Multi-feature testing in telecommunications

Roberto Rossi	4C
S. Armagan Tarim	4C
Brahim Hnich	Izmir Univ.
Steven Prestwich	UCC

Why testing is important?

- Telecomunication devices are often assembled using different components
- Companies which operates in the telecomunication market usually buy components from different vendors and assemble complex and customized circuit packs
- The reliability of a complex system tends to decrease as the number of its elementary parts increases
- Testing is essential in order to control the overall quality of the final product
  - warranty issues!

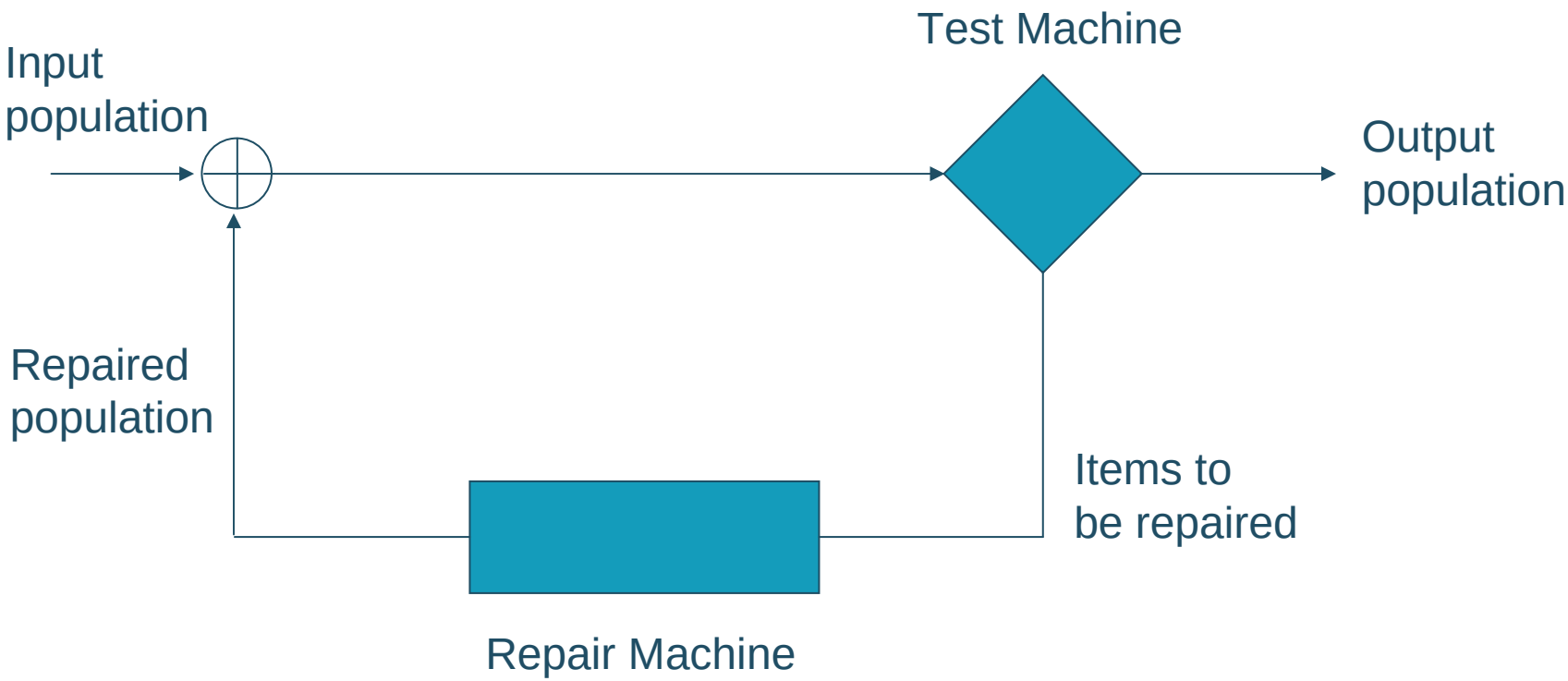
Why testing is important?

- A simple example from Lucent

Complete XXXXXX Pack: XXXXXX	*Pop. Mix.*
Feature 1	99
Feature 2	95
Feature 3	95
Feature 4	96
Feature 5	96
Feature 6	96
Feature 7	99
Feature 8	99
Feature 9	**98**
Feature 10	**98**
Feature 11	***90***
Feature 12	***98***
Feature 13	94
Feature 14	94
Feature 15	94
Feature 16	90

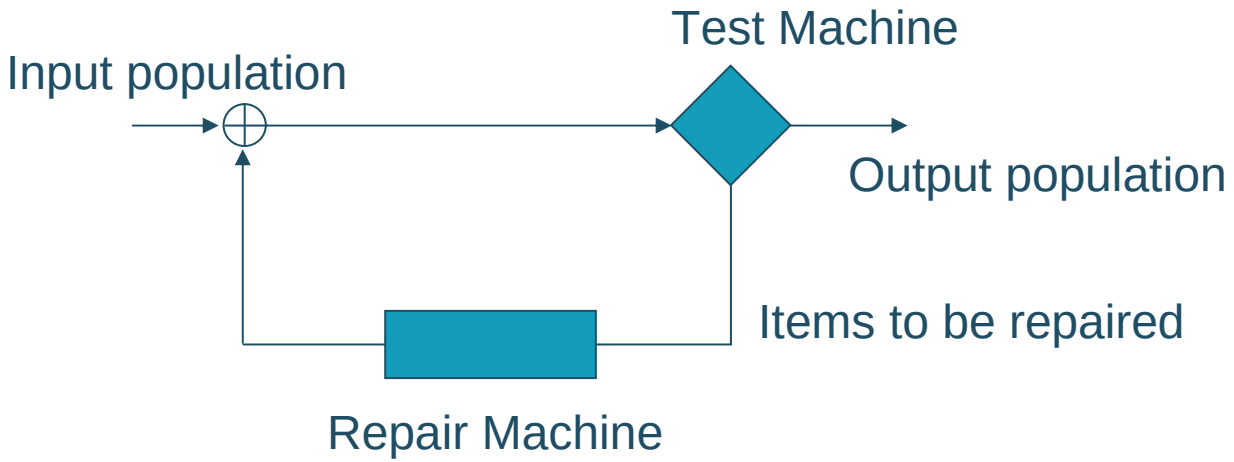
Only one pack over two would work if no test was performed after the assembling process

A simple T/R network



A simple T/R network

- Inputs:
  - Test Machine parameters
  - Repair Machine parameters
  - Input population good/bad item distribution for each feature



Aims of Test Strategy Optimization

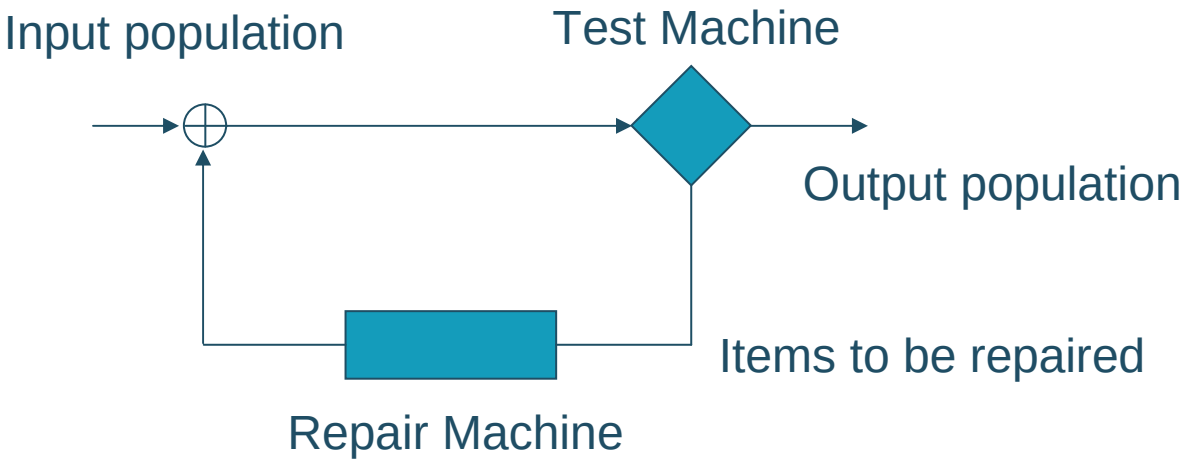
- Modeling
  - Analyze the throughput for a given T/R network and its internal steady state
- Optimisation
  - Minimise costs related to T/R machine activity
    - Running costs
    - Set up costs
  - Control the “Field Return Rate”, that is the number of bad items sold because considered good since tests are failed

A simple T/R network
Exact Approach

- Bell Labs US proposed an exact approach to model the problem of computing flows in the given test/repair network:

  E. Fisher et al., *"Economic Modeling of Global Test Strategy I: Mathematical Models", Bell Labs US*

Population (2 feat.):			
81%	9%	9%	1%
1	1	0	0
1	0	1	0

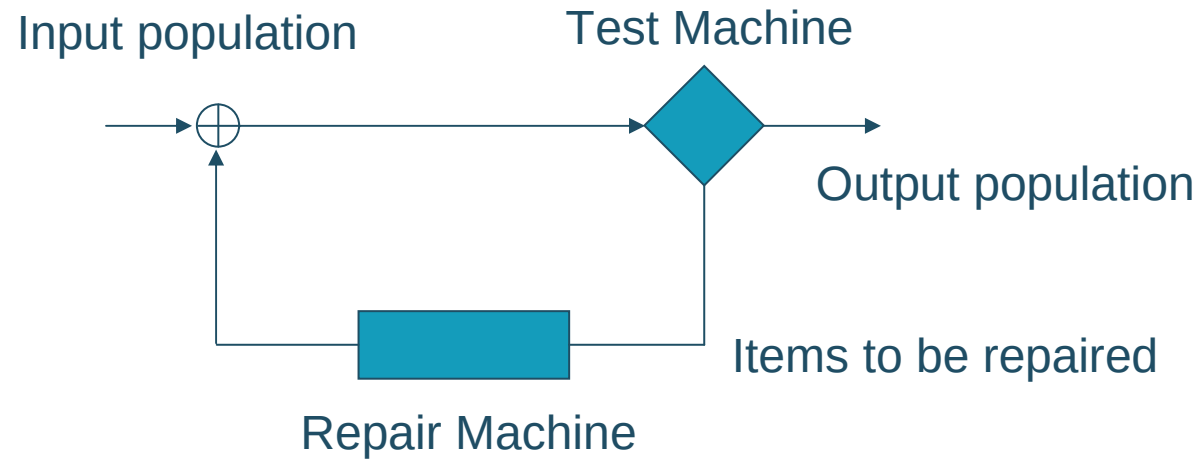


A simple T/R network
Exact Approach

- Bell Labs US proposed an exact approach to model the problem of computing flows in the given test/repair network:

  Exponential number of variables and linear equations to represent flows in the network!

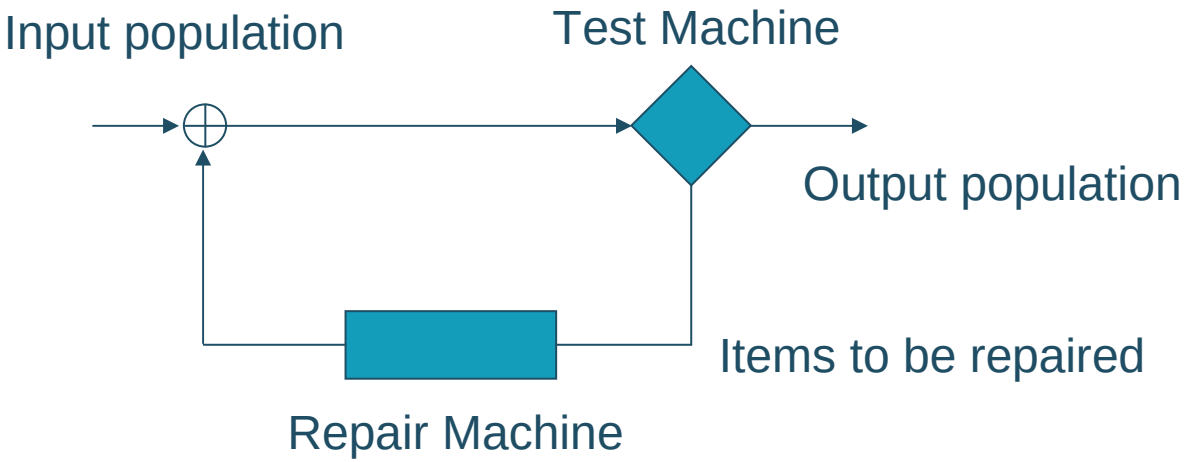
Population (2 feat.):			
81%	9%	9%	1%
1	1	0	0
1	0	1	0



A simple T/R network
Exact Approach

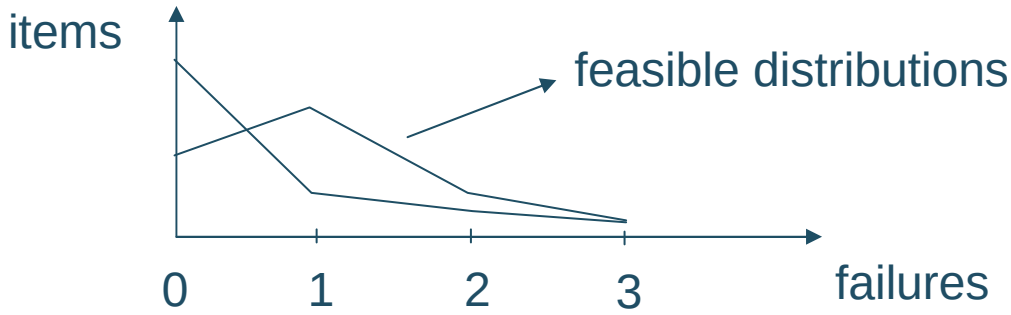
- A typical circuit pack can contain **up to 30 features**, therefore such an approach cannot be applied to analyze flows of good/bad items in the test/repair network.
  - An approximate method is needed

Population (2 feat.):			
81%	9%	9%	1%
1	1	0	0
1	0	1	0



A simple T/R network Approximate Approach

- Since we are performing tests on a population of items made up of features that are independent, the initial distribution of failures follows a binomial trend
  - **Intuitively** this means that the probability of having items with a few failures must be higher than that of having items with many failures



- We assume that such a distribution is preserved all over the network
  - strong assumption ☹
  - good empirical results ☺

A simple T/R network Approximate Approach

- Our assumption is reasonable when
  - the population has independent features being tested, each with a low probability to result bad
  - The test(repair) machine is a “good” machine, therefore TP(BG) and TF(GG) must be high values.

- The population is therefore represented as

Population (2 feat.) - exact:

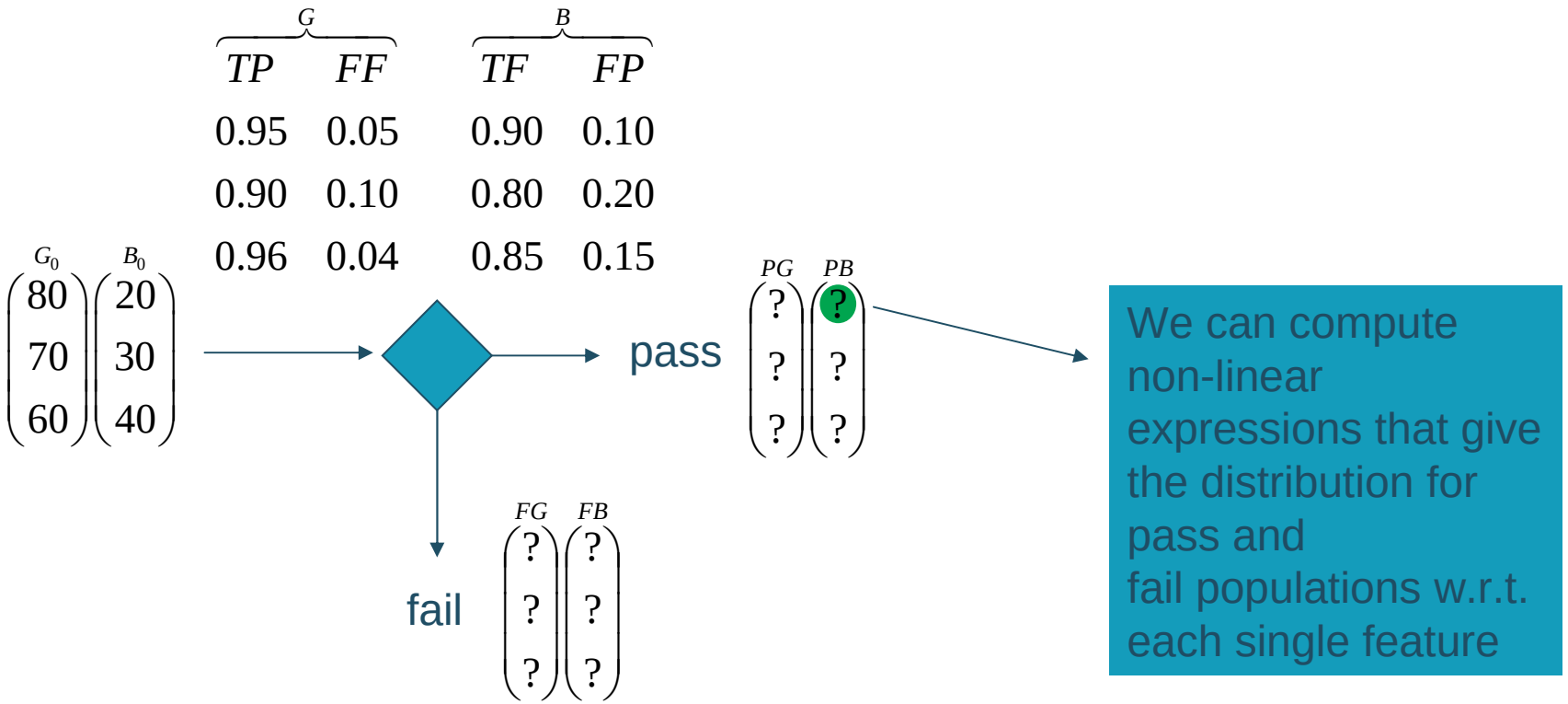
	81%	9%	9%	1%
F1	G	G	B	B
F2	G	B	G	B

Population (2 feat.) - approx:

G	B
90%	10%
90%	10%

A simple T/R network Approximate Approach

- Modeling a Test Machine:

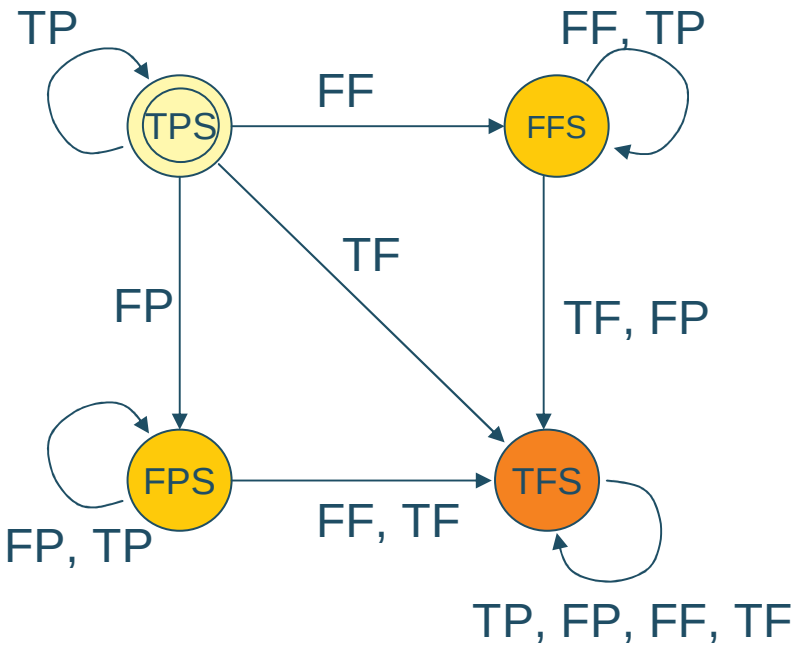


We can compute non-linear expressions that give the distribution for pass and fail populations w.r.t. each single feature

A simple T/R network

- Modeling a Test Machine:

  We can model a test machine and compute pass/fail probabilities for each single feature in linear time and linear space using the following finite state automaton:



Each feature pass/fail probability can be computed considering all the possible paths that lead from the initial state to the desired pass/fail situation for an item.

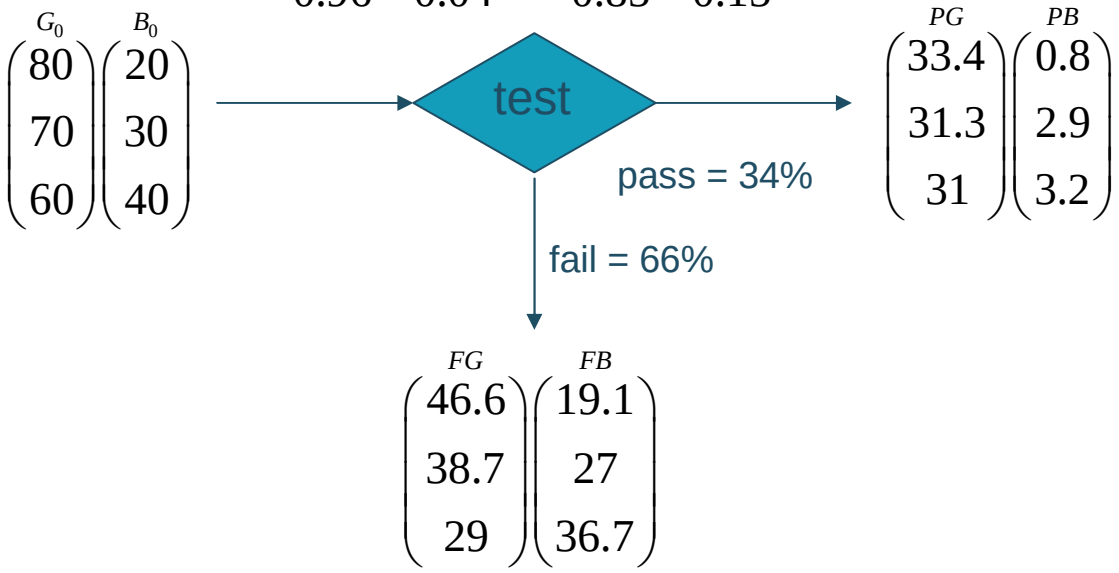
Example:

$$PG[x] = TP[x] \prod_{i=1, i \neq x}^{f} (TP[i] + FP[i])$$

A simple T/R network

- Modeling a Test Machine:

$\overbrace{\quad}^{G}$		$\overbrace{\quad}^{B}$	
TP	FF	TF	FP
0.95	0.05	0.90	0.10
0.90	0.10	0.80	0.20
0.96	0.04	0.85	0.15

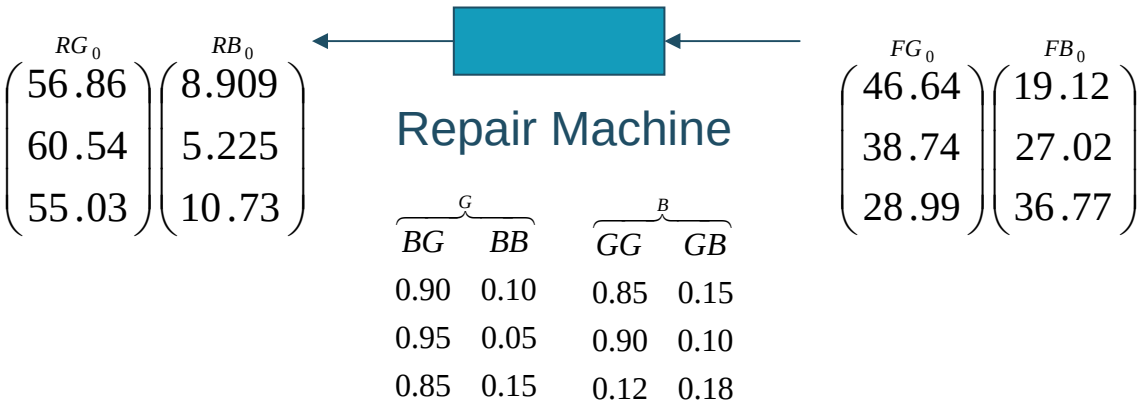


A simple T/R network

- Modeling a Repair Machine:

  since a repair machine is not a branching point its straightforward to compute output population when the input one is given.

Example: $RG_0[0] = FG_0[0] \cdot GG[0] + FB_0[0] \cdot BG[0]$

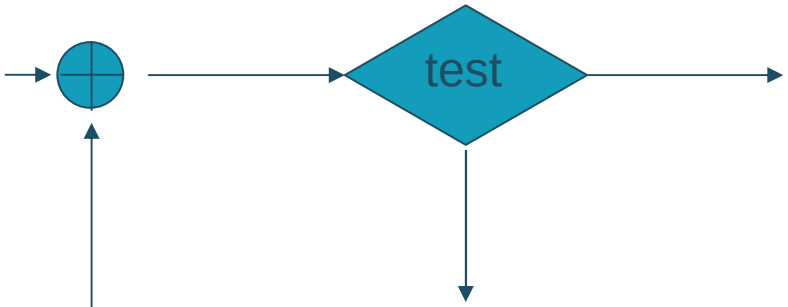


A simple T/R network

Initial state

$$\begin{matrix} G_0 \\ \begin{pmatrix} 80 \\ 70 \\ 60 \end{pmatrix} \end{matrix} \quad \begin{matrix} B_0 \\ \begin{pmatrix} 20 \\ 30 \\ 40 \end{pmatrix} \end{matrix}$$

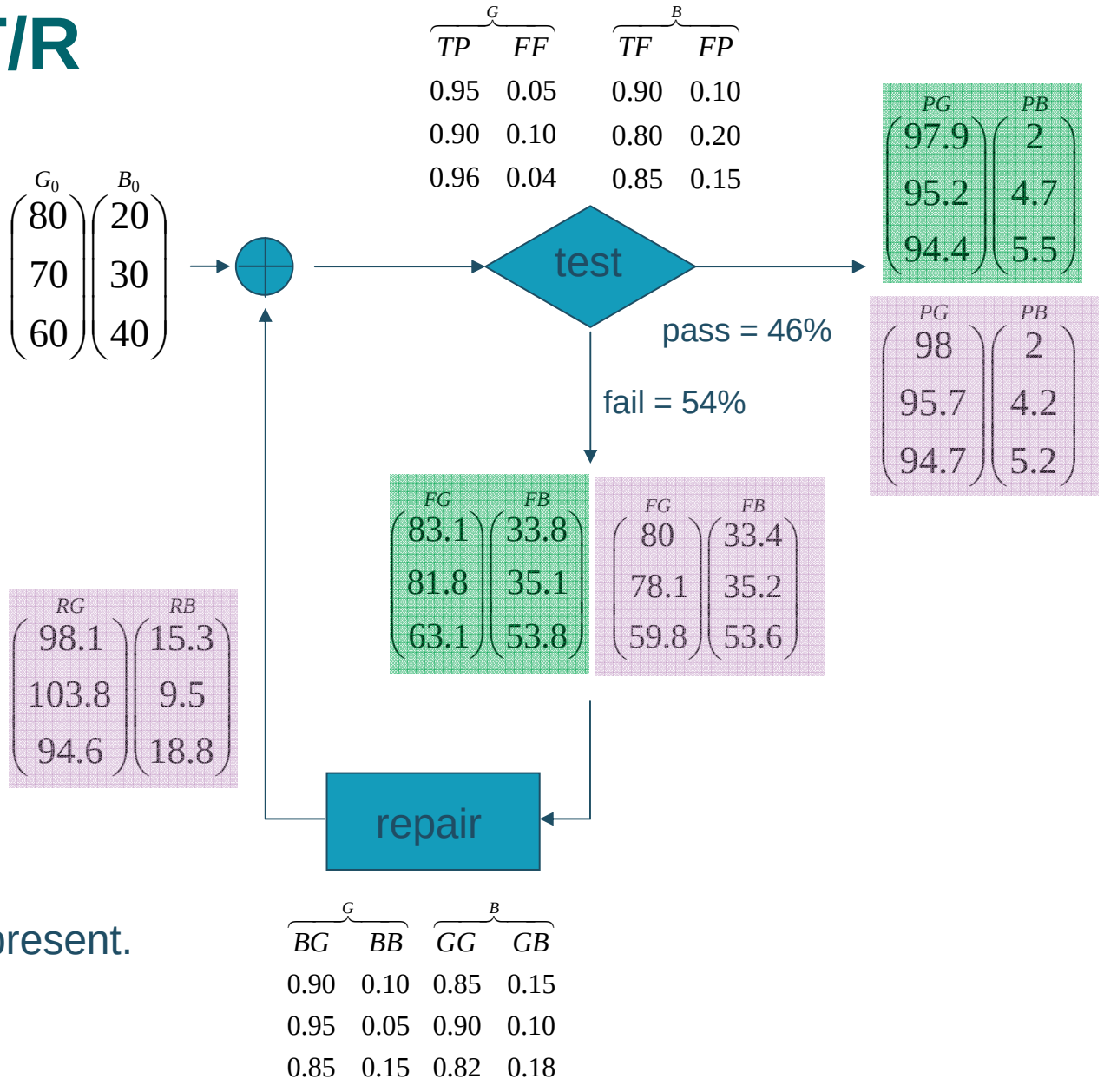
$\overbrace{TP}^{G}$	FF	$\overbrace{TF}^{B}$	FP
0.95	0.05	0.90	0.10
0.90	0.10	0.80	0.20
0.96	0.04	0.85	0.15



$\overbrace{BG}^{G}$	BB	$\overbrace{GG}^{B}$	GB
0.90	0.10	0.85	0.15
0.95	0.05	0.90	0.10
0.85	0.15	0.12	0.18

A simple T/R network

Steady state



Approximate represent.

Exact represent.

Experimental results

- We performed a broad set of tests to analyze the effectiveness of our approximation when it is compared to the exact method proposed by Bell Labs US
  - Population mix from 50% of good items to 100% (step 5%)
  - Test (Repair) Machine quality from 50% to 100% of correct assesments (step 5%)
  - Circuit packs comprising 3,5 and 7 features being tested
    - In the single feature case the two approaches are equivalent

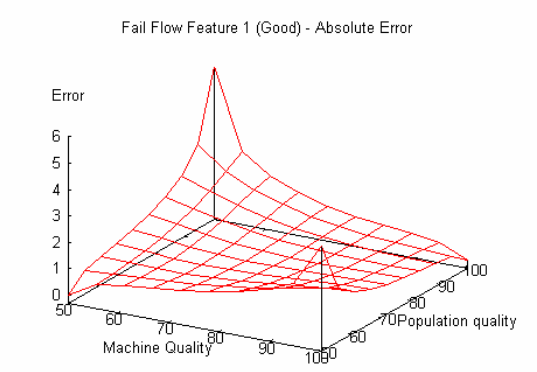
Experimental results

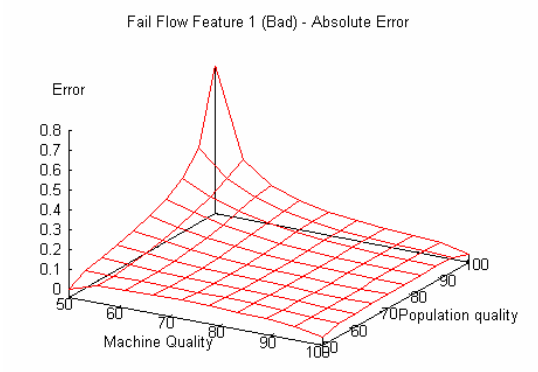
- For the single test/single repair case we observed the following errors:

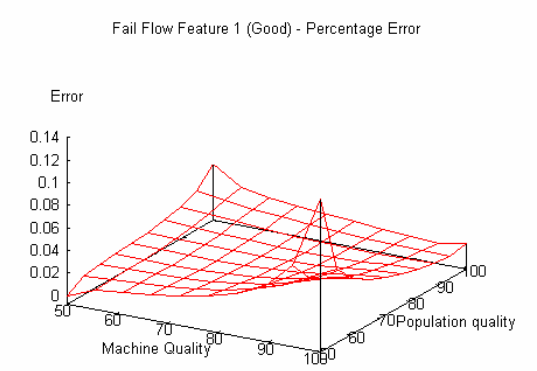
in percentage on the exact flow size computed with the exponential representation

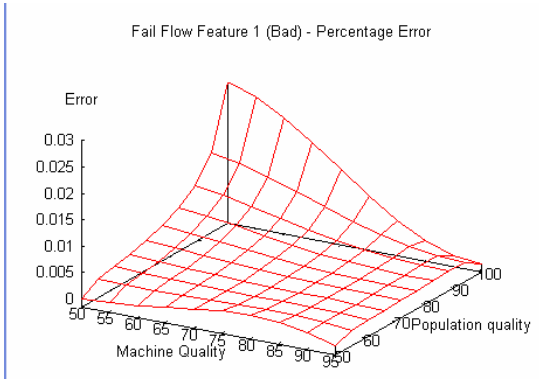
	Pass (G)	Pass(B)	Fail (G)	Fail (B)	Rep (G)	Rep (B)	Overall (G)	Overall (B)
Mean - 3	0,06	5,07	1,66	0,37	1,29	1,29	1,00	2,24
Var -3	0,00	0,33	0,05	0,00	0,02	0,02	0,03	0,16
Mean - 5	0,03	6,11	1,84	0,70	1,63	1,63	1,17	2,82
Var - 5	0,00	0,52	0,05	0,01	0,04	0,04	0,04	0,24
Mean - 7	0,02	6,51	1,90	0,92	1,77	1,77	1,23	3,07
Var - 7	0,00	0,64	0,05	0,02	0,04	0,04	0,04	0,30

Experimental results – 3 feat.

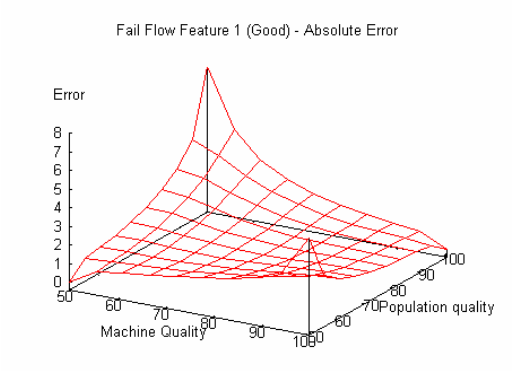


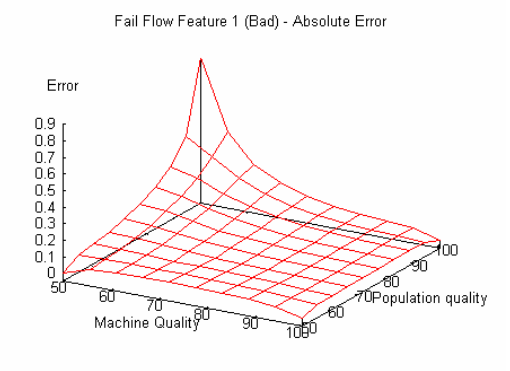


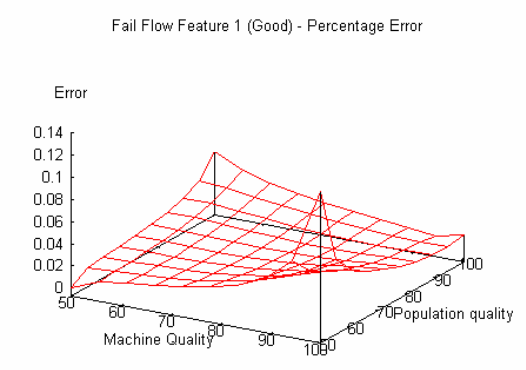


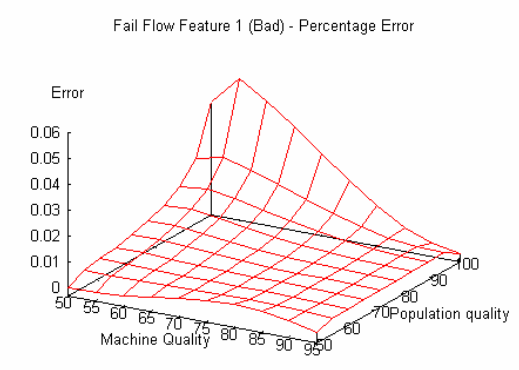


Experimental results – 5 feat.

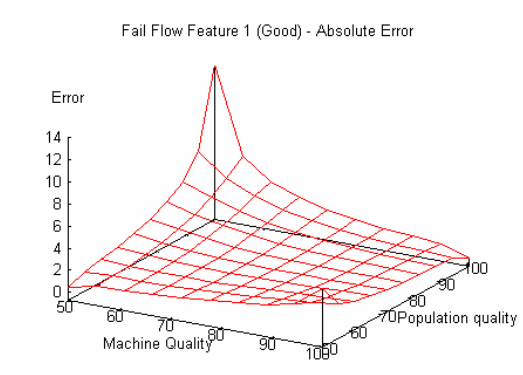


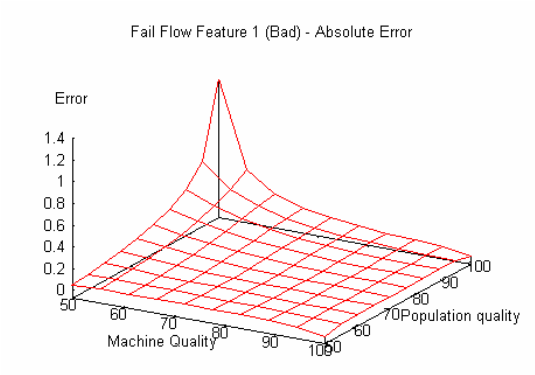


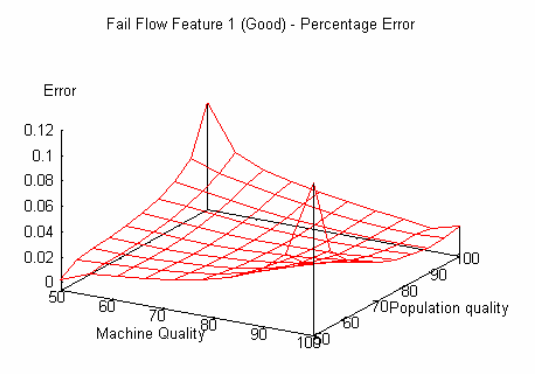


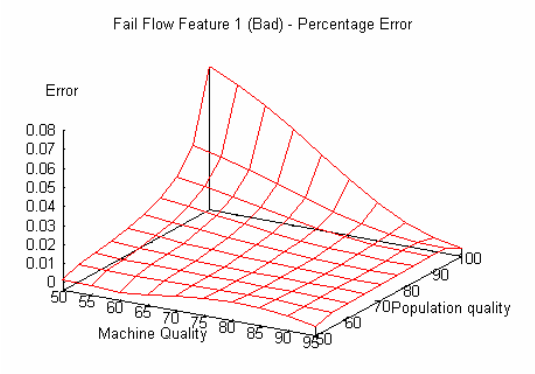


Experimental results – 7 feat.

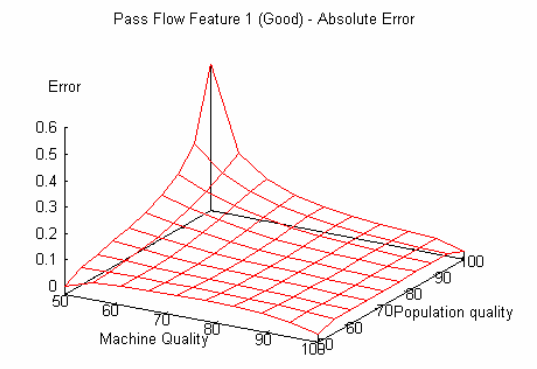


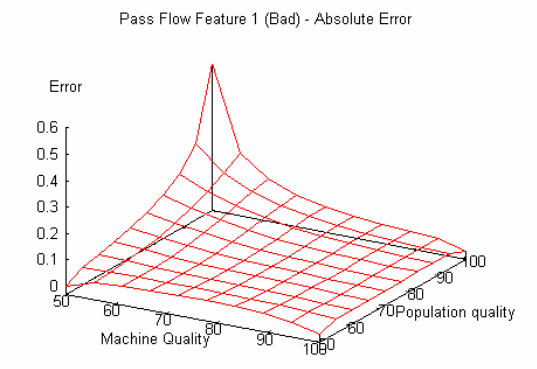


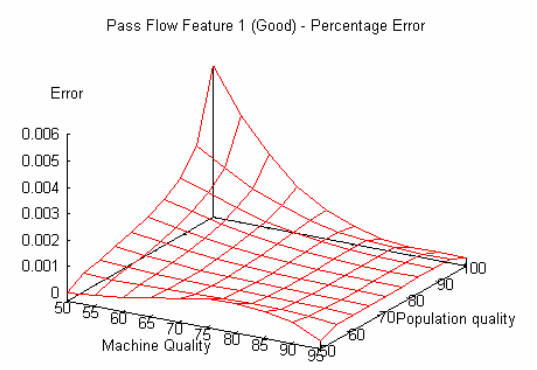


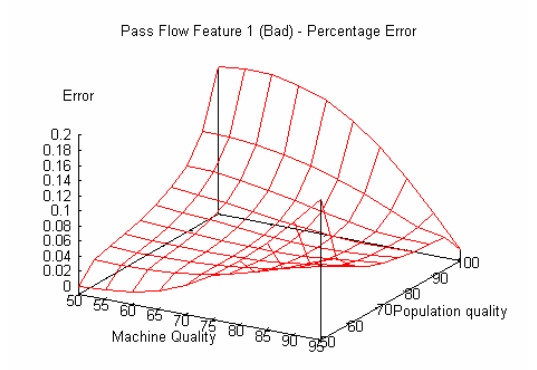


Experimental results – 3 feat.

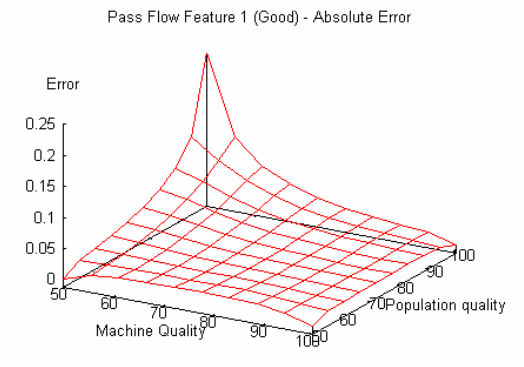


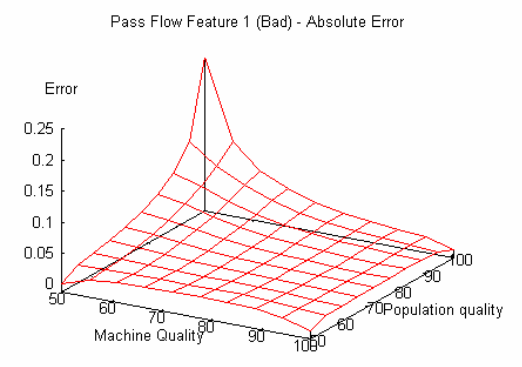


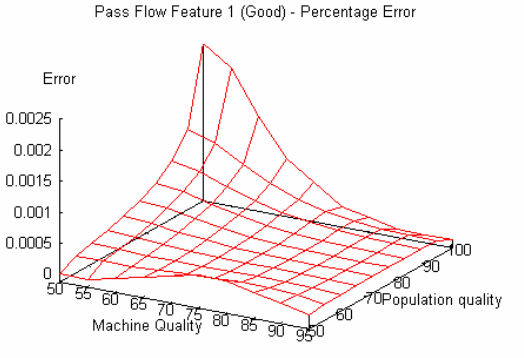


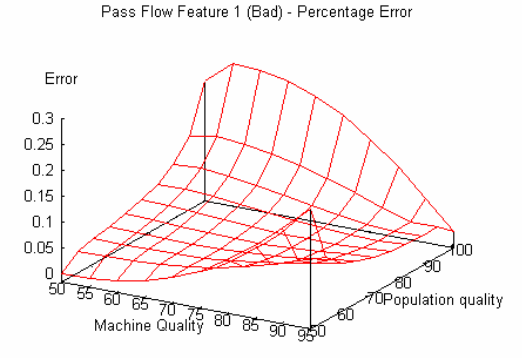


Experimental results – 5 feat.

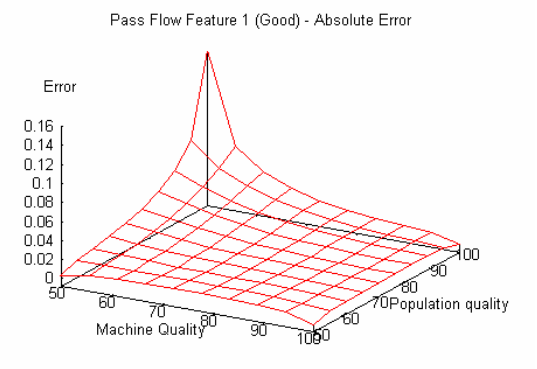


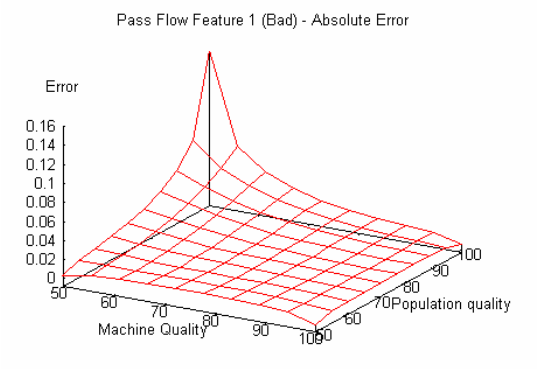


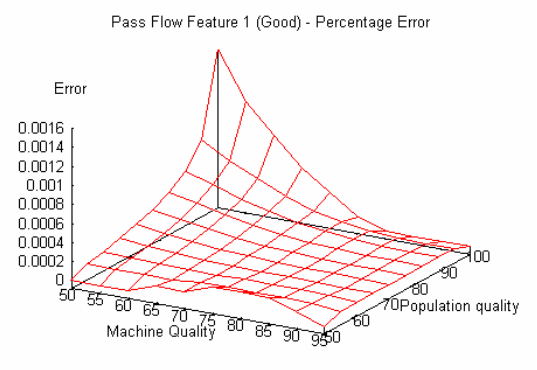


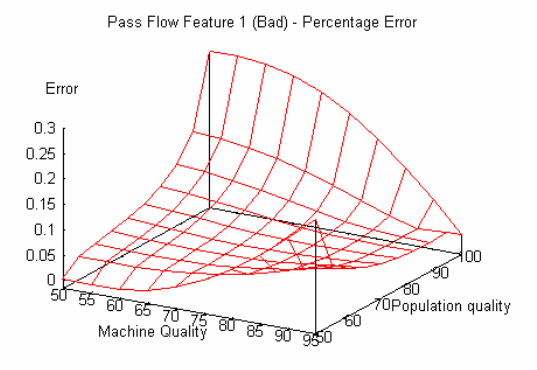


Experimental results – 7 feat.

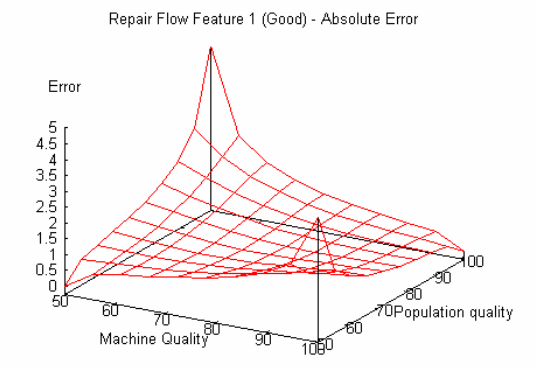


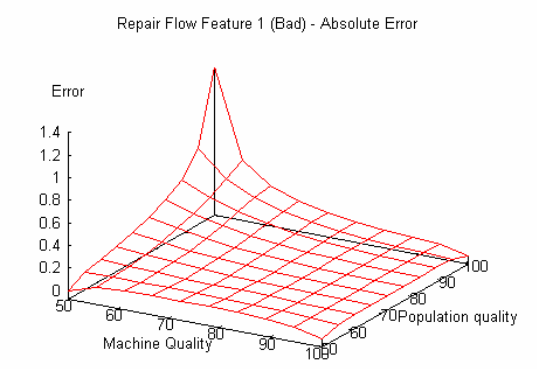


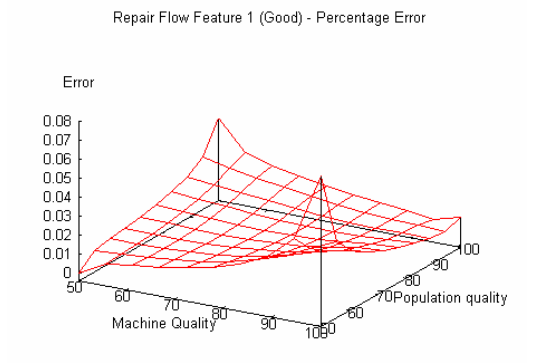


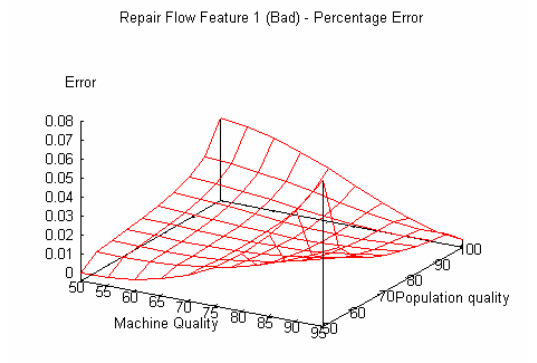


Experimental results – 3 feat.

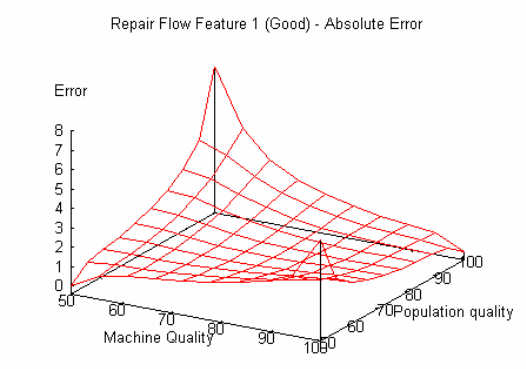


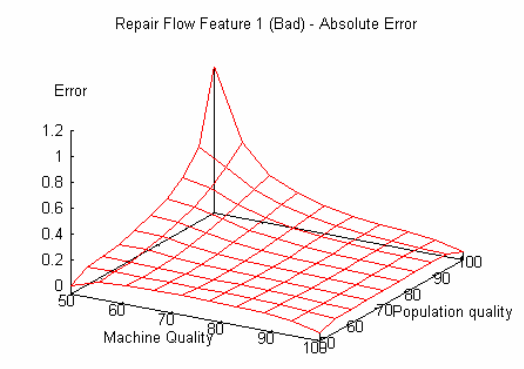


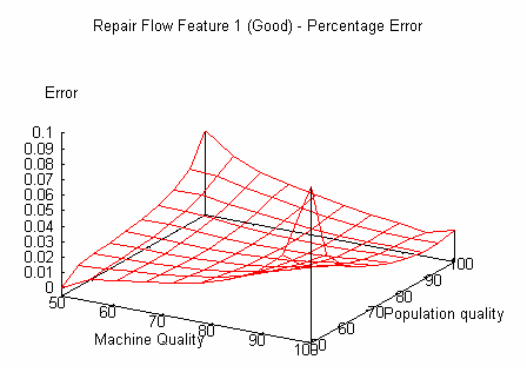


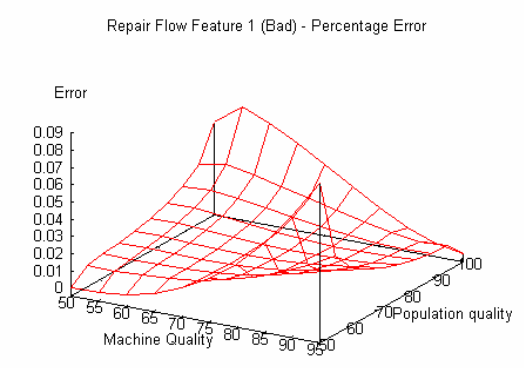


Experimental results – 5 feat.

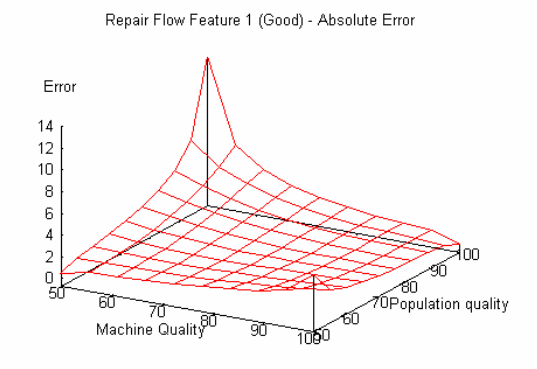


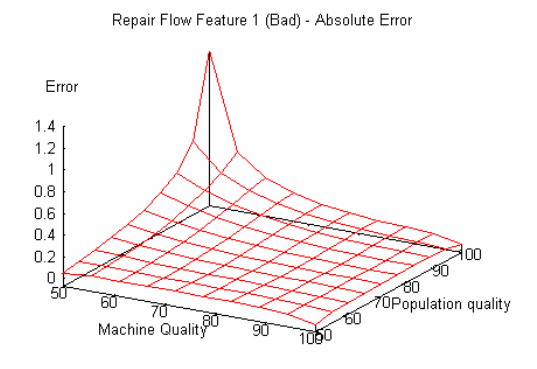


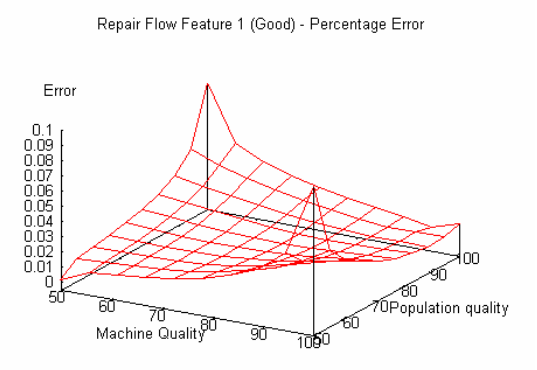


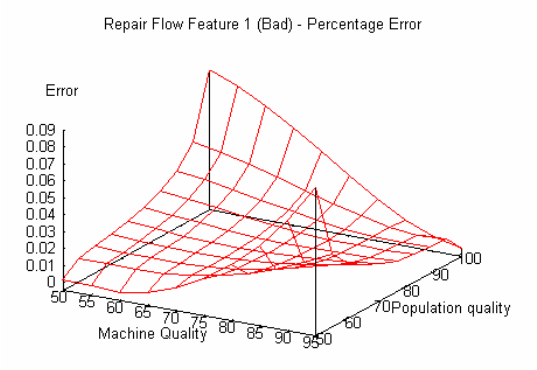


Experimental results – 7 feat.









Conclusions

- Our approach proved to be a valid approximation for modeling flows in simple test/repair networks
- Our uniformity assumption for the distribution of failures in the population being tested doesn’t produce relevant errors in the proposed test bed
- Since we just require a number of equation and variables that is linear in the number of features being tested, It is possible now to model test/repair networks for circuit packs composed by many independent features

Extensions…

- Optimisation of:
  - flows given parameters
  - parameters given flows
- Optimisation of cost when a stated FRR is required and a flow of items is given
  - Type of test machine used
  - Type of repair machine used
- Modeling and Optimisation of complex network structures with more test/repair stages