Estimating Production Test Properties from Test Measurement Data Using Gaussian Mixtures

Simon P. Wilson[*], Suresh Goyal[†]

Abstract

A complex sequence of tests on components and the system is a part of many manufacturing processes. Statistical imperfect test and repair models can be used to derive the properties of such test sequences but require model parameters to be specified. We describe a technique for estimating such parameters from typical data that are available from past testing. A Gaussian mixture model is used to model the wide variety of statistical properties of test data, including outliers, multi-modality and skewness, from which test properties are derived. Model fitting is through a Bayesian approach, implented by Markov chain Monte Carlo.

Keywords: imperfect repair, imperfect test, Gaussian mixture model, optimal test, parameter estimation.

[*] Author for Correspondence. Department of Statistics and Centre for Telecommunications Value-Chain Research, Trinity College, Dublin 2, Ireland. E-mail simon.wilson@tcd.ie

[†] Bell Labs Ireland, Blanchardstown, Dublin 15, Ireland

1 INTRODUCTION

Many manufacturing processes involve a complex sequence of tests on components and sub-systems that are done prior to shipping the product to the customer. Which tests are conducted, the action to be taken when a unit fails a test — such as scrap, repair and retest or ignore — can have a dramatic effect on field reliability. Optimal test sequencing, that trades off test cost with product reliability, therefore can be an important element in the production design process, particularly in industries that manufacture complicated and expensive products that must be reliable and have high availability; it is no surprise that the problem has received most attention in the literature on integrated circuit design and production (Grout, 2005; Sun, 2008). Another example, and the motivation for the work presented in this paper, is to optimizing production testing in the manufacture of telecommunications network components, where testing may involve a complex schedule of tests and it is recognized that large cost savings arise from well-planned testing (Fisher et al., 2007a,b).

The typical test optimization process begins with specification of the parameters of each test, such as the product quality (e.g. proportion of units that should pass the test) and the test accuracy (e.g. the rate of false positives and negatives). Other factors, such as the time and cost of the tests and constraints on the schedule, like orderings on tests and when integration of components or subsystems must take place, are also specified. Using this information, a test sequence that optimizes a metric of interest, such as cost, production time, coverage (proportion of potential faults that could be detected) or achieved system reliability, is derived.

In this paper we do not look at the issue of optimizing the test schedule given test parameters, but consider a solution to the problem of how the test parameters are defined in the first place. Thus our work can be thought of as providing the input into the test optimization process. It is potentially important to do this well; it is recognized that the optimal test sequence can be highly sensitive to test parameter values (Dick et al., 1994). Specifically, we propose a solution for how data from production test results — that is to say the measurements that are recorded when the test is conducted — may be used for inference of the test model parameters of a particular imperfect test and repair model (Fisher et al., 2007a). This model is certainly applicable to most of the tests that are done in telecommuncations equipment manufacture, and we contend it should be applicable to testing in other similar manufacturing industries where test measurement results are routinely stored. In the conclusion the general principle of the solution is discussed and how it is applicable to other test models.

Work on test optimization is found in the statistics, operations research, electronic engineering, software engineering and production management literature; recent papers from these fields include Feyzioglu et al. (2006), Jei-Zheng and Chen-Fu (2008), Li and Lam (2005), Nachlas et al. (1990) and Shi and Sandborn (2006). Typically, deriving the sequence will start with the specification of a model for the tests under consideration. Then the test model parameters are specified and finally an optimization procedure to search for an optimal test sequence is implemented. A wide variety of models and optimization algorithms have been proposed; ant colony (Song et al., 2007) and genetic (Sakthivel and Narayanasamy, 2007) algorithms are currently popular for optimization. Ideally, some sensitivity analysis is also conducted.

The inference of the value of these parameters from test measurement data is not as straightforward as it might appear at first for several reasons. First, the test measurement data are not directly related to the test model parameters. Rather, the test measurements themselves are what are stored. For example, consider a test that measures a voltage and passes the unit if the voltage is within specified limits. A natural model for the voltage measurement is not specified in terms of test parameters such as the probability of a false positive or false negative test result. We address this by defining a model for the measurement data and then relate that model to the test model parameters. This relationship is a deterministic function for the test model of Fisher et al. (2007a) that we use. We fit the measurement data to the measurement model by a Bayesian approach which then gives us a posterior distribution of the test parameters through a change of variable. Second, we want the inference procedure to be applicable to a wide variety of tests that are conducted. Our experience with telecommunications equipment is that the measurement data display a wide variety of behaviour that are challenging to model, such as multimodality and extreme outliers. We address this by using a Gaussian mixture model for the measurement data, which is fitted by the standard Bayesian approach of reversible jump MCMC (Richardson and Green, 1997). Finally, the data are sometimes censored; for example, the database may only record that a measurement was within acceptable limits, rather than the actual value. In this case inference with a Gaussian mixture model is very difficult but it is possible to fit a single Gaussian model. We also discuss specification of prior distributions on measurement model parameters given that the inference procedure should be as automatic as possible, with intervention from the test engineer only for unusual cases.

There is considerably less work on this initial stage of the problem than on the test sequence optimization itself. The closest work that we have found is that of Bounceur et al. (2007), where the problem of estimating test metrics for analogue circuits, before production is started, is considered. Monte Carlo simulation of the circuit and potential faults is used to derive estimates, hence the approach does not make use of production data. The first attempt by the authors to estimate test parameters from measurement data used a classical point estimate approach based on modelling the data as Gaussian, but it did not give an estimation error and produced very inaccurate estimates if the data were not normal (Wilson et al., 2007) or few. This led to the development of the model in this paper. So the novelty of our work is the observation that measurement data can be used to estimate test model parameters, but that this is only successful if the data can be modelled reasonably well, and the implementation of this for a specific test model. It also adds to the growing literature on applications of mixture models in reliability and elsewhere.

The paper is organized as follows. Section 2 describes the data. In Section 3 we describe the imperfect test and repair model. Section 4, describes the Bayesian estimation method and how it is implemented. Section 5 applies the method to simulated and real data, and illustrates the effect of censoring on the inference.

2 DATA

We assume that there are three sets of measurements available from the testing process:

1. The most numerous data are the measured values from each unit when it is tested for the first time. This is known as the "first-pass" data. Units that pass are not observed again. Let $\boldsymbol{y} = y_1, \ldots, y_n$ denote the measured values from the first pass tests on n units;

2. There are some data available from retesting of units that failed the test and were repaired. This is known as the "second-pass" data. For our application, the second pass data consists of how many of the units passed the test after repair, rather than the measurements themselves. The number of units to be retested following repair is denoted l and the number that passed is denoted n_s.

3. There are data from a set of "one-off" tests, where a single unit is tested m times. Such tests are occasionally carried out by the engineers to learn about the repeatability of the test results. We define $\boldsymbol{z} = z_1, \ldots, z_m$ to be the measured values from the one-off test.

While the quantity being measured in the test may be continuous, sometimes the available one-off and first pass data only show whether the test was passed or not. In this case we define z_j and y_i to be 1 or 0, according to whether the test on the unit passed or failed.

3 MODEL

3.1 The Test Model

Figure 1 is a diagram of the test model of Fisher et al. (2007a) that we use throughout this paper. In this model, units are tested and a value y is observed. The test is passed if y lies within specified limits (L, U). In the telecommunications application, the measurements may be continuous (e.g. a voltage) or discrete (e.g. a defect count). Units that fail the test are sent for repair and then retested. For the purposes of optimizing the test sequence, the parameters of interest for this test are:

- The incoming quality of the units, defined to be the proportion of units that "truly" satisfy the specification that is being tested, which is denoted P_{good};
- The probability of a false negative test result (e.g. the test fails a unit that is actually good), which is denoted α;
- The probability of a false positive test result (e.g. the test passes a unit that is actually bad), which is denoted β;
- The probability that a unit that is truly bad is repaired successfully, which is denoted P_{repair}.

We assume that truly good units that fail the test cannot then be repaired to be bad.

3.2 The Data Measurement Model

As defined in Section 2, the data pertain to the value that is measured by the test. We denote the value of the property being measured as x. A Gaussian mixture model is used for x, since it is sufficiently flexible to be able to model the wide range of behaviour that we have seen in practical examples, such as extreme outliers, skewness and multimodality. Therefore:

$$p_x(x) = \sum_{k=1}^{K} p_k \frac{1}{\sqrt{2\pi\sigma_k^2}} \exp\left(-\frac{1}{2\sigma_k^2}(x-\mu_k)^2\right). \tag{1}$$

We denote the observed value of x as y, and assume Gaussian measurement error, so that:

$$p_{y|x}(y \mid x) = \frac{1}{\sqrt{2\pi s^2}} e^{-(y-x)^2/2s^2}. \tag{2}$$

This implies that, marginally, y is a Gaussian mixture with weights p_k , means μ_k and variances $\kappa_k^2 = \sigma_k^2 + s^2$:

$$p_y(y) = \sum_{k=1}^{K} p_k \frac{1}{\sqrt{2\pi\kappa_k^2}} \exp\left(-\frac{1}{2\kappa_k^2}(y-\mu_k)^2\right). \tag{3}$$

In what follows, it will be more convenient to parameterize the data measurement model in terms of the parameters for the mixture model for y. We define $\boldsymbol{\theta} = \{K, p_k, \mu_k, \kappa_k^2 \mid k =$

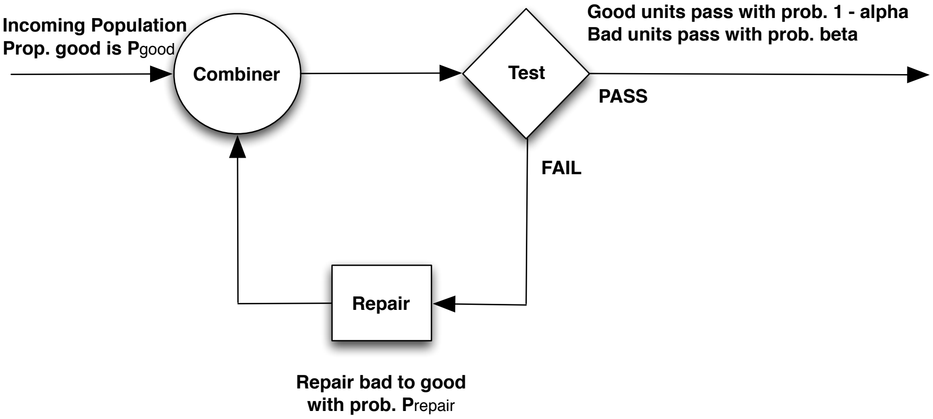

Figure 1: Flow chart of the test and repair model.

$1, \dots, K\}$ to be the set of mixture model parameters for y. Hence the data measurement model is parameterized by $\boldsymbol{\theta}$ and s^2, with the restriction $\min_k \kappa_k^2 > s^2$.

The joint distribution of x and y is a bivariate Gaussian mixture with weights p_k, means $(\mu_k,\ \mu_k)^T$ and covariances

$$\begin{pmatrix} \kappa_k^2 - s^2 & \kappa_k^2 - s^2 \\ \kappa_k^2 - s^2 & \kappa_k^2 \end{pmatrix}, \tag{4}$$

for $k = 1, \dots, K$.

3.3 Relating the Test Model to the Data Measurement Model

Since a unit is classified to be good if x is in the interval (L, U), and passes the test if y is in the interval (L, U), we can say the following:

- The incoming quality is:

$$P_{\text{good}} = P(L \le x \le U \,|\, \boldsymbol{\theta}, s^2) = \sum_k p_k \left[\Phi\left(\frac{U - \mu_k}{\sqrt{\kappa_k^2 - s^2}} \right) - \Phi\left(\frac{L - \mu_k}{\sqrt{\kappa_k^2 - s^2}} \right) \right],$$

  where Φ is the standard normal distribution function;

- The false positive probability is

$$\begin{aligned} \alpha = P(y < L \text{ or } y > U \,|\, L \le x \le U, \boldsymbol{\theta}, s^2) &= 1 - P(L \le y \le U \,|\, L \le x \le U, \boldsymbol{\theta}, s^2) \\ &= 1 - \frac{P(L \le x \le U, L \le y \le U \,|\, \boldsymbol{\theta}, s^2)}{P_{\text{good}}}, \end{aligned}$$

the numerator being a weighted sum of bivariate normal probabilities:

$$\begin{aligned}
&P(L \le x \le U, L \le y \le U \,|\, \boldsymbol{\theta}, s^2) \\
&= \sum_k p_k \Bigg[\Phi_{x,y}(U, U \,|\, \mu_k, \kappa_k^2, s^2) - \Phi_{x,y}(U, L \,|\, \mu_k, \kappa_k^2, s^2) - \Phi_{x,y}(L, U \,|\, \mu_k, \kappa_k^2, s^2) \\
&\qquad\qquad + \Phi_{x,y}(L, L \,|\, \mu_k, \kappa_k^2, s^2) \Bigg], \qquad (5)
\end{aligned}$$

where $\Phi_{x,y}(\cdot \,|\, \mu_k, \kappa_k^2, s^2)$ is the bivariate normal distribution function with mean $(\mu_k,\ \mu_k)^T$ and the covariance matrix of Equation 4;

- The false negative probability is $\beta = P(L \le y \le U \,|\, x < L \text{ or } x > U, \boldsymbol{\theta}, s^2)$, which can be written as

$$\begin{aligned}
\beta &= \frac{P(Y \le U \,|\, \boldsymbol{\theta}) - P(Y \le L \,|\, \boldsymbol{\theta}) - P(L \le x \le U, L \le y \le U \,|\, \boldsymbol{\theta}, s^2)}{1 - P(L \le x \le U \,|\, \boldsymbol{\theta}, s^2)} \\
&= \frac{\sum_k p_k \left[\Phi\left(\frac{U-\mu_k}{\kappa_k}\right) - \Phi\left(\frac{L-\mu_k}{\kappa_k}\right) \right] - P(L \le x \le U, L \le y \le U \,|\, \boldsymbol{\theta}, s^2)}{1 - P_{\text{good}}},
\end{aligned}$$

where $P(L \le x \le U, L \le y \le U \,|\, \boldsymbol{\theta}, s^2)$ is given in Equation 5.

So 3 of the 4 test model parameters — P_{good}, α and β — are easily-computed functions of the data measurement model parameters, being weighted sums of univariate and bivariate Gaussian probabilities. In the next section we see that the remaining test model parameter — P_{repair} — appears in the likelihood for the measurement data, so can be inferred directly.

Figure 2 is a directed acyclic graph representation of the measurement and test models.

4 STATISTICAL INFERENCE

The goal is to compute $p(P_{\text{good}}, \alpha, \beta, P_{\text{repair}} \,|\, \text{data})$. In this section, it is seen that the likelihood is most easily written in terms of $\boldsymbol{\theta}$, s^2, x_{one} and P_{repair}, from which the posterior distribution $p(\boldsymbol{\theta}, s^2, x_{\text{one}}, P_{\text{repair}} \,|\, \text{data})$ may be computed. The (deterministic)

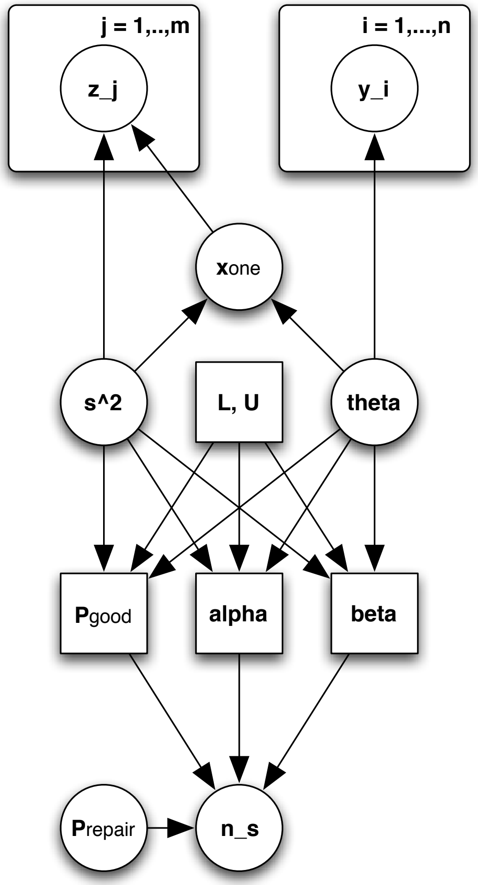

Figure 2: Directed acyclic graph representation of the measurement and test models, and the relationship between them.

definitions in Section 3.3 allow us to compute the desired posterior distribution of P_{good}, α, β and P_{repair} by a change of variable. This is implemented by Monte Carlo simulation from $p(\boldsymbol{\theta}, s^2, x_{\text{one}}, P_{\text{repair}} \,|\, \text{data})$, from which the change of variable is achieved by applying the functions of Section 3.3 to the simulated values $\boldsymbol{\theta}$ and s^2.

4.1 Likelihood for the One-off and First Pass Data

If the measurements themselves are recorded then the likelihood for $\boldsymbol{y}$ is

$$p(\boldsymbol{y} \,|\, \boldsymbol{\theta}) = \prod_{i=1}^{n} p_y(y_i \,|\, \boldsymbol{\theta}) \tag{6}$$

and that for $\boldsymbol{z}$ is:

$$p(\boldsymbol{z} \,|\, s^2, x_{\text{one}}) = \prod_{j=1}^{m} p_{y|x}(z_j \,|\, s^2, x_{\text{one}}), \tag{7}$$

where x_{one} is the unknown true value of the unit used in the one-off test. Equations 2 and 3 give the definitions of p_y and $p_{y|x}$.

If interval censored data are recorded then the likelihood is Bernoulli for both $\boldsymbol{y}$ and $\boldsymbol{z}$ with success probabilities

$$P_y(\boldsymbol{\theta}) = P_y(L \leq y \leq U \,|\, \boldsymbol{\theta}) = \sum_{k=1}^{K} p_k \left[\Phi\left(\frac{U - \mu_k}{\kappa_k}\right) - \Phi\left(\frac{L - \mu_k}{\kappa_k}\right) \right] \tag{8}$$

and

$$P_z(s^2, x_{\text{one}}) = P_{y|x}(L \leq y \leq U \,|\, x = x_{\text{one}}, s^2) = \Phi\left(\frac{U - x_{\text{one}}}{s}\right) - \Phi\left(\frac{L - x_{\text{one}}}{s}\right). \tag{9}$$

respectively. Hence:

$$\begin{aligned} p(\boldsymbol{y} \,|\, \boldsymbol{\theta}) &= P_y(\boldsymbol{\theta})^{n_y}(1 - P_y(\boldsymbol{\theta}))^{n-n_y}; & (10) \\ p(\boldsymbol{z} \,|\, s^2, x_{\text{one}}) &= P_z(s^2, x_{\text{one}})^{n_z}(1 - P_z(s^2, x_{\text{one}}))^{m-n_z} & (11) \end{aligned}$$

where n_y and n_z are the number of units passing in the first pass and one off data respectively.

4.2 Likelihood for the Second Pass Data

We always assume pass/fail data for the second-pass test. The probability of passing is $P_s = P(\text{pass 2nd test} \,|\, \text{fail 1st test})$. Applying the partition law and Bayes' law we can show that:

$$P_s = \frac{\alpha(1-\alpha)P_{\text{good}} + (1-\alpha)P_{\text{repair}}(1-\beta)(1-P_{\text{good}}) + \beta(1-P_{\text{repair}})(1-\beta)(1-P_{\text{good}})}{\alpha P_{\text{good}} + (1-\beta)(1-P_{\text{good}})}. \tag{12}$$

The likelihood for n_s units passing from l is Bernoulli:

$$p(n_s \,|\, P_{\text{good}}, \alpha, \beta, P_{\text{repair}}) \;=\; P_s^{n_s}(1-P_s)^{l-n_s}. \tag{13}$$

4.3 The Prior

The likelihood is written in terms of $\boldsymbol{\theta}$, s^2, x_{one} and P_{repair}, and so for convenience it is preferable to write a prior for these variables as well. For practical implementation of the method, it is unreasonable, on the grounds of the time and effort required, to expect the test engineer to enter much information for each test separately, and so we focus on generic *a priori* knowledge about the model parameters that can be used for all tests.

We assume prior independence between P_{repair} and the others, hence

$$p(\boldsymbol{\theta}, s^2, x_{\text{one}}, P_{\text{repair}}) \;=\; p(P_{\text{repair}})\, p(\boldsymbol{\theta})\, p(s^2 \,|\, \boldsymbol{\theta})\, p(x_{\text{one}} \,|\, \boldsymbol{\theta}, s^2).$$

Prior for $\boldsymbol{P_{\text{repair}}}$: Since P_{repair} is a probability, we propose to use a beta distribution:

$$p(P_{\text{repair}}) = \frac{\Gamma(B_1+B_2)}{\Gamma(B_1)\Gamma(B_2)} P_{\text{repair}}^{B_1-1}(1-P_{\text{repair}})^{B_2-1}, \tag{14}$$

for $0 \le P_{\text{repair}} \le 1$. We suggest the usual non-informative values $B_1 = B_2 = 0.5$ as a default setting.

Prior for $\boldsymbol{\theta}$: We expect most units to pass the test, and so it is reasonable to say that most mixture components in the model for y have means that are in or close to (L, U). We do this by placing independent normal priors on the μ_k, with means $m_\mu = 0.5(U+L)$ (e.g. the centre of the accept interval) and variances $s_\mu^2 = (L+U)^2$, so that about 38%

of the prior masses of the μ_k lie in (L, U). An inverse gamma prior is placed on each κ_k^2 with scale parameter a_κ and shape parameter b_κ, chosen so that for a typical mixture component with a mean in (L, U), the probability over the interval (L, U) is not small, but that also allows for large values of κ_k^2 if necessary. We use $a_\kappa = 3(L + U)^2/32$ and $b_\kappa = 0.5$, which gives a heavy-tailed prior with mode at $\kappa_k^2 = (L+U)^2/16$; at this modal value for κ_k^2 the mixture component has between 50% and 96% of its probability mass in (L, U). A uniform prior is placed on the weights p_k with support $\{p_k \geq 0 \,|\, \sum_k p_k = 1\}$. The normal, inverse gamma and uniform distributions are conjugate priors for the Gaussian mixture so simplify the calculation of the posterior.

Our experience for the telecommunications tests is that the number of components K in the mixture model for x and y will not be large. We place a geometric prior on K with mean 3.

Prior for $\boldsymbol{s^2}$ given $\boldsymbol{\theta}$: An inverse gamma prior is defined for s^2 with scale parameter a_s and shape parameter b_s, that must be truncated to $s^2 < \min_k \kappa_k^2$ (see Section 3.2). Non-informative default settings of $a_s = b_s = 0.5$ may be used if there is no further information on the likely value of s^2.

Prior for $\boldsymbol{x}_{\textbf{one}}$ given $\boldsymbol{\theta}$ and $\boldsymbol{s^2}$: This is the Gaussian mixture of Equation 1.

Prior in the case of censored data: When the data are interval censored, we fit only the single component Gaussian model because the data say almost nothing about the number of components in the mixture. Here, the prior specification becomes more important because the data are less informative. The measurement model parameters are now μ, κ^2 and s^2. We assume a priori that μ is likely to be in the accept interval. If there is some a priori knowledge that μ should be located roughly at m_μ, let μ be normally distributed with a mean m_μ and a variance s_μ^2 such that 95% of the probability mass lies in (L, U) e.g. s_μ^2 is such that $\Phi((U - m_\mu)/s_\mu) - \Phi((L - m_\mu)/s_\mu) = 0.95$. If there is no a priori knowledge about the location of μ then we use $m_\mu = (L + U)/2$ as a default setting. To determine a prior for κ^2 and s^2, we opt for determining hard bounds to their value. First, it should be that the measurement variance is less than the population variance, so that $s^2 < \mathrm{Var}(x) = \kappa^2 - s^2$, which implies $s^2 < 0.5\kappa^2$. A proportion π is specified that is a prior lower bound to the proportion of units passing the test; typically π should be close to 1 so a value closer to 0, say $\pi = 0.05$, would be very conservative. This equates with variances κ^2 and s^2 that are bounded so that $P(L \leq y \leq U) \geq \pi$. Roughly speaking, this means that κ^2 is bounded above by $(0.5(U-L)/z_{0.5(1+\pi)})^2$, where $z_{0.5(1+\pi)}$ is the $50(1+\pi)$ percentile of the standard normal distribution. So we can place a uniform prior distribution for (κ^2, s^2) on the triangle $0 \leq \kappa^2 \leq (0.5(U-L)/z_{0.5(1+\pi)})^2, s^2 \geq 0, s^2 < 0.5\kappa^2$.

4.4 Computation of the Posterior of the Test Model Parameters

The posterior distribution $p(P_{\text{good}}, \alpha, \beta, P_{\text{repair}} \,|\, \boldsymbol{y}, \boldsymbol{z}, n_s)$ is computed by Monte Carlo simulation, which is facilitated by making one approximating assumption. The simulation uses the decomposition

$$p(P_{\text{good}}, \alpha, \beta, P_{\text{repair}} \,|\, \boldsymbol{y}, \boldsymbol{z}, n_s) = p(P_{\text{good}}, \alpha, \beta \,|\, \boldsymbol{y}, \boldsymbol{z}, n_s)\, p(P_{\text{repair}} \,|\, \boldsymbol{y}, \boldsymbol{z}, n_s, P_{\text{good}}, \alpha, \beta).$$

For the former term, it has been noted that the likelihood for n_s as a function of P_{good}, α and β is dominated by the likelihoods for $\boldsymbol{y}$ and $\boldsymbol{z}$, so a reasonable approximation is

$$p(P_{\text{good}}, \alpha, \beta \,|\, \boldsymbol{y}, \boldsymbol{z}, n_s) \;\approx\; p(P_{\text{good}}, \alpha, \beta \,|\, \boldsymbol{y}, \boldsymbol{z}).$$

Making this approximation allows much simpler sampling of $\boldsymbol{\theta}$ and s^2, as shown next. So the simulation is done by generating $(P_{\text{good}}, \alpha, \beta)$ from $p(P_{\text{good}}, \alpha, \beta \,|\, \boldsymbol{y}, \boldsymbol{z})$ and then P_{repair} is simulated from $p(P_{\text{repair}} \,|\, \boldsymbol{y}, \boldsymbol{z}, n_s, P_{\text{good}}, \alpha, \beta)$.

Simulation from $\boldsymbol{p(P_{\text{good}}, \alpha, \beta \,|\, y, z)}$: Given that P_{good}, α and β are functions of $\boldsymbol{\theta}$ and s^2, samples from $p(P_{\text{good}}, \alpha, \beta \,|\, \boldsymbol{y}, \boldsymbol{z})$ can be generated by simulating $\boldsymbol{\theta}$ and s^2 from $p(\boldsymbol{\theta}, s^2, x_{\text{one}} \,|\, \boldsymbol{y}, \boldsymbol{z})$ and computing the corresponding values of P_{good}, α and β from the functions of Section 3.3.

A Gibb's sampler is applied to $p(\boldsymbol{\theta}, s^2, x_{\text{one}} \,|\, \boldsymbol{y}, \boldsymbol{z})$. Samples of s^2 are drawn from

$$p(s^2 \,|\, \boldsymbol{\theta}, x_{\text{one}}, \boldsymbol{y}, \boldsymbol{z}) \propto\; p(\boldsymbol{z} \,|\, x_{\text{one}}, s^2)\, p(x_{\text{one}} \,|\, \boldsymbol{\theta}, s^2)\, p(s^2),$$

for $s^2 < \min_k \kappa_k^2$. Samples of x_{one} are drawn from

$$p(x_{\text{one}} \,|\, \boldsymbol{\theta}, s^2, \boldsymbol{y}, \boldsymbol{z}) \propto \; p(\boldsymbol{z} \,|\, x_{\text{one}}, s^2)\, p(x_{\text{one}} \,|\, \boldsymbol{\theta}, s^2),$$

and samples of $\boldsymbol{\theta} = \{K, p_k, \mu_k, \kappa_k^2 \,|\, k = 1, \ldots, K\}$ are drawn from

$$p(\boldsymbol{\theta} \,|\, s^2, x_{\text{one}}, \boldsymbol{y}, \boldsymbol{z}) \propto \; p(\boldsymbol{y} \,|\, \boldsymbol{\theta})\, p(x_{\text{one}} \,|\, \boldsymbol{\theta}, s^2)\, p(\boldsymbol{\theta}), \;\; \kappa_k^2 > s^2.$$

Since the number of mixture components K is is unknown, $\boldsymbol{\theta}$ is of variable dimension. The details of the full conditional distributions and how they are sampled are in the appendix.

Simulation from $\boldsymbol{p(P_{\text{repair}} \,|\, y, z, n_s, P_{\text{good}}, \alpha, \beta)}$: Because P_{repair} does not appear in the likelihood for either $\boldsymbol{y}$ or $\boldsymbol{z}$, by Bayes Law:

$$p(P_{\text{repair}} \,|\, \boldsymbol{y}, \boldsymbol{z}, n_s, P_{\text{good}}, \alpha, \beta) \;\propto\; p(n_s \,|\, P_{\text{good}}, \alpha, \beta, P_{\text{repair}})\, p(P_{\text{repair}}), \; 0 \leq P_{\text{repair}} \leq 1,$$

which is easily computed, being proportional to the likelihood of Equation 13 multiplied by the prior term in Equation 14. So, given a sample of $(P_{\text{good}}, \alpha, \beta)$, the corresponding sample of P_{repair} may be generated from $p(P_{\text{repair}} \,|\, \text{all data}, P_{\text{good}}, \alpha, \beta)$ by the inverse distribution method.

Computation of posterior distribution in the censored data case: In this case a single Gaussian is assumed for x. The likelihood is the product of Equations 10, 11 and 13 and can be written in terms of only 5 parameters: μ, κ^2, s^2, x_{one} and P_{repair}. A discrete approximation to this posterior can be computed exactly on a grid of values, and then the change of variable to P_{good}, α, β and P_{repair} applied to the grid to yield a discrete approximation to $p(P_{\text{good}}, \alpha, \beta, P_{\text{repair}} \,|\, \boldsymbol{y}, \boldsymbol{z}, n_s)$.

5 EXAMPLES

5.1 Simulation Study

To explore the effectiveness of the algorithm of Section 4, two simulation studies are presented.

In the first, data were simulated from a Gaussian mixture model with 7 components. The means, variances and weights of the components were (1.1,2.1,2.5,3.5,4,6,20), (0.1,0.1,0.1,1.0,0.5,0.2,0.2) and (0.05,0.2,0.2,0.3,0.1,0.05,0.1) respectively. The measurement variance was $s2 = 0.05$ and the true value of the unit used in the one-off tests was $x_{\text{one}} = 3.6$. Units were accepted if they were measured to be in the interval $(1.1, 4.5)$. This led to true test model parameter values of $P_{\text{good}} = 0.754$, $\alpha = 0.0251$ and $\beta = 0.0631$; P_{repair} is specified to be 0.8. Sample sizes were $n = 1000$, $m = 100$ and $n_2 = 6$.

The method of Section 4 was implemented to compute the posterior distribution, based on an MCMC run of 2×10^5 iterations, with the first 5×10^4 iterations discarded as burn-in. It was run on 100 sets of data that were simulated from the Gaussian mixture model. For each data set, a posterior mean and 90% marginal posterior probability interval of P_{good}, α, β and P_{repair} were computed from the sample average, 5th percentile and 95th percentile of the MCMC samples. Table 1 summarizes the results of the study, which indicate that the method performed quite well, with perhaps some slight bias in the posterior mean and the coverage results indicating that generally the uncertainty is somewhat overestimated; the result for P_{repair} is out of line with this but this is for a posterior with high variance.

A second study was done to explore the performance of the Gaussian mixture model on skewed data. Values for x were assumed to come from a skewed Weibull distribution with distribution function $P(X \leq x) = 1 - \exp(-(0.25x)^{1.4})$, $x \geq 0$; this has mean 3.64 and median 3.08. Gaussian measurement error was still assumed. The measurement

		Average (std. dev.)	Coverage of (5%, 95%)
Parameter	True value	of posterior means	probability intervals
P_{good}	0.754	0.723 (0.014)	100%
α	0.0251	0.0302 (0.0029)	98%
β	0.0631	0.0585 (0.0038)	100%
P_{repair}	0.800	0.702 (0.159)	89%

Table 1: Summary of performance of method over 100 simulated Gaussian mixture data sets. Coverage is percentage of probability intervals that contained the true value.

variance and true value of the one-off test unit were kept at 0.05 and 3.6; the accept interval was $(0, 8)$. A simulation of 10^7 values of x and y was used to estimate the true values of the test parameters as: $P_{\text{good}} = 0.572$, $\alpha = 0.0161$ and $\beta = 0.0215$; P_{repair} was specified to be 0.8. Sample sizes for the data were $n = 1000$, $m = 100$ and $n_2 = 6$.

The method of Section 4 was implemented to compute the posterior distribution, based on an MCMC run of 2×10^5 iterations, with the first 5×10^4 iterations discarded as burn-in. This was run on 100 sets of data. Table 2 summarizes the results of the study, and we see that the method returns accurate estimates of the test model parameters. The performance of the posterior means is similar to the first study; broadly accurate with modest evidence of some bias towards 0.5. The exception is α, which has not been estimated well with significant bias. The true value of 0.0161 is the closest

5.2 A Real Example

Figure 4 shows data from a real test that have been linearly transformed to preserve confidentiality. The accept limits for this test are $(43, 53)$. The histogram of the first pass data shows three distinct modes in the data, of which one lies well inside the accept interval and two lie well outside (at more than 10 standard deviations of the data lying inside the accept interval). The middle mode in the histogram consists of 61 values that have exactly the same value.

Sample sizes are: $n = 878$ (of which 807 passed), $m = 22$ (of which all passed and the sample standard deviation is 0.035) and $n_2 = 3$ (of which 2 passed). The method of Section 4 was implemented, with 10^6 iterations of the MCMC sampler computed, with the first 10^5 iterations ignored as burn-in. Convergence and mixing of the sampler are

Parameter	True value	Average (std. dev.) of posterior means	Coverage of (5%, 95%) probability intervals
P_{good}	0.572	0.516 (0.032)	100%
α	0.0161	0.0244 (0.0018)	68%
β	0.0215	0.0258 (0.0020)	100%
P_{repair}	0.800	0.752 (0.148)	95%

Table 2: Summary of performance of method over 100 simulated Weibull data sets.

illustrated in Figure 3. There are some concerns about mixing of some components of $\boldsymbol{\theta}$, but they do not seem to be reflected in the sampled values of the test model parameters, which appear to exhibit good convergence and mixing.

Figure 5 are estimated marginal posterior distributions, based on a kernel density approximation from the MCMC samples. The resulting posterior means and $(2.5\%, 97.5\%)$ probability intervals were: $P_{\text{good}} = 0.92,\ (0.89, 0.94)$; $\alpha = 0.0030,\ (0.0011, 0.0066)$; $\beta = 0.020,\ (0.006, 0.042)$; $P_{\text{repair}} = 0.60,\ (0.11, 0.96)$. The estimate of P_{repair} has high variance due to the small sample of second pass data.

As regards model fit, Figure 4 also shows the Gaussian model fit to the first pass data, using the usual MCMC approximation:

$$p(y \,|\, \text{data}) \;\approx\; \frac{1}{D} \sum_{k} p_y(y \,|\, \boldsymbol{\theta}^{(d)}),$$

where $\boldsymbol{\theta}^{(d)}$ is the dth of D samples after burn-in from the MCMC method. While the principal mode in the data is captured well, the middle mode has been modelled by a component with large variance, that also accounts for the right mode. It is noted that while this may be unsatisfactory from the point of view of modelling the first pass data, it is not necessarily of concern from the point of view of estimating the test model parameters because they only depend on interval probabilities of the mixture model, which may be estimated well. As another model assessment check, the measurement error is quite small in this example, so P_{good} should be close to the proportion of first pass data that pass the test, which is $807/878 = 0.92$. That matches the posterior mean of P_{good}, suggesting that we have estimated it well in spite of the poor estimation of some of the modes of the first pass data by the mixture. Computation time was about 3×10^5 iterations per hour, using code written in MATLAB and run on an i-Mac with 2.8GHz dual core processor.

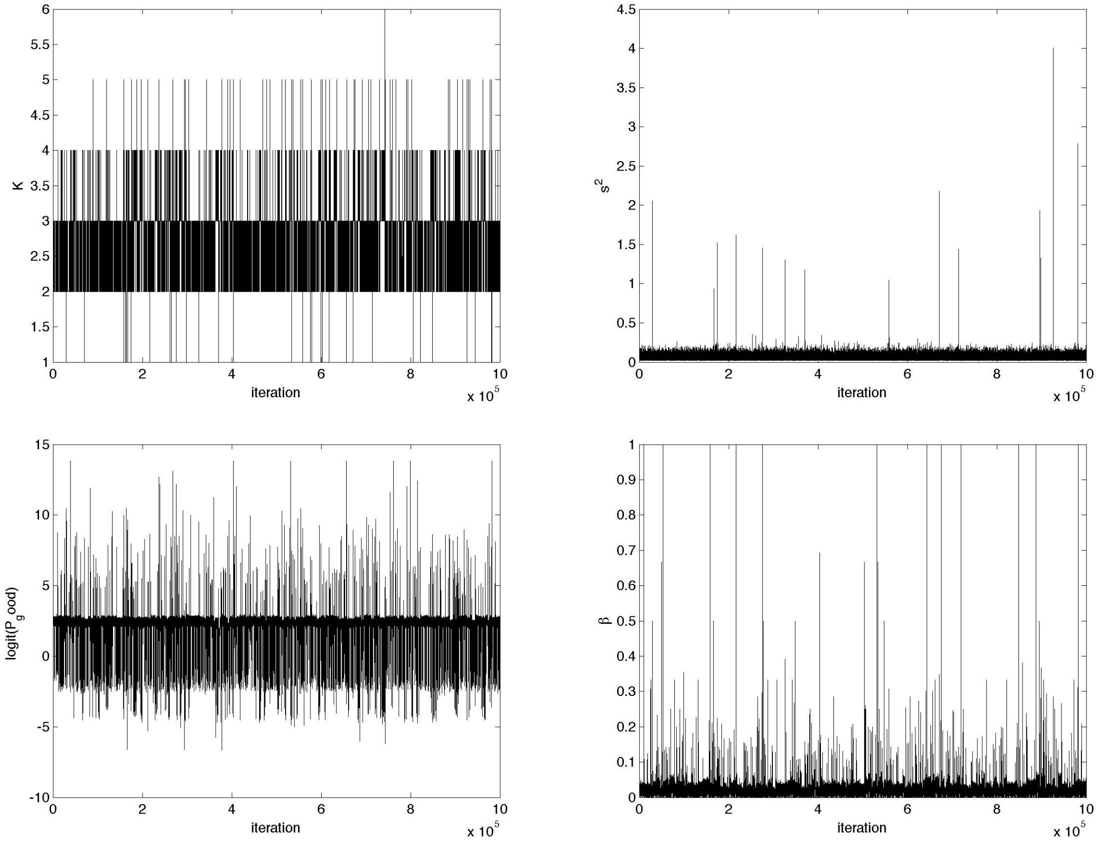

Figure 3: Trace plots of MCMC output for the real test data example (clockwise from top left) K, s^2, $\mathrm{logit}(P_{\mathrm{good}})$ and β.

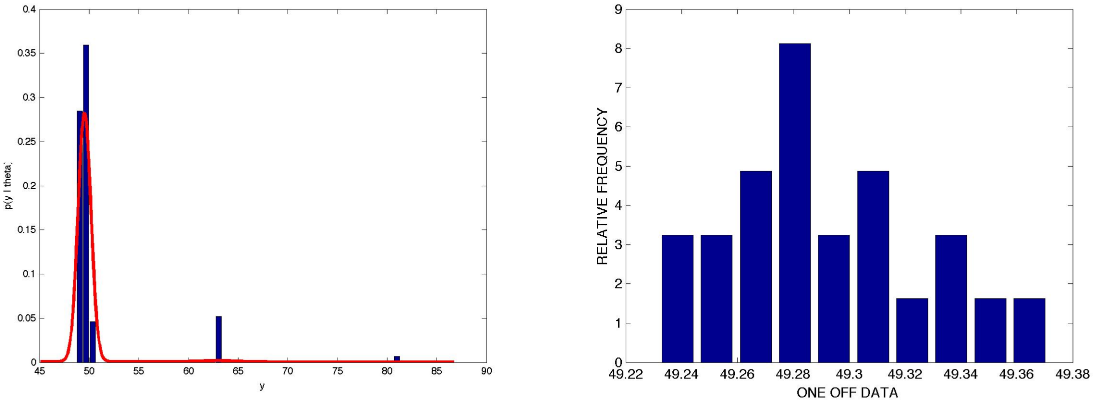

Figure 4: Histograms of first pass data, with model fit (left) and one off data (right) for a telecommunications equipment test.

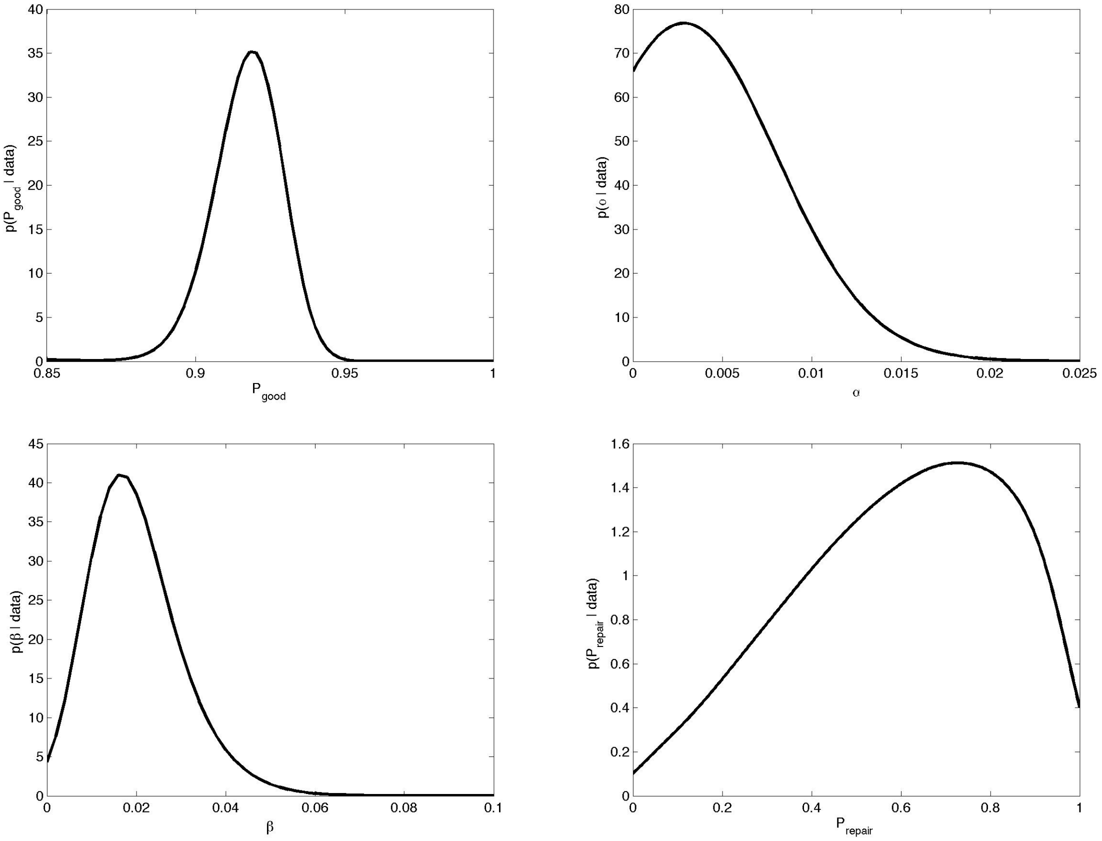

Figure 5: Posterior distributions of (clockwise from top left) P_{good}, α, P_{repair} and β for the real test data example.

5.3 Real Example with Censored Data

To illustrate the effect of censoring, the real example data was taken and it was assumed that the first pass and one-off data were interval censored, so that our data were simply pass/fail. A single component Gaussian model was fitted. The prior specification of Section 4.3 was followed. The analysis gave posterior means and $(2.5\%, 97.5\%)$ probability intervals as: $P_{\text{good}} = 0.95\ (0.87, 0.997)$; $\alpha = 0.061\ (0.007, 0.120)$; $\beta = 0.303\ (0.038, 0.620)$; $P_{\text{repair}} = 0.55\ (0.13, 0.99)$. Contrasting these with their analysis in Section 5.2, it can be seen that the posterior distributions have considerably higher variance, reflecting the loss of information from the censoring, particularly for estimation of α and β.

6 CONCLUSIONS

A Bayesian approach to estimating test properties of an imperfect test and repair model from test measurement data has been presented. The method is able to handle multimodal distributions for the data through a Gaussian mixture. The first observation is that the usual problems of good estimation of mixture models are encountered, particularly with the use of the reversible jump sampler (Brooks and Guidici, 2000), such as difficulty in identifying all modes, and mixing, but they are not necessarily detrimental to estimation of the test parameters, which depend on good estimation of interval probabilities of the mixture model. It is seen from the example data and others that have been fitted, that these interval probabilities are often estimated quite well even when the mixture model does not perform so well. When data are censored, a single Gaussian can be fitted and good estimates of test parameters can be obtained, although with higher posterior variance.

The examples show that estimation seems reasonably accurate but could be improved. However we point out that the procedure is completely automatic, with no tuning parameters to be specified. This is an important property of the method for its use in the field. We argue that the performance of the method, given this lack of intervention, makes it useful. A disadvantage of the method is the computation time — at least 1 hour per data set is recommended for our MATLAB code with 2008 processing power — meaning that this is not a method that can be employed in an online test sequence design. On the other hand, these estimations can often be carried out offline and the estimates stored for use in an online setting.

The general scheme outlined — specifying a test measurement model for x and y, deriving the test parameters as functions of the parameters of that model, fitting data to the measurement model and applying that fit to estimate the test parameters — is equally applicable to other distributions for x and y given x, either continuous or discrete. It can also be applied to other test models, as long as the relationship with the test measurement model can be established. The mixture of Gaussian model was chosen because of its flexibility, in line with the desire to keep the method automatic. This leads to the question of how much better the performance would be with an informed choice of model, and if this is worth the burden of model selection.

Several other issues have come out of the work so far. First, it is reasonable for some tests to assume that measurement error variance changes with x e.g. multiplicative error. The extension of the model to this case, by transformation of the data or parametric modelling of the variance structure, should be straightforward and present few new computational difficulties. Also, in some tests we have observed that the measurement error variance s^2 can in fact be larger than the population variance, contrary to our prior opinion. In this case the data are not as informative but the model is still fitted satisfactorily, although the posterior distributions will have high variance. Second, more work can be done to specify informative priors that could help analyses, especially in the censored data case. Another issue is that we have left P_{repair} to be defined directly rather than through the measurement model. This means that we have no way of using the test measurements to infer anything more than the probability that a repair succeeds or not. A model for the effect of repair on the measured value would allow a more detailed understanding of the performance of the repair process.

Currently this approach is used to compute point and interval estimates of the many tests that are used in typical production testing in telecommunications. These are done separately for each test and used as inputs into a separate test sequence optimization tool. The full benefit of the Bayesian approach would be realized by treating optimal test sequencing as decision making under uncertainty, the uncertainty quantified by the posterior distributions of test parameters. There are clearly dependencies between the tests, since they are applied to the same units, and may occur in the same production facility, which are ignored. A hierarchical model for the set of tests, that allowed borrowing of strength from tests with a lot of data, is one approach to modelling this dependence that would be easy to implement.

ACKNOWLEDGEMENTS

This research is work of the Centre for Telecommunications Value-Chain Research `http://www.ctvr.ie`, supported by Science Foundation Ireland under grant number 03/CE3/I405. Bell Labs Ireland research was also partly funded by a grant from the Irish Development Agency.

References

Bounceur, A., S. Mir, E. Simeu, and L. Rolindez (2007). Estimation of test metrics for the optimisation of analogue circuit testing. *Journal of Electronic Testing – Theory and Applications 23*, 471–484.

Brooks, S. P. and P. Guidici (2000). Markov chain Monte Carlo convergence assessment via two-way analysis of variance. *J. Comput. Graphi. Stat. 9*(2), 266–285.

Dick, J. H., E. Trischler, C. Dislis, and A. P. Ambler (1994). Sensitivity analysis in economic based test strategy planning. *Journal of Electronic Testing: Theory and Applications 5*, 239–252.

Feyzioglu, O., I. K. Altmel, and S. Ozekici (2006). The design of optimum component test plans for system reliability. *Computational Statistics and Data Analysis 50*, 3099–3112.

Fisher, E., S. Fortune, M. Gladstein, S. Goyal, W. Lyons, J. Mosher, and G. Wilfong

(2007a). Economic modeling of global test strategy I: mathematical models. *Bell Labs Technical Journal 12*, 161–174.

Fisher, E., S. Fortune, M. Gladstein, S. Goyal, W. Lyons, J. Mosher, and G. Wilfong (2007b). Economic modeling of global test strategy II: software system and examples. *Bell Labs Technical Journal 12*, 175–186.

Grout, I. A. (2005). *Integrated Circuit Test Engineering: Modern Techniques.* London: Springer Verlag.

Jei-Zheng, W. and C. Chen-Fu (2008). Modeling semiconductor testing job scheduling and dynamic machine configuration. *Expert Systems with Applications 35*, 485–496.

Li, H. Z. and C. P. Lam (2005). An ant colony optimization approach to test sequence generation for statebased software testing. In *Proceedings of the 5th International Conference on Quality Software*, Los Alamitos, Cal., pp. 255–262. IEEE Computer Society.

Nachlas, J. A., S. R. Loney, and B. A. Binney (1990). Diagnostic-strategy selection for series systems. *IEEE Trans. Rel. 30*, 271–280.

Richardson, S. and P. Green (1997). On Bayesian analysis of mixtures with an unknown number of components (with discussion). *Journal of the Royal Statistical Society, Series B 59*, 731–792.

Sakthivel, P. and P. Narayanasamy (2007). An approach to the design of optimal test scheduling for system-on-chip based on genetic algorithm. In T. Sobh (Ed.), *Innovations and Advanced techniques in Computer and Information Sciences*, pp. 25–28. Proceedings of the International Conference on Systems, Computing Science and Software Engineering, 2006.

Shi, Z. and P. Sandborn (2006). Optimization of test/diagnosis/rework location(s) and characteristics in electronic system assembly. *Journal of Electronic Testing – Theory and Applications 1*, 49–60.

Song, Y., M. T. Zhang, J. Yi, L. Zhang, and L. Zheng (2007). Bottleneck station scheduling in semiconductor assembly and test manufacturing using ant colony optimization. *IEEE Transactions on Automation Science and Engineering 4*, 569–578.

Sun, Y. (2008). *Test and Diagnosis of Analogue, Mixed-signal and Rf Integrated Circuits: the System on a Chip Approach.* London: Institution of Engineering and Technology.

Wilson, S. P., B. Flood, S. Goyal, J. Mosher, J. O'Brien, and R. Kennedy (2007). Parameter estimation for a model with both imperfect test and repair. In *Proceedings of the 25th IEEE VLSI Test Symposium*, pp. 271–276. IEEE Computer Society.

Derivation of the Distributions for Sampling from $p(\boldsymbol{\theta}, s^2, x_{\text{one}} \,|\, \boldsymbol{y}, \boldsymbol{z})$

Monte Carlo simulation from $p(\boldsymbol{\theta}, s^2, x_{\text{one}} \,|\, \boldsymbol{y}, \boldsymbol{z})$ is done by a Gibbs sampler. Samples of s^2, x_{one} and the components of $\boldsymbol{\theta} = \{K, p_k, \mu_k, \kappa_k^2 \,|\, k = 1, \ldots, K\}$ are sampled from their full conditional distributions. Because x_{one} and $\boldsymbol{y}$ are Gaussian mixtures, it is convenient to define mixture component indicator variables $a_x \in \{1, \ldots, K\}$ for x_{one} and $a_i \in \{1, \ldots, K\}$ for y_i, $i = 1, \ldots, n$, with probability distributions given by the mixture model weights: $p(a_x \,|\, \boldsymbol{\theta}) = p_{a_x}$ and $p(a_i \,|\, \boldsymbol{\theta}) = p_{a_i}$. Let $\boldsymbol{A} = \{a_i \,|\, i = 1, \ldots, n\}$. Both a_x and $\boldsymbol{A}$ are also sampled, hence the Gibb's sampler is applied to $p(\boldsymbol{\theta}, s^2, x_{\text{one}}, \boldsymbol{A}, a_x \,|\, \boldsymbol{y}, \boldsymbol{z})$.

Sampling $\boldsymbol{a_x}$: This is sampled by the inverse transform method from the discrete distribution

$$\begin{aligned} p(a_x \,|\, \boldsymbol{\theta}, s^2, x_{\text{one}}, \boldsymbol{A}, \boldsymbol{y}, \boldsymbol{z}) &\propto p(x_{\text{one}} \,|\, \boldsymbol{\theta}, s^2, a_x)\, p(a_x \,|\, \boldsymbol{\theta}) \\ &\propto \frac{p_{a_x}}{\sqrt{\kappa_{a_x}^2 - s^2}} \exp\left(-\frac{1}{2(\kappa_{a_x}^2 - s^2)} \left((x_{\text{one}} - \mu_{a_x})^2 \right) \right), \; a_x = 1, \ldots, K. \end{aligned}$$

Sampling $\boldsymbol{x}_{\mathbf{one}}$: This is sampled from

$$\begin{aligned} p(x_{\text{one}} \,|\, \boldsymbol{\theta}, s^2, \boldsymbol{A}, a_x, \boldsymbol{y}, \boldsymbol{z}) &\propto p(\boldsymbol{z} \,|\, x_{\text{one}}, s^2)\, p(x_{\text{one}} \,|\, \boldsymbol{\theta}, s^2, a_x) \\ &\propto \exp\left(-\frac{1}{2} \left(\frac{(x_{\text{one}} - \mu_{a_x})^2}{(\kappa_{a_x}^2 - s^2)} + \frac{1}{s^2} \sum_j (z_j - x_{\text{one}})^2 \right) \right), \end{aligned}$$

which is a Gaussian distribution with variance $\sigma_{\text{one}}^2 = s^2(\kappa_{a_x}^2 - s^2)/(s^2 + m(\kappa_{a_x}^2 - s^2))$ and mean

$$\sigma_{\text{one}}^2 \left(\frac{m}{s^2} \bar{z} + \frac{\mu_{a_x}}{\kappa_{a_x}^2 - s^2} \right).$$

Sampling $\boldsymbol{s^2}$: This is sampled from

$$\begin{aligned} p(s^2 \,|\, \boldsymbol{\theta}, x_{\text{one}}, \boldsymbol{A}, a_x, \boldsymbol{y}, \boldsymbol{z}) &\propto p(\boldsymbol{z} \,|\, s^2, x_{\text{one}})\, p(x_{\text{one}} \,|\, \boldsymbol{\theta}, s^2, a_x)\, p(s^2) \\ &\propto \left[(s^2)^{-(1+b_s+0.5m)} e^{-(a_s + 0.5 \sum_j (z_j - x_{\text{one}})^2)/s^2} \right] \frac{e^{-0.5(x_{\text{one}} - \mu_{a_x})^2/(\kappa_{a_x}^2 - s^2)}}{\sqrt{\kappa_{a_x}^2 - s^2}}, \end{aligned}$$

with the constraint that $s^2 < \kappa_{a_x}^2$. The term in square brackets is an inverse gamma distribution and dominates the function when m is not small. This suggests an independent Metropolis sampler with proposal distribution as this inverse gamma distribution, with scale parameter $a_s + 0.5 \sum_j (z_j - x_{\text{one}})^2$ and shape parameter $b_s + 0.5m$, truncated to $(0, \kappa_{a_x}^2)$. Given a current value s^2, a new value s_*^2 is proposed from this truncated inverse gamma and accepted with probability

$$\min\left\{ 1, \frac{(\kappa_{a_x}^2 - s_*^2)^{-0.5} e^{-0.5(x_{\text{one}} - \mu_{a_x})^2/(\kappa_{a_x}^2 - s_*^2)}}{(\kappa_{a_x}^2 - s^2)^{-0.5} e^{-0.5(x_{\text{one}} - \mu_{a_x})^2/(\kappa_{a_x}^2 - s^2)}} \right\}.$$

Sampling $\boldsymbol{A}$: The ith component a_i of $\boldsymbol{A}$ is sampled by the inverse transform method from the discrete distribution

$$\begin{aligned} p(a_i \,|\, \boldsymbol{\theta}, x_{\text{one}}, \boldsymbol{A}_{-i}, a_x, \boldsymbol{y}, \boldsymbol{z}) &\propto p(y_i \,|\, \boldsymbol{\theta}, a_i)\, p(a_i \,|\, \boldsymbol{\theta}) \\ &\propto \frac{p_{a_i}}{\kappa_{a_i}} \exp\left(-\frac{1}{2\kappa_{a_i}^2} (y_i - \mu_{a_i})^2 \right), \; a_i = 1, \ldots, K, \end{aligned}$$

for $i = 1, \ldots, n$.

Sampling $\boldsymbol{\theta} = \{\boldsymbol{K}, \boldsymbol{p_k}, \boldsymbol{\mu_k}, \boldsymbol{\kappa_k^2} \,|\, \boldsymbol{k} = \mathbf{1}, \ldots, \boldsymbol{K}\}$: Because K is unknown, $\boldsymbol{\theta}$ is of unknown dimension. We adapt the reversible jump method of Richardson and Green (1997) to sample from its full conditional distribution

$$p(\boldsymbol{\theta} \,|\, s^2, x_{\text{one}}, \boldsymbol{A}, a_x, \boldsymbol{y}, \boldsymbol{z}) \;\propto\; p(\boldsymbol{y} \,|\, \boldsymbol{\theta}, \boldsymbol{A})\, p(x_{\text{one}} \,|\, \boldsymbol{\theta}, s^2, a_x)\, p(\boldsymbol{\theta}).$$

Were the full conditional distribution of $\boldsymbol{\theta}$ to be proportional to simply $p(\boldsymbol{y} \,|\, \boldsymbol{\theta}, \boldsymbol{A})\, p(\boldsymbol{\theta})$ then it would follow the model and method of Richardson and Green (1997) for updating the components of $\boldsymbol{\theta}$, including proposing changes in the number of mixture components K. The full conditional distributions of the μ_k, κ_k^2 and p_k would be Gaussian, inverse gamma and Dirichlet respectively. The only change to the method of Richardson and Green (1997) would come from the fact that the inverse gamma distributions for κ_k^2 are truncated to (s^2, ∞), which is easily handled.

However we have the extra term $p(x_{\text{one}} \,|\, \boldsymbol{\theta}, s^2, a_x)$ in the full conditional distribution that complicates matters somewhat in the following manner:

- For the p_k, the effect is minimal. The full conditional distribution of $(p_1, \ldots, p_K)$ is Dirichlet with parameters $1 + I(a_x = 1) + N_1, \ldots, 1 + I(a_x = K) + N_K$, where $N_k = |\{i \,|\, a_i = k\}|$.

- For the μ_k and κ_k^2, the effect is a little more complicated. A simple idea is to use $p(\boldsymbol{y} \,|\, \boldsymbol{\theta}, \boldsymbol{A})\, p(\boldsymbol{\theta})$ as proposal distributions in a Metropolis step. For μ_k, this leads to proposing μ_k^* from a Gaussian with mean

$$\frac{\kappa_k^{-2} \sum_{i:\, a_i = k} y_i + s_\mu^{-2} m_\mu}{\kappa_k^{-2} N_k + s_\mu^{-2}}$$

and variance $(\kappa_k^{-2} N_k + s_\mu^{-2})^{-1}$, then accepting with probability

$$\begin{aligned}
&\min\left\{1, \frac{p(x_{\text{one}} \,|\, \boldsymbol{\theta} \text{ including } \mu_k^*, s^2, a_x)}{p(x_{\text{one}} \,|\, \boldsymbol{\theta} \text{ including } \mu_k, s^2, a_x)}\right\} \\
&\quad = \begin{cases} 1, & \text{if } k \neq a_x, \\ \min\left\{1, \exp\left(\frac{1}{2(\kappa_k^2 - s^2)}\left((x_{\text{one}} - \mu_k)^2 - (x_{\text{one}} - \mu_k^*)^2\right)\right)\right\}, & \text{if } k = a_x. \end{cases}
\end{aligned}$$

- For κ_k^2, a value $\kappa_k^{2,*}$ is proposed from an inverse gamma distribution with scale parameter $a_\kappa + 0.5\sum_{i:\,a_i=k}(y_i - \mu_k)^2$ and shape parameter $b_\kappa + 0.5N_k$, truncated to (s^2, ∞) in order to respect the constraint that $s^2 < \min_k \kappa_k^2$. This is accepted with probability:

$$\min\left\{1, \frac{p(x_{\text{one}}\,|\,\boldsymbol{\theta}\text{ including }\kappa_k^{2,*}, s^2, a_x)}{p(x_{\text{one}}\,|\,\boldsymbol{\theta}\text{ including }\kappa_k^2, s^2, a_x)}\right\} = \begin{cases} 1, & \text{if } k \neq a_x,\\ \min\left\{1, \frac{\sqrt{\kappa_k^2 - s^2}\,\exp\left(-\frac{(x_{\text{one}}-\mu_k)^2}{2(\kappa_k^{2,*}-s^2)}\right)}{\sqrt{\kappa_k^{2,*}-s^2}\,\exp\left(-\frac{(x_{\text{one}}-\mu_k)^2}{2(\kappa_k^2-s^2)}\right)}\right\}, & \text{if } k = a_x.\end{cases}$$

- Finally, for K, we can use the split/merge/birth/death proposals from Richardson and Green (1997), the only change being that the accept probabilities for the proposals are multiplied by the expression $p(x_{\text{one}}\,|\,\boldsymbol{\theta}$ for the proposed $K, s^2, a_x)/p(x_{\text{one}}\,|\,\boldsymbol{\theta}$ for the current The explanation of these moves and the derivation of the accept probability is lengthy and we refer to Richardson and Green (1997) for the details.