Proceedings of SPIE, Sensor Fusion: Architecture, Algorithms, and Applications III, Vol. 3719, 7-9 April '99,, pp. 52-61, 1999

Fuzzy sensor fusion for gas turbine power plants

Kai Goebel[*] [a] and Alice Agogino[b]

[a]GE Corporate Research & Development, K1-5C4A, Niskayuna, NY 12309
[b]Department of Mechanical Engineering, University of California at Berkeley

ABSTRACT

In this paper we present a methodology for fuzzy sensor fusion. We then apply this methodology to sensor data from a gas turbine power plant. The developed fusion algorithm tackles several problems: 1.) It aggregates redundant (but uncertain) sensor information; this allows making decisions which sensors (and to what degree) should be considered for propagation of sensor information. 2.) It filters out noise and sensor failure from measurements; this allows a system to operate despite temporary or permanent failure of one or more sensors. For the fusion, we use a combination of direct and functional redundancy. The fusion algorithm uses confidence values obtained for each sensor reading from validation curves and performs a weighted average fusion. With increasing distance from the predicted value, readings are discounted through a non-linear validation function. They are assigned a confidence value accordingly. The predicted value in the described algorithm is obtained through application of a fuzzy exponential weighted moving average time series predictor with adaptive coefficients. Experiments on real data from a gas turbine power plant show the robustness of the fusion algorithm which leads to smooth controller input values.

Keywords: Fuzzy Fusion, Fuzzy Validation, Fuzzy Monitoring, Diagnosis

1. INTRODUCTION

To keep systems running at peak performance, to improve operating safety, and to avoid unwarranted shutdowns, often times redundant sensor measurements are taken. The intuitive and empirical notion is that several sensors can give a more reliable and possibly a more correct estimate of the true system status. However, using several sensors necessitates the aggregation of sensor information. Decisions regarding which sensor(s) to believe – and they always disagree at some level of granularity – or to what degree to believe them have to be made. It is the task of validation and fusion techniques to make this decision in the best possible manner[1] and, in conjunction with a supervisory control algorithm, to determine whether sensor failure or system malfunction has occurred[2]. Furthermore, if there are poor readings corrupted by noise and degrading sensor performance, the validation and fusion should give back corrected readings which have filtered out detrimental effects[3]. The proposed FUSVAF (Fuzzy Sensor Validation and Fusion) algorithm uses readings obtained from direct redundant sensors and values calculated through functional redundancy. These values are then fused to render a corrected value which can be used for control purposes.

In the gas turbine power plant application described, a controller attempts to run the gas turbine at a peak temperature subject to the limitations imposed by the material properties of the combustion chamber. This temperature can not be measured directly and must be calculated by other indirect measurements. Unfortunately, these indirect measurements are subject to a number of internal and external disturbances such as changes in ambient temperature and pressure as well as aerodynamic disturbances, internal pressure fluctuations, erosion, corrosion, vibration impingement, flutter, combustion instabilities, material failure, etc.[4,5,6]. Sensor fusion and validation is therefore seen as an important contributor in improving the integrity of the sensor data and efficiency of the operation of the gas turbine.

2. METHODOLOGY FOR FUZZY SENSOR FUSION

The fusion algorithm works in the following manner: incoming sensor readings are validated based on their position within validation gates and based on the position of the old fused value. This fused value is then used to assess the state of the system expressed by α. It is also used for prediction which in turn is necessary to perform the validation of the next time step. The fused value is also used for the machine level controller and supervisory control tasks. The algorithm is displayed in Fig. 1 where z^{-1} denotes a unit delay.

[*] Correspondence: Email: goebelk@crd.ge.com; www: http://best.me.berkeley.edu/~goebelk/kai.html

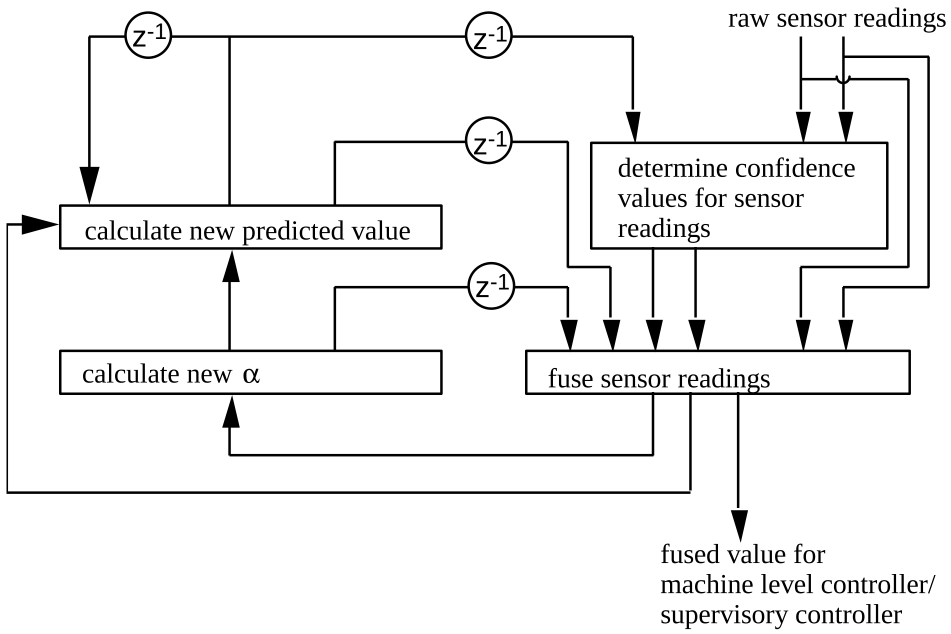


Fig. 1: FUSVAF algorithm for fuzzy sensor validation and fusion

The confidence value which is assigned to all sensor measurements depends on the specific sensor characteristics, the predicted value, and the physical limitations of the sensor value. The assignment takes place in a validation gate which is bounded by the physically possible changes of the system it can undergo in one time step. If the sensor readings show a change beyond that limit, it cannot be a correct value[10]. These limits are the boundaries for the validation gate where sensor readings outside the validation gate are assigned a confidence value of 0 since they do not make sense. Within the validation gates, a maximum value of 1 will be assigned to readings which coincide with the predicted value. The curve between the maximum and the two minima is dependent on the sensor behavior. Generally, this curve is non-symmetric. It has a larger width for a more reliable sensor and vice versa. The curve changes dynamically with the operating conditions which allows to capture the variation in behavior of the sensor over its operating span. Other external effects – such as changes of humidity or temperature on the sensor – resulting in variable sensor performance can be integrated into the validation curves as well by modification of parameter a_{left} and a_{right} in equation (1). These parameters determine the width of the validation curve. A choice for validation curves v(z) for a particular situation could be a piecewise bell curve which need to be scaled from 0 and 1 between the validation border and the predicted value $\hat{x}$. The resulting assignment for confidence values is

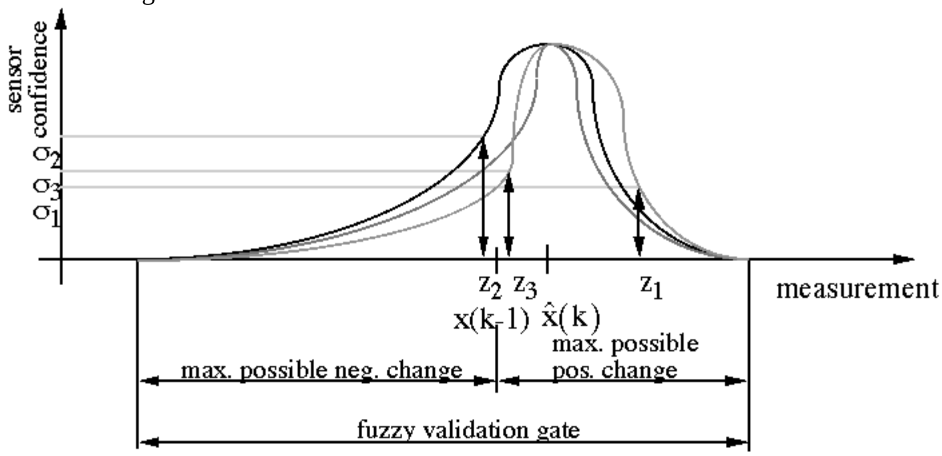
$$\sigma = \begin{cases} 0 & z < v_{left} \\ \dfrac{e^{-\left(\frac{\hat{x}-z}{a_{left}}\right)^2} - e^{-\left(\frac{\hat{x}-v_{left}}{a_{left}}\right)^2}}{1-e^{-\left(\frac{\hat{x}-v_{left}}{a_{left}}\right)^2}} & v_{left} < z \le \hat{x} \\ \dfrac{e^{-\left(\frac{\hat{x}-z}{a_{right}}\right)^2} - e^{-\left(\frac{\hat{x}-v_{right}}{a_{right}}\right)^2}}{1-e^{-\left(\frac{\hat{x}-v_{right}}{a_{right}}\right)^2}} & \hat{x} < z \le v_{right} \\ 0 & z > v_{right} \end{cases} \qquad (1)$$

where

σ is the confidence value for a particular sensor

v_{right} and v_{left} are the right and left validation gate borders, respectively
a_{right} and a_{left} are the parameters for the left and right validation curve
z is the sensor reading
$\hat{x}$ is the predicted value.

A validation gate is shown in Fig. 2.



z_i	sensor measurements
σ_i	sensor confidence values
$\hat{x}(k)$	predicted value
x(k-1)	old value at previous time step

Fig. 2: Validation gate for the assignment of confidence values

The fusion is performed through a weighted average of confidence values and measurements. The sum of the confidence values times the measurements rewards measurements closest to the predicted value the most, depending on the validation curve which expresses a trust in the operation of each sensor through the design of its shape. Measurements further away are discounted. The confidence value drops to 0 at the border of the validation so that measurements beyond the gate borders are not taken into account. The term including the predicted value permits the calculation of fused values even when all sensors have readings outside the validation region. The operative equation in the FUSVAF algorithm is

$$\hat{x}_f = \frac{\sum_{i=1}^{n} z_i \sigma(z_i)}{\sum_{i=1}^{n} \sigma(z_i)} \qquad (2)$$

where

$\hat{x}_f$: fused value
z_i: measurements
σ: confidence values

Note that if all sensors lie on one side of the predicted value, the fused value will also be pulled to the same side. This ensures that evidence from the sensors is closely followed yet discounted the further it gets away from the predicted value.

Using an adaptive time series helps in the trade off between responsiveness, smoothness, stability, and lag of predictors with fixed parameters. We will briefly introduce some of the basic ideas and point to a more thorough description elsewhere[7,11]. The standard EWMA predictor has the form

$$\hat{x}(k+1) = \alpha\hat{x}(k) + (1-\alpha)y(k)\ . \quad (3)$$

If parameter α is set to a fixed value the degree to which new information from sensor readings is used to update the system state is also fixed. This means that the predictor will usually lag behind the true state to some degree. Reducing α results in a predictor which follows the data so closely that noise is not filtered out. Depending on the state of a system, different designs for α would be desirable. If the system is in a more or less steady state, then noise filtering should be the primary task. If, on the other hand, the system is in some sort of transient state, it is more important to track the system change and α should be more responsive. Ideally, α would be adaptive according to the system state. In particular, α should be large when the system is in a steady state and it should be small when the system is in a transient state[11]. These statements can be interpreted as linguistic rules and cast in the framework of fuzzy logic. The problem is more difficult because the predictor under consideration has to rely on the measurements alone and has no external information telling it whether the system is in a steady or a transient state. In that case, conclusions about the state of the system have to be made solely on the basis of past and current observations. Assuming that noise is smaller than the change of the system, one can infer the state of the system from changes in the measurements. This means that if the changes are small, the system is likely to be in a steady state. Otherwise, the system is in some dynamic (or transient) state. Changes in the readings are due more to noise and less to a system change; therefore more weight should be given to the past history and less to the new reading which is likely corrupted by noise[11]. If on the other hand the system is in a transient state, then the change of readings will be due more to the change of the state and less due to noise. As a result, more weight should be given to the incoming reading and less to the past history to allow good responsiveness and little lag. Thus, we would like to have a large α when the system is in a steady state and a small α otherwise.

The membership functions "small", "medium", and "large" have to be designed and were found for the system under investigation as described in Goebel[8]. Tuning of domain specific parameters of the rules should involve optimization for a particular application because learning data have to be conditioned to the environment in which the system is supposed to operate, taking into account all possible scenarios. For this particular application, genetic algorithms were used[7,8] as the optimization technique mainly for its ease of use[12]. While the employment of genetic algorithms helped in fine tuning the parameters, any initial choice would have lead to a functioning system.

Fig. 3 shows data of a non-linear system which was subject to both noise and outliers with a proper choice of the parameters. The sum squared error (SSE) is 0.7361 which allows the system to respond fast yet eliminate outliers and filter out noise.

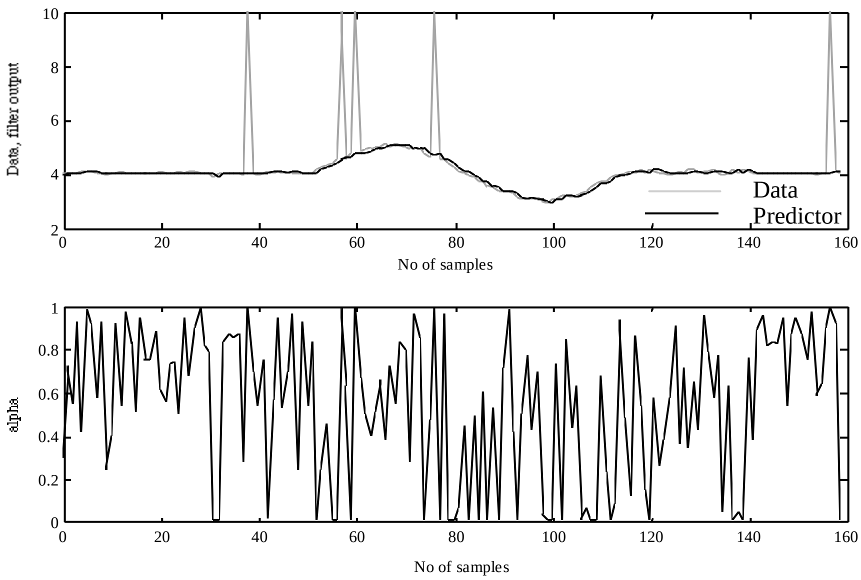


Fig. 3: Predictor with flexible α; superimposed noise and outliers

3. APPLICATION TO GAS TURBINES

This section will first introduce a few domain specific preliminaries before showing the application of the FUSVAF to a particular gas turbine. The central process of a gas turbine – Fig. 4 shows a typical block diagram – is the Brayton cycle.

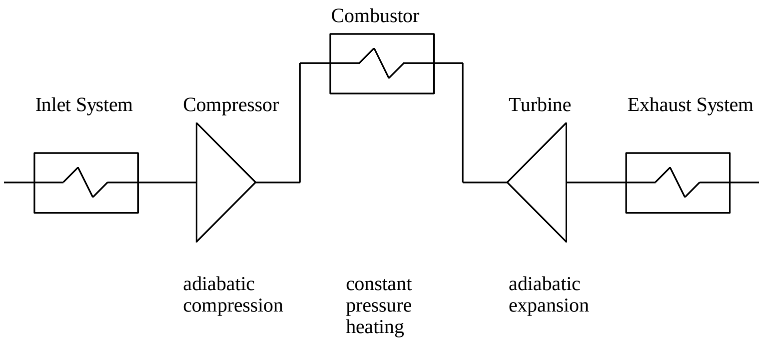


Fig. 4: The gas-turbine process

The ideal Brayton cycle involves isentropic compression in a compressor (no heat exchange), constant pressure heating (burning) in the combustion chamber, and isentropic expansion through a turbine. Fig. 5 shows the enthalpy-entropy diagram of a Brayton cycle. While the ideal Brayton cycle consists of the two vertical isentropics and two constant pressure lines in real life there are efficiency losses within the compressor and the turbine which are represented by the sloped lines. Moreover, there are pressure losses in the inlet, the combustion, and the exhaust which means that the lines from 2 - 3 and 4 - 1 are not true constant pressure lines. Furthermore, the mass flow rate around the cycle diagram is not constant either because air is introduced and extracted for cooling purposes which affect the total work input and output of various processes in the cycle. Therefore, the enthalpy-entropy diagram and its accompanying equations are only an approximation of the real quantities. The net work W_{net} of the cycle is the difference between the work obtained in the expansion process W_t and the work of the compression W_c. The efficiency of the whole process is calculated as the total output divided by the input. Gas turbines typically achieve efficiencies in the mid 30% range.

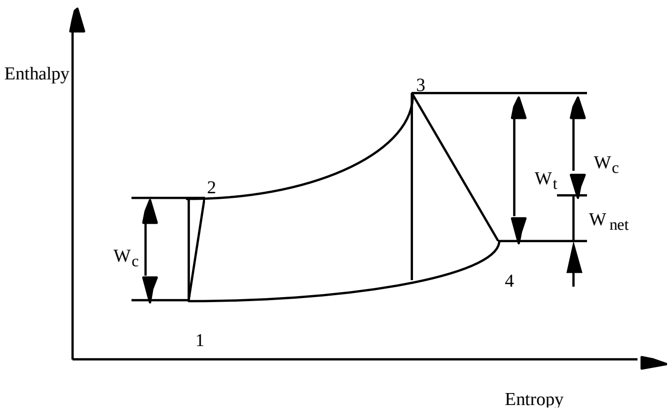


Fig. 5: Enthalpy-entropy diagram of the Brayton cycle

The efficiency is greatest for a maximum firing temperature which is limited to the maximum turbine inlet temperature because of the material properties of the turbine. Given a maximum firing temperature, many cycles could be devised which can be distinguished by their different pressure ratios. At low pressure ratios, the work generated by the turbine greatly exceeds the work by the compressor. However, both works are small, therefore the total output is low. At larger pressure ratios, the efficiency decreases, but the total output increases. If the pressure ratio is very high, the difference between the work of the turbine and the work generated by the compressor becomes so small that the total output decreases as well. Therefore, the cycle peaks at a midrange pressure ratio as displayed in Fig. 6.

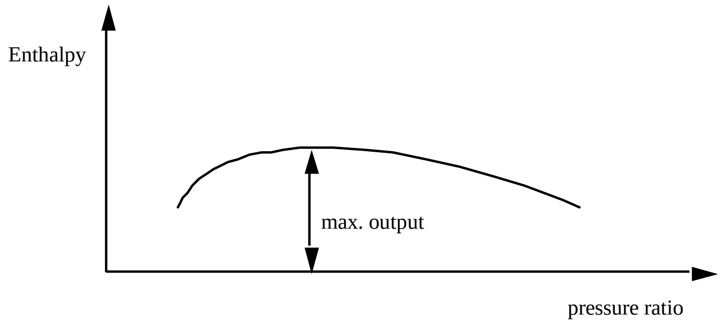


Fig. 6: Pressure ratio-enthalpy diagram

The goal is to operate the turbine at maximum allowable temperature T_3 which cannot be measured directly. Instead, the control algorithm uses measurements of other variables in the power plant to control the relevant parameters. The compressor discharge pressure p_2 is often used in control algorithms for gas turbine driven power plants. p_2 can be measured directly. However, the pressure sensor is subject to noise which has a direct influence on the performance of the control algorithm and ultimately on the load generated. For evaluation of the algorithm, we used data of an extant gas turbine power plant of the 40MW class to create models used for simulation of sensors and system components[8]. The data observed exhibited several different types of noise. One type was slow changes with superimposed Gaussian noise where the slow changes (24 hours) can be attributed to changes of ambient temperature and pressure throughout the day. There is another even slower cycle with frequency 1 cycle/year which reflects the change of the mean ambient temperature over the seasons. This means the mean compressor discharge pressure is different when observed at different days in the year. Similarly, weather conditions may have an influence in the readings as well. Another noise type is due to quantization effects of data acquisition systems which round off data and result in apparent jumps. Yet other noise was observed as random outliers of small magnitude. The control algorithm of the power plant reacts to all these changes to keep the power plant at maximum performance. It also reacts to the noise which is often times not due to the system behavior. To achieve a better control, noise has to be filtered out and an improved value close to the real one needs to be given back. Measurements from sensors measuring different variables can be used to compute the control input value and can be fused to achieve smoother and more robust control. Noise has to be rejected for each sensor. The sensors used are listed in Table 1.

TABLE 1
Sensors used for gas turbine

Quantity measured	**Symbol**
Exhaust Temperature	T_4
Compressor Inlet Flange Temperature	T_{11}, T_{12}
Compressor Discharge Temperature	T_{21}, T_{22}
Compressor Discharge Pressure	p_2
Load generated	W
Fuel Flow	$\dot{m}$
Inlet Duct Manifold Pressure	p_i
Ambient Barometric Pressure	p_a

In the following section we will show how sensor validation and fusion can be used to achieve better measurements and hence better performance of the controller. Furthermore, we will show that the algorithm is robust against sensor failure which could result in wrong input to the controller with consequences not only for the performance but also for a potential shutdown which is to be avoided if possible. The efficiencies of the compressor and the turbine were assumed to be η_c =0.86 and η_t =0.9275, respectively. The validation and fusion algorithm fuses the two values from the two inlet flange temperature sensors to get one inlet flange temperature value. It then does the same with the two compressor discharge temperature sensors to get one fused compressor discharge sensor. Then the compressor discharge pressure is calculated as

$$p_2 = p_1 \left(\frac{\eta_c \left(T_{2_f} - T_{1_f} \right) + T_{1_f}}{T_{1_f}} \right)^{\frac{\kappa}{\kappa - 1}}$$

where

T_{1_f} is the fused inlet flange temperature

T_{2_f} is the fused compressor discharge temperature

To reduce the lag, smoothing the rugged readings and retaining the robust nature, the scheme displayed in Fig. 7 is used. It consists of smoothing all sensor readings through the use of the fuzzy time series predictor, followed by fusing the values for temperature sensors T_{11} and T_{12} to form T_{1f} as well as for T_{21} and T_{22} to form T_{2f}. Then p_{22} and p_{23} are calculated where p_{22} is calculated as in the previous example and p_{23} is calculated as

$$p_2 = p_1 \left(\frac{T_{2i}}{T_{1_f}} \right)^{\frac{\kappa}{\kappa - 1}}$$

where

$$T_{2i} = T_1 \left(\frac{T_3}{T_{4i}} \right)$$

$$T_3 = T_4 + \frac{W}{\dot{m} c_p}$$

$$T_{4i} = T_3 - \left(\frac{T_3 - T_4}{\eta_t} \right)$$

Finally, the three values for p_2 are fused to give p_{2f}.

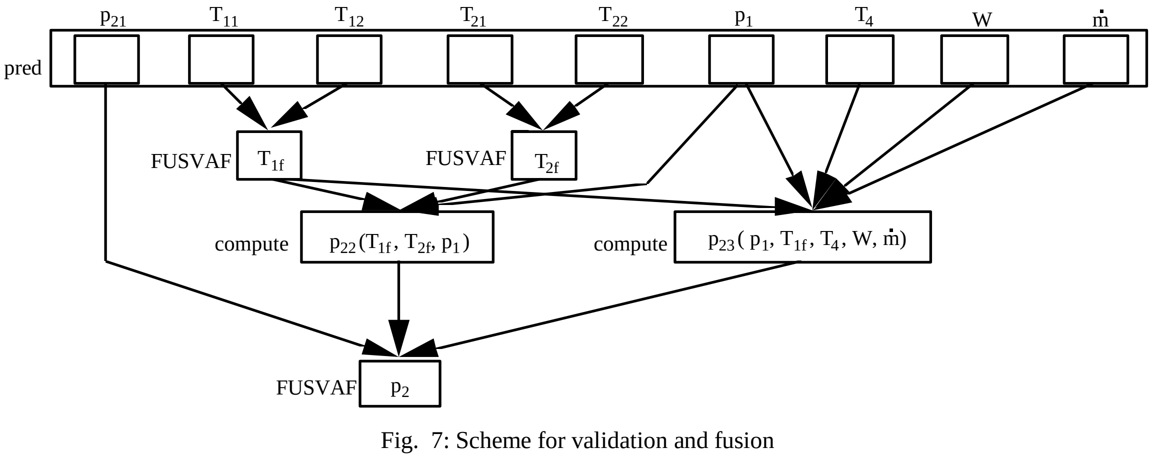


Fig. 7: Scheme for validation and fusion

Fig. 8 shows the output of the scheme used under normal operating conditions. Apart from the measured value of p_2, the computed value for p_{21} and p_{22} are also displayed along with the fused value p_{2f} and the smoothed fused value p_{2fm}. The latter has the most desirable features. It is smooth yet responsive to the change of the readings.

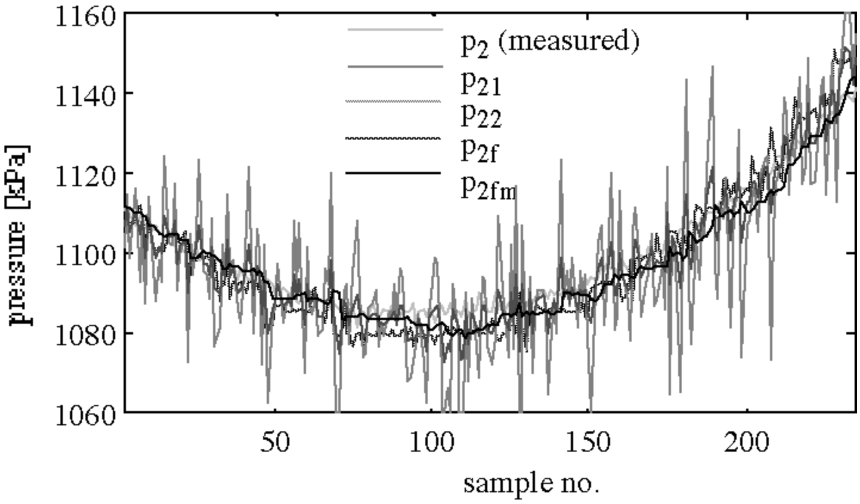


Fig. 8: Validation and multi-fusion under normal operation

In a next step, Monte Carlo simulations were used to investigate how the system behaves under sensor failure. We show in Fig. 9 a summary of the observations with forced multi-sensor failures for illustrative purposes. Failures were superimposed every 25 samples as summarized in Table 2. T_{11} fails from samples 51 to 200, T_{21} fails from samples 76 to 225, p_2 fails from samples 101 to 175, and T_4 fails between samples 126 and 150. The mass flow sensor fails as well in the range from 76 to 150 samples. The fusion algorithm disregards the calculation based on the faulty readings and uses the calculations based on T_{12}, T_{22}, and p_1 only. The smoothed value for the estimation of p_2 still provides a good input to the power plant controller.

TABLE 2
Summary of failures imposed on power plant data for five simultaneous sensor failures

Sensor failed	Range of failure
T_{11}	51-200
T_{21}	76-225
p_2	101-175
T_4	126-150
$\dot{m}$	76-150

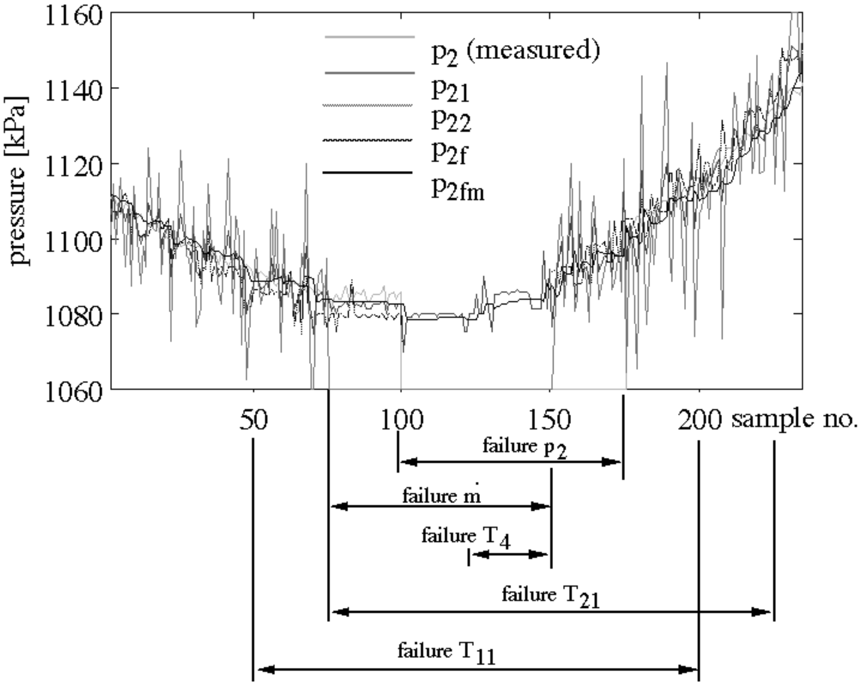


Fig. 9: Output of validation and fusion scheme subject to failure of five sensors

There are of course fault scenarios when too many sensors fail resulting in a lack of information for the algorithm to come up with any meaningful value for p_2. If in addition to the five failures shown in the previous example, the inlet duct manifold pressure sensor fails as well in the range from samples 126 to 225, there is not enough sensor information available from any sensor combination to calculate a good value for p_2. Rather, the algorithm keeps the last recorded value as the best estimate for the system state. The algorithm also issues a warning alerting the operator to a possible system failure which can trigger appropriate action. When some sensors come back on-line, they are again used to update the estimate for p_2. This validation and fusion scheme has the most desirable features. It filters out noise and bad sensor readings and provides a robust platform for the power plant controller.

It is important to observe the combination of sensors which in a failed state render the algorithm without enough information for a acceptable value. There are other sensor failure combinations than the ones shown which also still allow proper determination of p_2 some involving more than five simultaneous failed sensors. However, there are also combinations of sensor failures which occur when fewer sensors fail. In particular, p_1 is used in both calculating p_{22} and p_{23}. Therefore, if p_2 and p_i or p_a fail at the same time over longer periods, the algorithm will not be able to give proper results. This stresses the importance of the use of independent sensors. From the suite of sensors at hand, however, this bottleneck cannot be circumvented.

4. SUMMARY AND CONCLUSIONS

The FUSVAF algorithm calculates confidence values for sensor readings using non-linear dynamic validation curves and a weighted average aggregation. Validation curves are centered within validation gates at a predicted value which is arrived at using an adaptive exponential weighted moving average predictor. In the application considered, we used the fusion algorithm to integrate data obtained through functional redundancy in algorithms using double and triple redundancy. A higher degree of redundancy leads to higher robustness against sensor failure, e.g., the algorithm used with three redundant values for the control variable p_2 is more robust than when only two redundant readings are used, which in turn is more robust than the controller relying on the direct measurement of p_2 alone. All three values for the control input discharge pressure p_2 are obtained through largely independent readings. Thus, if sensors fail on a particular subsystem, the readings from the other subsystems are able to maintain proper control input and unnecessary shutdowns can be avoided. In essence, the more sensor readings are integrated into the fusion algorithm, the smoother, reactive, and robust is the control input. We also show how superimposed sensor failure is rejected for multiple simultaneous sensor failures. The fusion scheme has been successfully applied to other applications, in particular to intelligent highway systems[9] and aircraft engines. The strength of the algorithm is its flexibility in dealing with changing situations and in dealing with different noise types, including non-Gaussian noise.

Future research should address how best to fine tune these parameters, depending on operating conditions and sensor history. We accomplished this through machine learning techniques using real data[8] but a more intuitive way would be preferred. Also important is to establish sensor characteristics which allow proper modeling of the validation curves and in addition provide means for diagnosis. Since a higher degree of redundancy results in more robust operation, it would be worthwhile to investigate the use of other sensors in the validation and fusion algorithms for the gas turbine application, especially if known functional relationships can be exploited[10]. It is crucial, however, to maintain maximum independence of the values computed – i.e., no two calculated values should rely on the same sensors – to avoid failure of several calculated values when one sensor malfunctions.

5. ACKNOWLEDGMENTS

This research was in part supported by PATH (Partners in Advanced Transit Highways) and Caltrans through grants MOU-132 and MOU-157 as well as project MICRO (contract #93-004). Special acknowledgments go also to Mahesh Morjaria from General Electric CR&D.

6. REFERENCES

1. J.H.P. Miller, "Total Process Surveillance (TOPS)", *Nuclear Energy*, vol. 32 (no.1), 1993.
2. A.M. Agogino, S. Srinivas, and K. Schneider, "Multiple Sensor Expert System for Diagnostic Reasoning, Monitoring, and Control of Mechanical Systems", *Mechanical Systems and Signal Processing*, Vol. 2(2), 1988.
3. C. Jiaa and D. Dornfeld, "Detection of Tool Wear Using Gradient Adaptive Lattice and Pattern Recognition Analysis", *Mechanical Systems and Signal Processing*, vol.6, (no.2), 1992.
4. G.P. Singh, D.A. Steinke, J.L. Tsou, "Heat Exchanger Workstation: A Comprehensive Software Tool for Heat Exchanger Operation and Maintenance", *Proceedings of the American Power Conference*,1993.
5. Sir H.R. Cox, *Gas Turbine Principles and Practice*, London: George Newnes Limited, 1953.
6. H. Constant, *Gas Turbines and Their Problems*, London: Todd Publishing Group Limited, 1953.
7. K. Goebel, and A.M. Agogino, "An Architecture for Fuzzy Sensor Validation and Fusion for Vehicle Following in Automated Highways", *Proceedings of the 29th ISATA*, 1996.
8. K.Goebel, Management of Uncertainty for Sensor Validation, Sensor Fusion, and Diagnosis Using Soft Computing Techniques, Ph.D. Thesis, University of California at Berkeley, 1996.
9. A.M. Agogino, S. Alag, and K. Goebel, "A Framework for Intelligent Sensor Validation, Sensor Fusion, and Supervisory Control of Automated Vehicles in IVHS", *Proceedings of the 1995 Annual Meeting of ITS America*, 1995.
10. Y.J. Kim, W.H. Wood, and A.M. Agogino, " Signal Validation for Expert System Development", Proceedings of the 2nd International Forum on Expert Systems and Computer Simulations in Energy Engineering, 1992.
11. P. Khedkar and S. Keshav, "Fuzzy Prediction of Time Series, *Proceedings of the IEEE International Conference on Fuzzy Systems*, 1992.
12. D.E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Reading: Addison Wesley, 1989.