A Model-Based Framework for Fault Estimation and Accommodation Applied to Distributed Energy Resources

James T. Allen and Nael H. El-Farra*

* Department of Chemical Engineering, University of California, Davis, CA 95616 (e-mail: nhelfarra@ucdavis.edu).

Abstract: This paper presents the development and approach of a model-based fault identification and accommodation framework applied to sampled-data controlled distributed energy resources subject to control actuator faults. The main objective of the proposed approach is to handle faults that degrade stability as well as performance, while remaining robust to false alarms. The proposed method allows for dual fault detection and estimation, through the use of an embedded system model that minimizes the residual between the estimated and sampled states at each sampling period by adjusting a fault parameter in the embedded model over a past horizon. The resulting fault parameter estimate is then used by the control system to find an optimal fault accommodation strategy by minimizing a predefined performance metric whilst ensuring closed-loop stability. The developed fault accommodation framework is then applied to a simulated model of a solid oxide fuel cell subject to both stability and performance degrading faults in the control actuators. A discussion of some of the practical implementation issues associated with the developed framework is also included.

Keywords: Fault estimation; Fault accommodation; Model-based control, Sampled-data systems, Distributed energy resources

1. INTRODUCTION

Distributed energy resources (DERs) are composed of modular energy generation units including, for example, micro-turbines, fuel cells, renewable energy systems, battery storage and other such technologies deployed close to the point of consumption. The modular nature of DERs allows them to be integrated with existing grid infrastructure or implemented in a stand-alone manner. DERs offer advantages over conventional grid electricity by offering end users a diversified fuel supply; higher power reliability, quality, and efficiency; lower emissions and greater flexibility to respond to changing energy needs.

While control of DERs is necessary to ensure that the load demand is met and that economical operation of each DER is maintained (e.g., see Tolbert et al. (2001); Illindala and Venkataramanan (2002); Marwali and Keyhani (2004); Dimeas and Hatziargyriou (2005); Roberts et al. (2006); Lasseter (2007); Sun et al. (2009); Qi et al. (2011)), the stability and performance of smart grid DERs have not been rigorously assessed in the presence of faults or failures at the local or network levels. This is an important problem given the fact that the distributed power market is driven by the need for reliable high-quality power, and the fact that local faults and disruptions in power flow can have a substantial impact, especially in situations when DERs are integrated to support grid operations. The timely identification and mitigation of faults at the local level before they cascade through the network are important capabilities that ensure autonomous control and protection, which when coupled with proper supervisory oversight enable distributed energy generation to provide highly reliable services under all disturbance and fault scenarios. In this context, fault-tolerant control is an important tool for reducing performance deterioration in the face of faults and uncertainties in the system components, such as actuators and sensors, resulting in increased reliability of the network.

In prior work (Sun et al. (2010)), fault-tolerant control has been studied within the context of a small-scale network of solid oxide fuel cells (SOFCs), where the focus has been on the detection and handling of destabilizing failure events at the local level without supervisory oversight. Local monitoring of the health status of each DER took place through use of a time-varying alarm threshold on a properly designed observer-based output residual. Exploiting the inherent actuator redundancy in SOFCs, three stabilizing controller configurations were designed, and a methodology for active switching between them in the event of a threshold breach was developed. The main contribution was the characterization of a stability region within which each controller configuration could operate. However, the faults considered were limited to total failure events, thus negating the need for fault isolation or estimation. The proposed stability-based scheme also was not designed to detect faults that only degrade performance but do not compromise stability, and this can lead to sub-optimal performance.

* Financial support by NSF, CBET-1438456, is gratefully acknowledged.
Motivated by these considerations, this paper presents a framework for local actuator fault-tolerant control that accounts for both performance and stability degrading faults, while optimizing suitable performance metrics and maintaining closed-loop stability. The developed framework is realized through an integrated approach that brings together model-based control, data-based fault estimation and performance-based fault accommodation. The rest of the paper is organized as follows. A SOFC system is initially introduced in Section 2 to motivate the development and application of the proposed framework. This model system is used to design a local model-based state feedback controller that operates on sampled data and characterizes its stability properties in the presence of faults. A fault estimation scheme that provides an estimate of the fault magnitude by solving a data-based moving horizon optimization problem is then introduced in Section 3. A performance index suitable for sampled-data systems is then introduced in Section 4 and used to develop a performance-based fault accommodation strategy that alters model and control parameters to minimize post-fault performance losses while maintaining closed-loop stability. Practical implementation issues, such as the need to minimize false alarms, are discussed and guidelines for dealing with these issues are presented. Finally, simulation results are presented in Section 5.

2. PRELIMINARIES

2.1 Motivating example: A solid oxide fuel cell

Due to their modular and stable nature SOFCs are of particular interest when it comes to decentralized distributed energy generation. A SOFC consists of two porous electrodes, an anode and a cathode, in contact with a solid metal oxide electrolyte between them. Hydrogen is fed along the surface of the anode where it releases electrons that migrate externally towards the cathode. The electrons combine with oxygen fed along the surface of the cathode to form oxide ions. These ions diffuse through the electrolyte towards the anode where they combine with the hydrogen ions to produce water and power. Under standard modeling assumptions (see Murshed et al. (2007)), the following dynamic model of the SOFC stack can be obtained from first principles:

\[
\dot{p}_H = \frac{T_s}{\tau H^* T + K_H^*} (q_{in}^H - K_H^* p_H^* - 2 K_r^* I)
\]

\[
\dot{p}_O = \frac{T_s}{\tau O^* T + K_O^*} (q_{in}^O - K_O^* p_O^* - K_r^* I)
\]

\[
\dot{p}_O^* = \frac{T_s}{\tau O^* T + K_O^*} (q_{in}^O - K_H^* p_H^* - 2 K_r^* I)
\]

\[
\dot{T}_s = \frac{1}{m_s c_p s} \left[ \sum q_{in}^i \int_{T_{ref}}^{T_s} C_p i(T) dT - \sum q_{in}^i \int_{T_{ref}}^{T_s} C_p i(T) dT - 2 K_r^* I \Delta H^* - V_s I \right]
\]

\[
V_s = N_0 \Delta E - r_0 \exp \left[ \alpha \left( \frac{1}{T_s} - \frac{1}{T_0} \right) \right] I
\]

\[
\Delta E = \left[ \Delta E_0 + \frac{RT}{2F} \ln \left( \frac{p_{H_2}^0}{p_{H_2}^*} \right) \right]
\]

For the component mass balances, \( p_i \) is the partial pressure of component \( i \), \( T_s \) is the stack temperature, \( \tau_i \) is the time constant for component \( i \), described by \( \frac{1}{\tau_i} = \frac{1}{\tau_{ref}} + \frac{1}{\tau_i^*} \), with \( V \) being the volume component \( i \) is contained in, \( K_i \) is the valve molar constant for component \( i \), \( R \) is the gas constant, \( \tau_i^* = \frac{1}{\tau_{ref}^*} \), and \( T^* \) is the operating temperature. \( q_{in}^i, p_i \) are the inlet flow rate and partial pressure of component \( i \), respectively, \( K_r = \frac{N_0}{4F} \) where \( N_0 \) is the number of cells in the stack and \( F \) is Faraday’s constant. Lastly \( I \) is the load current. As for the energy balance, \( m_s \) and \( c_p s \) are the mass and heat capacity of the stack, \( T_{ref} \) and \( T_0 \) are the reference and feed temperatures, \( C_{p,i} \) is the heat capacity of component \( i \), \( q_{out}^i \) is the outlet flow rate of gas \( i \), \( \Delta H^* \) is the specific heat of reaction, \( V_s \) is the stack voltage, \( r_0 \) is the internal resistance, \( \alpha \) is the resistance slope, and \( \Delta E_0 \) is the standard cell potential.

2.2 Control problem formulation with fault modeling

The SOFC system modeled in Eq.1 has three potential manipulated variables. Previous work has exploited this fact to implement multiple control configuration in the case of actuator failure (see Sun et al. (2010)). However, here the focus will be on the accommodation of faults through model manipulation, and only control configurations utilizing the hydrogen gas flow rate as the manipulated variable will be used. A similar approach can be used if other manipulated variables are chosen instead (the oxygen flow rate or the feed temperature).

As in Sun et al. (2010), the controller design is based on the linearized system model around the desired operating set-point, where the time evolution of the system is given by:

\[
\dot{x} = Ax + B\theta u
\]

\[
\dot{\hat{x}} = \hat{A}\hat{x} + \hat{B}\hat{u}
\]

\[
u = K\hat{x}
\]

where \( x = [p_H^* p_O^* p_{H_2}^* T_s]^T \) is the state vector of deviation variables, \( \hat{x} \) is the model state which provides an estimate of \( x \), \( \hat{A}, \hat{B} \) are constant model matrices that approximate the linearized system matrices, \( A, B \) allowing for consideration of plant-model mismatch (see Eq.5). The effect of faults is captured through \( \theta \) which, in general, is a diagonal matrix whose diagonal elements represent the fault or health status of each control actuator, where a value of 1 represents a healthy actuator and a value of zero reflects total failure. \( \hat{\theta} \) is an estimated fault parameter used in the control model and will serve as a key fault accommodation parameter. Finally, \( K \) denotes the feedback controller gain.

\[
A = \begin{bmatrix}
-0.034998 & 0 & 0 & 0 \\
0 & -0.3139 & 0 & 0 \\
0 & 0 & -0.011666 & 0 \\
-8.9278 & -28.673 & -3.2569 & -0.010988 \\
\end{bmatrix}
\]

\[
B = \begin{bmatrix}
0.0034721 \\
0 \\
0.63486 \\
\end{bmatrix}
\]

To analyze the stability and performance properties of the model-based controller of Eqs.2-4, we define an augmented state vector of the form \( \xi = [x \, e]^T \), where \( e = x - \hat{x} \) is
the model estimation error. The sampled-data closed-loop system can then be cast in the following form:
\[ \dot{\xi} = \Lambda \xi + e(t_j), \quad t \in [t_j, t_{j+1}) \]
where
\[ \Lambda = \begin{bmatrix} A + B\theta K & -B\theta K \\ (A - \hat{A}) + (B\theta - \hat{B}\theta) K & \hat{A} - (B\theta - \hat{B}\theta) K \end{bmatrix} \tag{7} \]

Note that the model estimation error is reset to zero at each sampling time due to the update of the model state using the actual sampled state. It can then be shown that the solution to the augmented system has the form:
\[ \xi(t) = e^{\Lambda(t-t_j)} (I_o e^{\Delta} I_o) \xi_0, \quad t \in [t_j, t_{j+1}) \tag{8} \]
where \( \Delta = t_{j+1} - t_j \) is the sampling period and \( I_o = I_o = \begin{bmatrix} I & 0 \\ 0 & O \end{bmatrix} \), with \( I \) being the identity matrix. Due to the discrete sampled nature of the system it can be shown that closed-loop stability is assured if the maximum eigenvalue magnitude of the matrix \( M := I_o e^{\Delta} I_o \) is less than 1, i.e., \( \lambda_{\max}(M) < 1 \). Owing to the parametrization of the stability test matrix \( M \) in terms of the various model, fault and control system parameters, the stability condition can be used to systematically explore the balance between these various parameters in influencing closed-loop stability. For example, for a fixed model and controller design parameters, the stability condition can be used to determine the range of tolerable fault magnitudes that do not compromise closed-loop stability. Alternatively, for a given fault size, the same condition can be used to identify the possible ranges for model and/or controller parameters that would ensure stability in the presence of the fault. This link is the basis for stability-based fault accommodation strategies.

3. FAULT DETECTION AND ESTIMATION VIA MOVING HORIZON OPTIMIZATION

The first step in fault handling is the detection and estimation of faults. This is done through a moving-horizon optimization problem formulation (inspired by the formulation presented in Samar et al. (2006)) in which an estimated value of \( \theta \) (this will be referred to as \( \theta^* \)) is computed at each sampling time by minimizing the error between the sampled plant state and the estimated value provided by the fault diagnoser model. The cost function to be minimized is given by:
\[ J(\zeta_j, \theta) = \sum_{p=j}^{N_f+j+1} \left( \|x[p+1] - \hat{x}[p+1]\|^2 \right) \tag{9} \]
where
\[ \zeta_j = \left( x[j-p], u[j-p], \hat{\theta}[j-p], \Delta[j-p] \right), \quad p \in \{1, 2, \ldots, N_f\} \tag{10} \]
and \( N_f \) is the horizon size, where each point in \( N_f \) is a time at which the plant state is sampled. \( \hat{x} \) is obtained through initializing an embedded model of the system at \( \hat{x}[i] = x[i] \) and then simulating the plant response over the sampling period using the \( \xi \) parameters fed to \( J \) at which point \( \hat{x}[p+1] \) is compared with \( x[p+1] \), and then \( \hat{x} \) is initialized at time \( p+1 \) and the process is repeated over the horizon window. By minimizing the cost function in Eq.9 the value for the estimated fault parameter can be obtained as:
\[ \theta^* = \arg\min_{\theta} J(\zeta_j, \theta) \tag{11} \]

Unlike residual-based approaches (e.g., Sun et al. (2010)) which allow only for fault detection, the above optimization-based approach provides capabilities for simultaneous detection, isolation and estimation of multiple faults, which is important for the selection of the appropriate fault accommodation strategy.

4. FAULT ACCOMMODATION STRATEGIES

4.1 Performance metric characterization

In order to accommodate a fault from a performance standpoint, a performance metric must be in place. The advantages of performance-based accommodation are that through the use of a performance metric, different stabilizing configurations can be compared objectively to decide which one is better suited to the needs of the process. Due to the sampled nature of the control system considered here, the extended H2-norm is chosen as a suitable performance metric (e.g., see Montestruque and Antsaklis (2006); Sun et al. (2009)). The extended H2-norm is used for measuring the 2-norm of a periodically sampled-data systems where continuous measurements are not available. It is a measure of the settling time of the performance output of a system after an impulse disturbance is introduced in a prescribed input.

To characterize the extended H2-norm as a function of the plant, model, fault and control system parameters, we reconsider the state-space representation in Eq.2, but now with disturbances and a performance output introduced as follows:
\[ \dot{x}(t) = Ax(t) + B\theta u(t) + Ed(t) \tag{12} \]
\[ z(t) = Jx(t) \]
where \( d \) is the disturbance input and \( z \) is the performance output. In the case of the SOFC plant, the performance output chosen is the power output of the system. After some manipulations, the sampled-data closed-loop system can be formulated as:
\[ \dot{\xi}(t) = \Lambda \xi(t) + H(\theta)d(t), \quad t \in [t_j, t_{j+1}) \]
\[ \xi(t_j) = [x^T(t_j) 0]^T, \quad j \in \{0, 1, 2, 3, \ldots\} \tag{13} \]
\[ z(t) = N^T \xi(t) \]
with \( H = [E^T E^T]^T \) and \( N = [J O] \). By solving Eq.13, it can be verified that the closed-loop response of the performance output to an impulse disturbance \( d = \delta(t-t_0) \) can be expressed as:
\[ z(t) = Ne^{\Lambda(t-t_j)} (I_o e^{\Delta})^j H, \quad t \in [t_j, t_{j+1}) \tag{14} \]
and the extended H2-norm can be computed as follows:
\[ \|G\|_{H_2} = \text{trace}(H^T XH)^{1/2} \tag{15} \]
where \( X \) is the solution of the following discrete Lyapunov equation:
\[ M(\Delta)^T X M(\Delta) - X + W_0(0, \Delta) = 0 \tag{16} \]
with \( M(\Delta) = I_o e^{\Delta} \) and \( W_0(0, \Delta) = \int_0^\Delta e^{\Delta T} N^T N e^{\Delta T} d\tau \). The extended H2-norm is of particular interest here seeing that it can be expressed in closed form and is also a
function of the same parameters used to characterize closed-loop stability, namely $\theta$, $\hat{\theta}$, $\Delta$, and $K$. This allows a quantitative assessment of the potential balance that exists between these various parameters in influencing closed-loop performance. For example, this characterization of the performance metric can be used to assess the extent of performance deterioration resulting from different fault situations. Similarly, for a given fault scenario, the impact of different choices of the control and model parameters on performance can be analyzed. Standard optimization techniques can be used, for example, to find the minimum extended $H_2$-norm as a function of these variables. This is useful in determining the best plan of action for the accommodation of faults.

4.2 Performance-based fault accommodation

Once a fault in need of accommodation has been deemed to occur, the controller must take a suitable corrective action to compensate for the effect of the fault. This is achieved by minimizing the performance objective (over one or more possible fault accommodation parameters), whilst ensuring closed-loop stability of the system. Note here that the performance objective is minimized because it denotes the settling time of the system from an impulse disturbance. For example, if a fault is established at time $t_j$ and $\hat{\theta}$ is chosen as the fault accommodation parameter, the value that $\hat{\theta}(t_{j+1})$ should be assigned is determined by solving the following constrained optimization problem:

$$\hat{\theta}(t_{j+1}) = \min_{\theta} ||G||_{H_2}$$

s.t. $\lambda_{\max}\left(M(\hat{A}, \hat{B}, \hat{\lambda}, \hat{B}, K, \theta, \hat{\theta}, \Delta)\right) < 1$

where $\hat{A}$ and $\hat{B}$ are the modeled values of $A$ and $B$ in the embedded fault estimator, and the constraint on the maximum eigenvalue magnitude is made to ensure closed-loop stability (see Section 2.2). In some cases, it may be necessary to perform the minimization of the extended $H_2$-norm over a constrained range of possible $\hat{\theta}$ values to avoid unreasonably large values that may result from the solution of the unconstrained problem. A similar formulation to the one given in Eq.17 can be used if other possible fault accommodation parameters are chosen (e.g., the feedback gain $K$, or the sampling period, $\Delta$). One can also optimize the performance index over multiple fault accommodation parameters simultaneously; this, however, comes at the expense of additional computational complexity.

4.3 Implementation issues

An important consideration in the implementation of fault-tolerant control is the need to minimize false detection alarms which may trigger unnecessary fault accommodation events that, at best, could lead to increased computational cost, and, at worst, could compromise performance and lead to instability. This is important, especially in the presence of unmodeled uncertainties and unknown perturbations that could impact the optimization-based fault estimation results. Determining whether or not a fault is in need of accommodation, therefore, requires careful consideration. For example, once a change in the estimated fault parameter has been observed, the controller must determine if the fault is new or one that has had previous accommodation. This determination can be made through the use of a reference fault parameter, $\theta_{ref}$, which is updated every time a fault is accommodated. An example heuristic that one can employ here is to check if the mean of the estimated fault parameter over the past $N_\mu$ sampled data points, $\mu$, has deviated by some pre-defined threshold, $\theta_{thresh}$, i.e.,

$$|\theta_{ref} - \mu| > \theta_{thresh}$$

(18)

$$\mu = \text{mean}\{\theta^*[p - N_\mu], \theta^*[p - N_\mu + 1], \ldots, \theta^*[p]\}$$

(19)

If such a deviation is observed, then the controller suspects that a new fault has occurred.

Another important aspect is the need to provide some robustness margin against unmodeled disturbances. Given that the performance of the system under impulse-like disturbances is considered (which is captured by the extended $H_2$-norm), it is important that such unmodeled disturbances not be mistaken for faults and thus not trigger unnecessary fault accommodation measures. One approach to deal with this is to exploit the fact that any impulse-like disturbance on the system will rapidly shift the states, and will be met by the embedded model attempting to find an estimated value of the fault size that would cause this stark shift, thus creating a rapid change in the estimated fault parameter. With this knowledge, a secondary condition for the determination of possible faults would be to require that the estimated fault parameter not exceed some predefined threshold, $\theta_{max}$, that would not be expected to be reached under normal operating conditions. In other words, if the following condition holds:

$$\max\{\theta^*[p - N_\mu], \theta^*[p - N_\mu + 1], \ldots, \theta^*[p]\} < \theta_{max}$$

(20)

as well as that in Eqs.18-19, then the controller decides that a fault has occurred that must be accommodated.

Since the fault accommodation strategy is based on the estimated fault size, it is important once a fault has been suspected to wait for the estimated fault value to settle prior to taking corrective action. This would have the benefit of not taking unnecessary action during disturbances, as well as reducing the computational load of calculating the accommodation action every time $\theta^*$ changes (recall that fault accommodation is based on solving the constrained optimization problem in Eq.17). The settling time of the estimated fault parameter is dependent on the size of the horizon used in the fault estimation scheme. The estimated fault value gradually changes as additional faulty data points are incorporated into the data set used for estimation. Eventually the estimated value settles at a constant once all data points in the horizon are ones containing the fault. An example heuristic that can be used here is to trigger an optimization of the control action only when the deviations in successive estimates of $\theta^*$ fall below a certain threshold, $\epsilon_{\mu}$, i.e.,

$$|\mu' - \theta^*[p]| < \epsilon_{\mu}$$

(21)

$$\mu' = \text{mean}\{\theta^*[p - N_\mu], \theta^*[p - N_\mu + 1], \ldots, \theta^*[p - 1]\}$$

(22)

where $\epsilon_{\mu}$ is a tunable parameter that $\theta^*$ must remain within for the optimization-based fault accommodation to take place.
5. SIMULATION STUDY

The objective of this section is to illustrate the implementation of the developed fault estimation and accommodation framework to the SOFC example introduced earlier in Section 2. In prior work (Sun et al. (2010)), stability regions were shown to aid in understanding the impact uncertainty had on closed-loop stability of the SOFC plant and used to develop a stability-based control system reconfiguration strategy that maintains stability in the presence of controller failures. In this study, a different approach is taken, and that is to show explicitly the dependence of stability and performance on the actual and modeled faults in the system. To this end, Fig.1 is a contour plot of \( \lambda_{\max}(M) \) as a function of \( \theta \) and \( \hat{\theta} \). The uncolored region is the region enclosed by the unit contour line and is therefore the stable region of operation. It can be seen from this plot that for any given fault, where \( \theta > 7.44 \) and \( \hat{\theta} = 1 \), the closed-loop system will be unstable. Furthermore, there is a wide range of possible accommodation scenarios that can be implemented through altering \( \hat{\theta} \) that lead to stable post-accommodation operation.

Fig.2 shows the advantage of fault accommodation when a destabilizing fault of \( \theta = 7.9 \) occurs at \( t = 500 \) s. The closed-loop system with the accommodated fault (\( \hat{\theta} = 15 \)) maintains a steady power output trajectory (black profile) in the face of a fault in the actuator manipulating the hydrogen flow rate, whereas the power output of the closed-loop system with no fault accommodation (\( \hat{\theta} = 1 \)) is shown to be unstable for the same fault (red profile). Given the wide range of values of \( \hat{\theta} \) that can be used in the region where the closed-loop system is stable (see Fig.1). Due to the shape of the stability contour plot, simply setting \( \hat{\theta} = \theta^* \) can be seen to place the system in a stable region of operation. This is referred to as “minimal accommodation” in the sense that the controller needs no knowledge of the stability or performance of the system as a function of \( \hat{\theta} \). This minimal accommodation strategy assumes that if the model matches the plant, then this will be a suitable accommodation strategy. It can be seen with the aid of the performance contour plot of the extended \( H_2 \)-norm (Fig.3), however, that there is in fact a better solution to the accommodation problem from a performance standpoint than simply setting \( \hat{\theta} = \theta^* \). It should be noted that the shape of this particular contour plot is unique to the chosen control configuration, and that, in general, setting \( \hat{\theta} = \theta^* \) cannot be guaranteed to result in a stable post-fault system even if \( \theta^* = \theta \).

To illustrate the performance-based accommodation, we consider a non-destabilizing fault in the control actuator at \( \theta = 1000 \) s, as shown in Fig.4 (red profile). This figure also shows the evolution of the estimated fault parameter (blue profile) which is computed from Eq.11. It can be seen that the value of \( \theta^* \) steps its way closer to the actual value of the fault parameter every sampling period (\( \Delta = 20 \) s). Once \( \theta^* \) has settled in accordance with Eqs.18-19, Eq.20, and Eqs.21-22, then fault accommodation is triggered and \( \hat{\theta} \) is adjusted per Eq.17 (green profile) to minimize performance losses while maintaining closed-loop stability. The impact of this fault accommodation strategy on closed-loop performance is shown in Fig.5 and Fig.6.

Fig.5 shows the accommodation result with \( \hat{\theta} \) being optimized to minimize the extended \( H_2 \)-norm. This shows the advantage of accommodating a performance-degrading fault, \( \theta = 6.9 \), occurring at \( t = 1000 \) followed by a disturbance of 10% of the steady state value of the \( O_2 \) flow rate. This figure shows the accommodated power output response in black (extended \( H_2 \)-norm = 3.23) and the response with no accommodation in red (extended \( H_2 \)-norm = 7.42). The minimal accommodation strategy, where
Fig. 4. Fault parameter estimation for a performance-degrading fault. θ̂ = θ∗ is not shown in the plot, but yields an extended $H_2$-norm of 3.78. It can be seen that the system with fault accommodation has a much smoother and faster response back to the steady state post the fault and disturbance events.

One can also see in Fig. 4 that the disturbances occurring at $t = 1400$ s do not trigger any change in $\dot{\theta}$ even though $\theta^*$ changes drastically. This is due to the precautions put in place from Eqs. 18-20. It should be noted again that these were obtained from the knowledge of the type of disturbances that would be affecting the system. Fig. 6 shows a similar accommodation result as in Fig. 5, but with a disturbance of 10% of the steady state value of the stack temperature introduced into the feed temperature following the fault event. This figure shows the accommodated power output response in black (extended $H_2$-norm = 0.012) and the response with no fault accommodation in red (extended $H_2$-norm = 0.028). The minimal accommodation strategy with $\dot{\theta} = \theta^*$ is not shown in the plot, but has an extended $H_2$-norm of 0.014. It can be seen that the system with accommodation has a much smoother and faster response back to the steady state value post the fault and disturbance events.

6. CONCLUDING REMARKS

When placed in the wider context of fault-tolerant control of DER networks, the focus of the current study has been on the detection and handling of faults solely at the local (i.e., the DER unit) level, without consideration of possible supervisory oversight or intervention. This emphasis is motivated by the fact that timely fault mitigation and protection measures at the local level are essential (when feasible) for preventing the effects of local faults from cascading through the system. However, in cases where local fault accommodation is no longer feasible, an intervention by a higher-level supervisor that takes into account the network-level dynamics becomes necessary to ensure fault-tolerance. The design of a supervisory fault-tolerant control system for DERs is the subject of ongoing work.

REFERENCES


