, L. and C. Onder, *Introduction to Modeling and Control of Internal Combustion Engine Systems*. Berlin: Springer, 2004.
- [3] Lai, T.L., "Efficient recursive algorithms for detection of abrupt changes in signals and control systems", *IEEE Trans. Information Theory*, Vol. 45, No. 5, 1999, pp. 952-965.
- [4] www.mathworks.com
- [5] www.dspaceinc.com/ww/en/inc/home/products
- [6] Ghose, J. and R. Gokarn, *Basic Ship Propulsion*. New Dehli: Allied, 2004.
- [7] Bertram, V., *Practical Ship Hydrodynamics*. Oxford: Butterworth-Heinemann, 2000.