FlexDx: A Reconfigurable Diagnosis Framework

Fredrik Heintz IDA (frehe@ida.liu.se)
Mattias Krysander, Jacob Roll, Erik Frisk ISY

Introduction

Detecting and isolating multiple faults in a dynamic process is a computationally intense task which typically consists of computing a set of tests and then computing the diagnoses based on the test results. FlexDx is a reconfigurable diagnosis framework which reduces the computational burden by only running the tests that are currently needed. The method selects tests such that the isolation performance of the diagnostic system is maintained. Special attention is given to issues introduced by a reconfigurable diagnosis framework which has to:

- add and remove tests dynamically,
- perform tests on partially historic data, and
- combine synchronous/asynchronous processing.

The FlexDx Framework

FlexDx incrementally refines the set of diagnoses according to the following procedure:

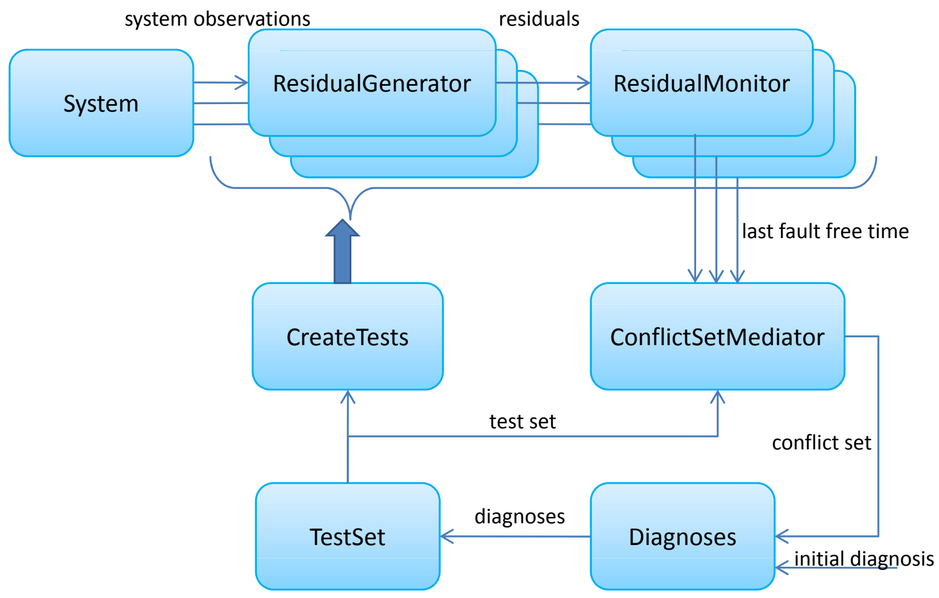
1. Initiate the set of diagnoses.
2. From the diagnoses compute the new set of tests.
3. Compute the initial state of the selected tests.
4. Run the tests until an alarm is triggered.
5. Compute the new diagnoses, go to step 2.

DyKnow

FlexDx is implemented using DyKnow, a stream-based knowledge processing middleware framework. Knowledge processing applications process asynchronous *streams* described by declarative *policies*. DyKnow provides four types of knowledge processes:

- *Primitive processes* provide an interface to outside information sources such as sensors and databases.
- *Refinement processes* take streams as input and produce refined and abstracted streams.
- *Configuration processes* take streams as input, add and remove streams and processes, but produce no output stream. Enable reconfiguration.
- *Mediation processes* generate streams by selecting or collecting information from other streams.

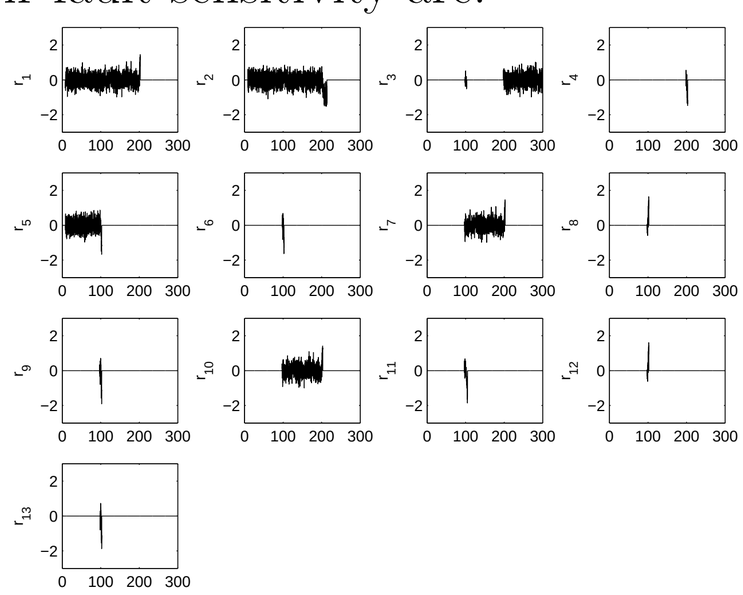
A FlexDx Instantiation



Test Reconfiguration

To illustrate the FlexDx framework consider a system modelled by 3 linear differential equations and 3 sensor equations. There is one possible single fault for each of the six equations. A set of 13 residuals are used and their fault sensitivity are:

	f_1	f_2	f_3	f_4	f_5	f_6
r_1				X	X	X
r_2		X	X		X	X
r_3		X	X	X		X
r_4		X	X	X	X	
r_5	X		X		X	X
r_6	X		X	X		X
r_7	X		X	X	X	
r_8	X	X			X	X
r_9	X	X		X		X
r_{10}	X	X		X	X	
r_{11}	X	X	X			X
r_{12}	X	X	X		X	
r_{13}	X	X	X	X		



To show how the diagnosis system is reconfigured during a fault transient, the fault f_1 at $t = 100$ was injected in a simulated scenario. The result is:

Minimal Diagnoses	Active Tests	t_{fault}	t_{alarm}
NF	1, 2, 5	0	102.6
1, 3, 5, 6	1, 3, 10, 13	98.9	102.7
1, 3, 25, 26, 45, 46	1, 2, 6, 7, 8, 11, 12	98.9	102.2
1, 23, 25, 26, 35, 36, 45	1, 2, 6, 7, 9, 10, 11	98.9	102.3
1, 23, 26, 35, 36, 45	1, 2, 7, 9, 10, 11	98.9	102.6
1, 23, 26, 36, 45	1, 2, 7, 10, 11	98.9	105.2
1, 23, 26, 36, 245, 345, 456	1, 2, 7, 10	100.6	–

By comparing the number of residuals computed when running all tests at all times with the number computed by FlexDx, a 78.3% reduction is obtained for the simulated scenario.

[1] F. Heintz, M. Krysander, J. Roll, and E. Frisk. FlexDx: A reconfigurable diagnosis framework. In *Submitted to International Workshop on Principles of Diagnosis*, 2008.

[2] M. Krysander, F. Heintz, J. Roll, and E. Frisk. Dynamic test selection for reconfigurable diagnosis. In *Submitted to Conference on Decision and Control*, 2008.