1. REVIEWER 1
1.1 Comment 1
The balance between detection delay and accuracy was controlled by the parameter $\sigma$, which may be dataset dependent or even sample dependent. A more principled way of choosing/updating it may affect the applicability of the model.

RESPONSE:
Developing such an adaptive scheme for choosing and updating $\sigma$ is a part of our ongoing work. We added a short description of it in the new paper version:

Different fault dynamics may require different optimal $\sigma$ values hence the choice of $\sigma$ ideally should be data dependent. One solution is to develop adaptive method within a robustness constraint, such that the adaptive procedure does not over emphasize short delays by sacrificing too much accuracy and also the maximum delay is also upper bounded by the minimum latency requirements.

1.2 Comment 2
Some theoretical analysis of algorithm convergence will be helpful for understanding its performance.

RESPONSE:
AdaBoost guarantees exponential progress towards minimization of the training error (10) with addition of each new weak classifier, as long as they classify the weighted training examples better than random guessing ($\alpha_{m+1} > 0$).

An appealing feature of Dem-Den Boost is that it retains the AdaBoost convergence properties. At any given time and for any given set of $m$ base classifiers $f_1, f_2, \ldots, f_m$, as long as the confidence parameters $\alpha_1, \alpha_2, \ldots, \alpha_m$ are positive and minimize $E_m^{\text{new}}$ (17), addition of new base classifier $f_{m+1}$ by minimizing (5) and calculation of $\alpha_{m+1}$ using (7) will lead towards minimization of $E_m^{\text{new}}$. This ensures the convergence of Dem-Den Boost.

However, more work has to be done to understand how minimizing (17) relates to minimizing the original cost function (10). And at what point it is not beneficial to continue with iterative optimization.

1.3 Comment 3
According to the experimental results, Dem-Den outperforms AdaBoost in both detection delay and true positive rate. It is understandable for the first result as Dem-Den was designed to minimize the delay. Why did it also beat AdaBoost in TPR with significant margin? Any theoretical justifications?

RESPONSE:
Dem-Den Boost outperforms AdaBoost in TPR because of the successful noise removal procedure (Den) that runs simultaneously with the detection delay minimization (Dem).

Clarification is provided in the new paper version.

1.4 Comment 3
Some comparisons with other commonly-used supervised algorithms, e.g. SVM, may strengthen the conclusions.

RESPONSE:
We have performed comparisons of SVM and Dem Den Boost on a smaller TEP data set. These results showed that at the same FPR level our algorithm achieves better DD, and with denoising, improves TPR on most of the faults as well. At this point, given the time constraint, we did not provide the SVM results for the data set used in this paper since best kernel parameters have to be found using cross-validation, which is costly.

We have added a comment related to this in the new paper version.

2. REVIEWER 2
2.1 Comment 1
Clarity and modifications are needed. In particular, the paragraph below figure 6 is very confusing and does not seem to be related to figure 6, even though that is the figure to which it refers.

RESPONSE:
2.2 Comment 2

Also, for Table 1, please indicate the statistical significance of the differences.

**RESPONSE:**
The bottom column of Table 1 now shows the statistical significance of Den and Dem-Den Boost TPR and DD improvement over AdaBoost in terms of paired t-test p-values.

2.3 Comment 3

Additionally, since the idea of the new weighting scheme is to reduce the detection delay, there should be some indication of how much the reduction in detection delay is for various faults rather than just accuracy of detection or true positive or false positive rates.

**RESPONSE:**
The delay reduction results are in the right three columns of Table 1.

3. REVIEWER 3

3.1 Comment 1

Figure 6, extend the range of the x-axis from 0 to 1, as is traditional. Different domains are interested in performance at different FP rates. Yes, I know things get weird in terms of DD at 0% FP - restrict the range of the y-axis.

**RESPONSE:**
In Figure 6 we report the FPR range of 1-5%, instead of the traditional 0-100% to better summarize our findings. Also, the FPR range 1-5% is the range of typical practical interest (high FPR would drastically reduce the usability of the system).

In the traditional FPR 0-100% range the figure is much less informative because DD becomes very high at FPR=0% and drops to 0 after FPR=15-25%.

3.2 Comment 2

A "within-subjects" ANOVA would be a good way to demonstrate statistical significance of results for TPR and DD changes. [Or honestly, just a couple of paired t-tests]

**RESPONSE:**
The bottom column of Table 1 now shows the statistical significance of Den and Dem-Den Boost TPR and DD improvement over AdaBoost in terms of paired t-test p-values.