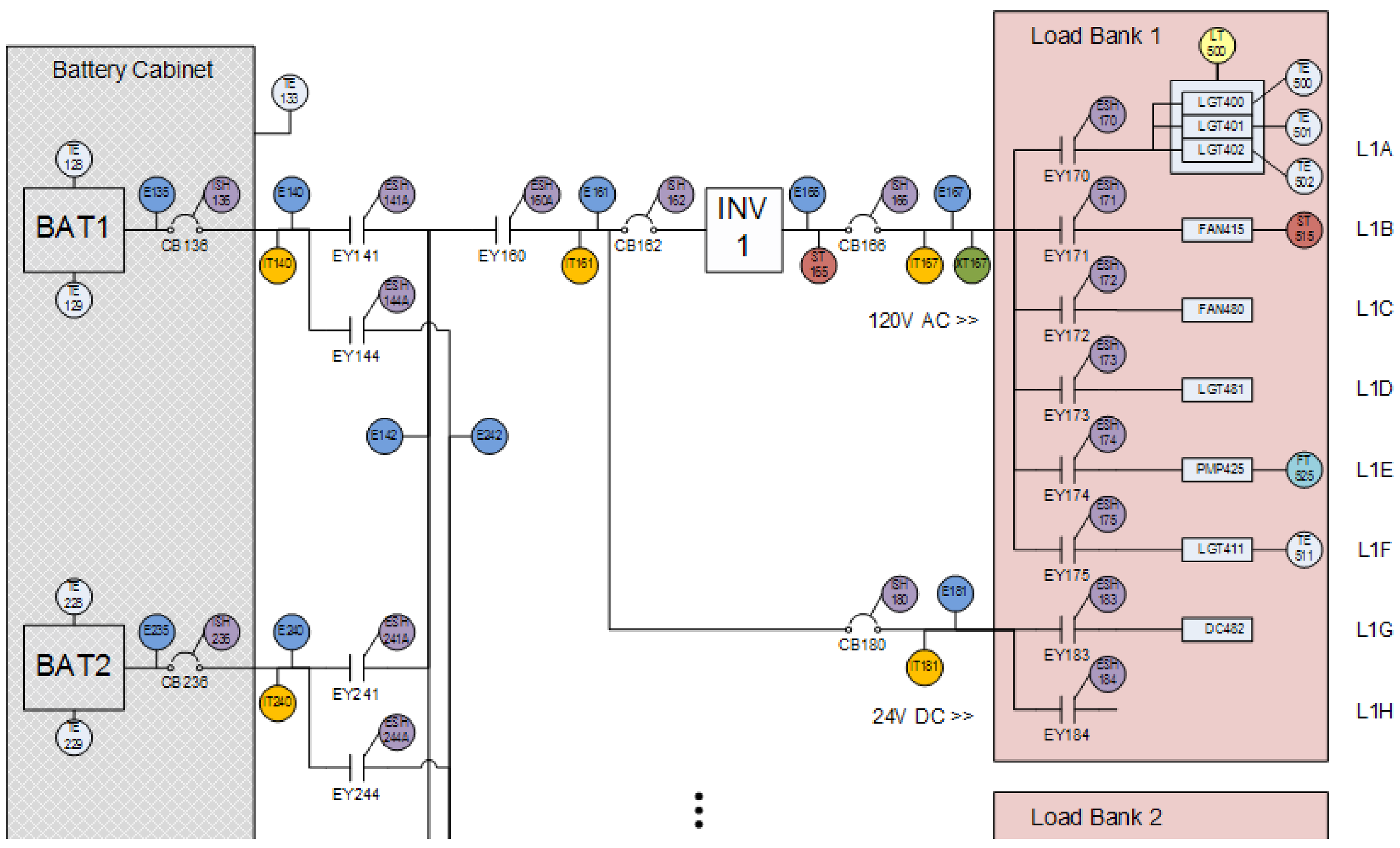
The FlexDx Diagnosis Framework Applied to ADAPT

Erik Frisk, Mattias Krysander, and Fredrik Heintz
{frisk,matkr}@isy.liu.se, frehe@ida.liu.se

Summary

To find new challenges for the FlexDx diagnosis framework we are applying it to the Advanced Diagnostics and Prognostics Testbed (ADAPT). It is a real physical system built by NASA consisting of more than 100 sensors and components. No model is provided, instead it has to be constructed from the available data. We present initial modeling, diagnosability, and residual generation results.

ADAPT



FlexDx

FlexDx provides real-time multiple fault model-based diagnosis of dynamical systems. Computational efficiency is achieve through adaptive reconfiguration, by exploiting the fact that only a subset of all possible tests are required to detect that some fault has occurred. Tests are thresholded residuals. If a fault is detected, the diagnosis is incrementally refined by iteratively running more tests according to the following procedure:

1. Compute the next set of tests based on the current diagnoses.
2. Compute the initial state of the selected tests.
3. Run the tests until an alarm is triggered.
4. Compute the new diagnoses from the test results, go to step 1.

FlexDx requires automatic model-based methods for generating residuals, selecting tests, and computing diagnoses. Residuals can either be computed in advance or on demand. Diagnoses are computed using a conflict-based approach by a minimal hitting set algorithm. Test selection is also based on conflicts.

FlexDx uses the DyKnow knowledge processing middleware to practically deal with the issues introduced by reconfigurability.

Modelling

A model of the system has been developed, represented in Mathematica. The model consists of static and dynamic equations, both linear and nonlinear, as well as logical expressions. Below are the models for the batteries and the inverters.

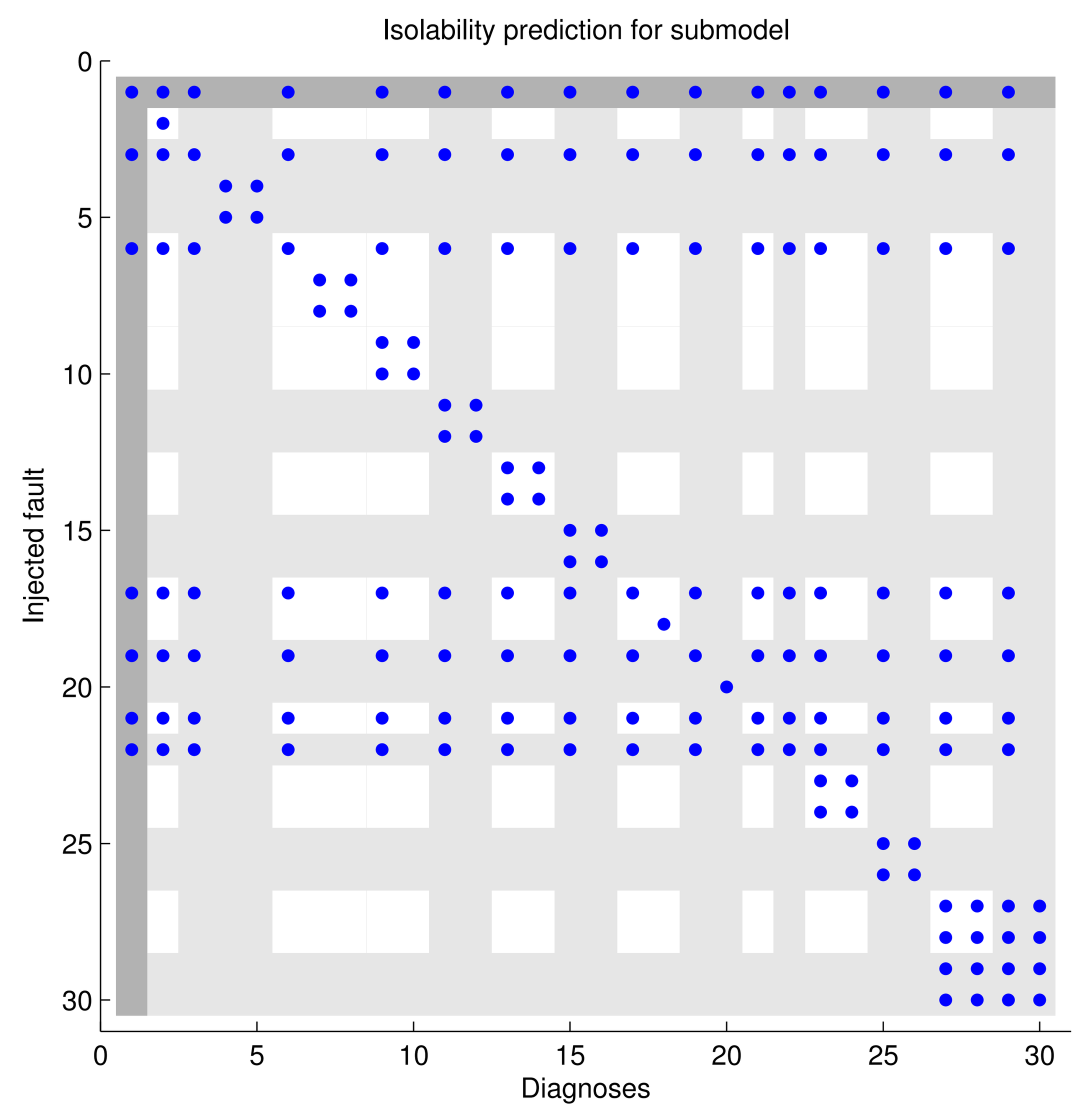
$$\begin{aligned} v &= U_0 - RI \\ OK(Bat) &\rightarrow R = R_0 \\ Deg(Bat) &\rightarrow R' = 0 \\ Deg(Bat) &\rightarrow T' = \alpha(T_{amb} - T) + \beta RI^2 \end{aligned}$$

$$\begin{aligned} & OK(Inv) \rightarrow V_{dc} > 22 \Leftrightarrow V_{ac} > 120 \\ & FO(Inv) \rightarrow V_{dc} > 22 \\ & FO(Inv) \rightarrow V_{ac} < 120 \\ & OK(Inv) \rightarrow V_{ac} I_{ac} = V_{dc} I_{dc} + P_0 \\ & OK(Inv) \rightarrow (V_{dc} > 22 \rightarrow f = 60) \end{aligned}$$

Diagnosability Analysis

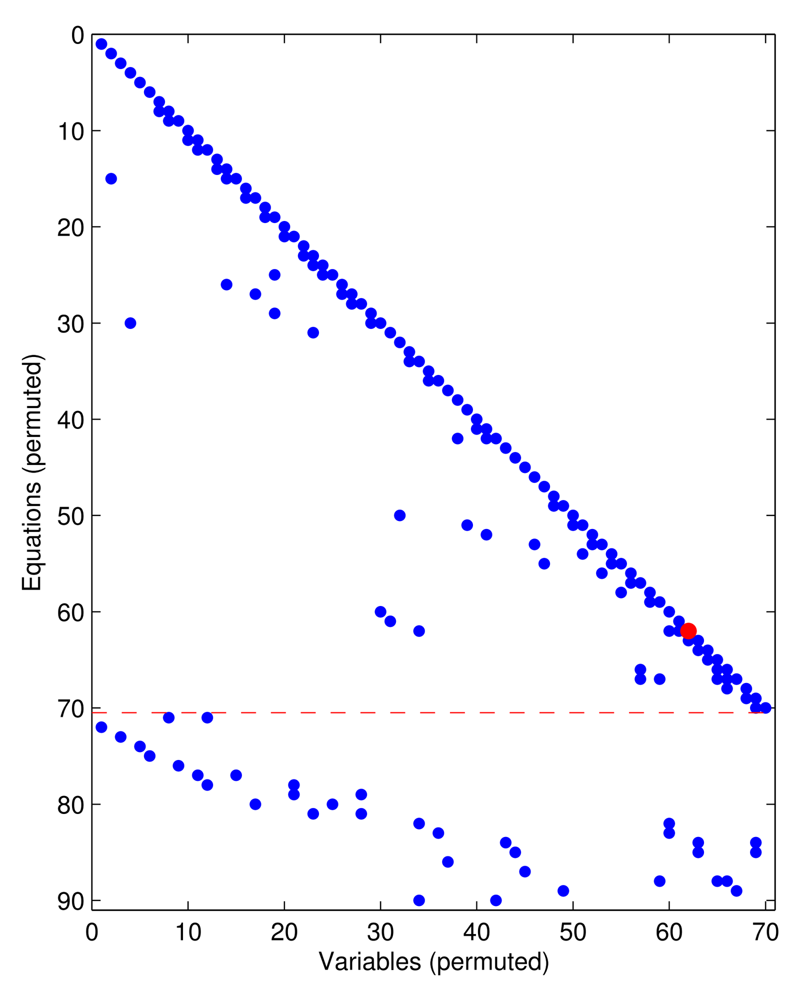
Diagnosability analysis can be automatically performed on the Mathematica model.

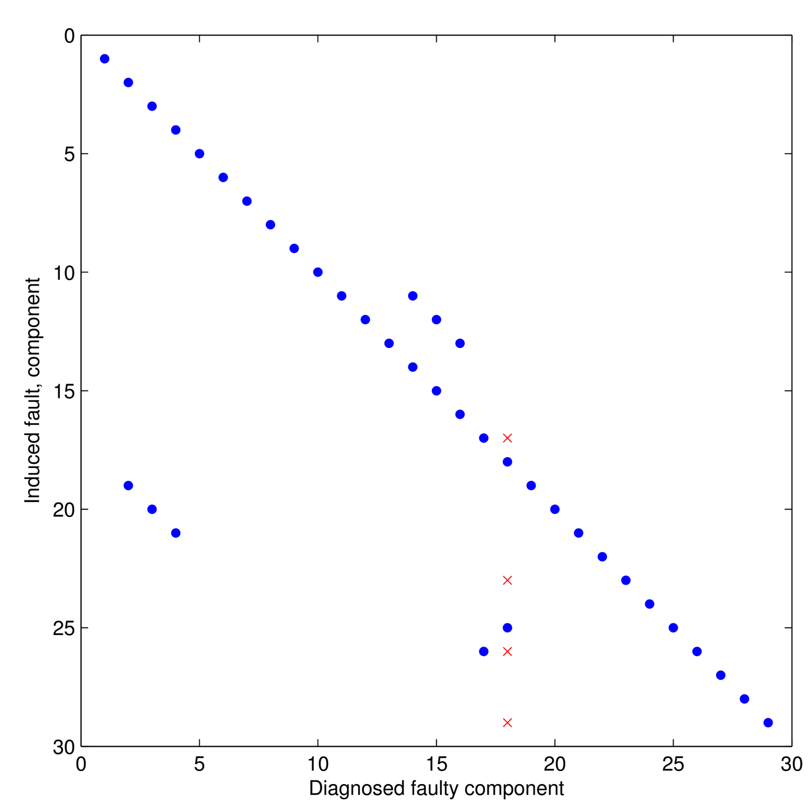
A single-fault analysis on the system can be illustrated using a matrix with induced faults on one axis and diagnosed faults on the other. The matrix here illustrates diagnosability properties of the system from component Bat1 to Inv1.



Residual Generation

Due to the high degree of redundancy, it is infeasible to precompute all possible residual generators and filters must be automatically designed on-line. An interesting class is completely diagonalizable residual generators. The figure below illustrates fully diagonalized residual generators for the nominal model. The right figure illustrates component isolability performance for all residual generators compared to only utilizing diagonalizable residual generators.





Supported by grants from the SSF Strategic Research Center MOVIII and the VR Linnaeus Center CADICS.