Towards Statistical Control of an Industrial Test Process

Gaetano Lombardi[1], Emilia Peciola [1], Raffaela Mirandola[2], Antonia Bertolino[3], Eda Marchetti[3]

[1] Ericsson Telecomunicazioni SpA, Roma, Italy
{teiloga, E.Peciola}@rd.tei.ericsson.se
[2] Dip. di Informatica, Sistemi e Produzione Univ. "Tor Vergata", Roma, Italy
mirandola@info.uniroma2.it
[3] Istituto di Elaborazione della Informazione, CNR, Pisa, Italy
{bertolino, e.marchetti}@iei.pi.cnr.it

Abstract. We present an ongoing experience aimed at introducing statistical process control techniques to one crucial test phase, namely Function Test, of a real world software development process. We have developed a prediction procedure, using which, among other things, we compare the performance of a Classical model vs. a Bayesian approach. We provide here the description of the prediction procedure, and a few examples of use of the models over real sets of data. However, far from aimed at identifying new statistical models, the focus of this work is rather about *putting measurement in practice*, and in easy and effective steps to improve the status of control over test processes in a soft, bottom-up approach. The experience described has started quite recently, and the results obtained so far, although limited in scope, are quite encouraging (as well as exciting for the involved team), both in terms of predictive validity of the models and of the positive response got from development personnel.

1 Introduction

It is only in the state of statistical control that statistical theory provides with a high degree of belief, prediction of performance in the immediate future

W. Edwards Deming

In this paper we report about an ongoing experience at Ericsson Telecomunicazioni S.p.A. in Rome (TEI in the following) aimed at applying statistical process control techniques to the Function Test process.

We discuss here the objective of this study and the starting point. In the next section we outline the approach taken. In Sections 3 and 4 we briefly describe two statistical estimators used for data analysis, from a Classical and a Bayesian viewpoint, respectively. Section 5 then gives a few examples, and Section 6 provides the conclusions.

M. Felici, K. Kanoun, A. Pasquini (Eds.): SAFECOMP'99, LNCS 1698, pp. 260-271, 1999

1.1 Objective

Within the frame of the company-wide Ericsson System Software Initiative (ESSI), regular assessments are being performed at all Ericsson Software Design Centers according to the Capability Maturity Model (CMM) for software, Version 2 draft C [1]. Main objectives of ESSI are to identify key areas for process improvement and to propose a framework for subsequent improvement actions. An assessment has been recently performed at TEI organization covering the AXE10 (multi-application, open-ended digital switching product for public telecommunications networks) software development area. The software processes at TEI were found to be at the Defined level of maturity (level 3). Although this result was very satisfying, TEI is now going to initiate some of the level 4 practices. The organization intends to improve its capabilities in statistical process control and in prediction methods. As it is not economically justifiable to apply statistical process control techniques to all processes, a set of processes has been selected according to the business objectives of the organization. One of the selected processes is Function Test, that is one of the four test phases in TEI test strategy, namely:

1) Basic Test, testing the smallest module (test object) in the system. The goal is to verify design specification;
2) Integration Test, testing a functional area: all modules in that area are integrated;
3) Function Test, verifying system functions;
4) System Test, verifying system performance and architectural requirements.

Function Test has been identified as strategic in meeting the commitments to customers with respect to quality objectives, for the following reason. One of the TEI objectives is reducing of a determined amount the fault density figures that are obtained by monitoring the first six months of operation of released products. Fault density is measured by the ratio between the cumulative number of failures observed in those six months and the product size, expressed in lines of code. Root Cause Analysis (RCA) of reported failures is routinely performed, to track back failures to the phase in which they have been originated. An important finding of RCA for TEI products was that a high percentage of failures (48%) corresponded to software faults that could have been discovered during the Function Test phase. Therefore, one of the actions proposed to reduce fault density figures is related to reducing the failures slipping through from Function Test to operation. The failures slipping through, that is one of TEI GPC quality objectives, is measured as the ratio between the number of failures found during first six months, and the sum of failures found during Function Test and first six months.

In this paper we apply statistical techniques taken from the literature to the Function Test process in order to put under control its effectiveness in failure detection.

1.2 Starting Point

Currently, Function Test is performed along a function test specification, with the goal of testing conformance of the target function to its specifications. Test cases are

derived manually by testers, by making a systematic analysis of the specification documentation and trying to cover all the specified functionalities (or use cases). It means that the test cases are deterministically chosen by examining the functional specifications and altogether before test execution starts (which implies that the number of tests to be executed is decided in advance).

Function Test execution is organised in a specified number of stages. The tests are not executed continuously, but only during the working days (i.e., five days in a week) and 8 hours per day. All the failures discovered within a stage are logged and reported to software designers, who trace failures back to code and correct them. A new software version is then released, which is resubmitted to test in the next test stage. For each project, the information registered consists of the start and end dates of the test phase, and of the calendar day (but not the day time) of discovery of each failure. Test execution (CPU) times were not recorded.

Function Test stops when all the test cases defined in the test case specification have been successfully performed, either at first try or after suitable fault repair. Specific exit criteria related to the measured rate of failures detected over the testing period are not explicitly considered in test process, and no estimation of achieved remaining number of faults is currently performed.

2 The Approach

Measurement provides an assessment of an entity under observation (more precisely, of some attributes of this entity [2]). However, the specific objectives for doing any measurement must be clearly stated within a well-defined measurement programme. In fact, only when the objectives are explicitly identified, by interpreting the results of measurement we can take appropriate decisions, and put these into useful actions [2].

2.1 Measurement Objectives

In this study the final objectives are: to put under statistical control the Function Test phase and to investigate the feasibility of introducing testing exit criteria according to remaining faults prediction.

One of the attributes to measure to achieve these goals is the *effectiveness in failure detection* during Function Test, i.e., the rate of failures detected over a fixed period of testing. In particular we use the failure data observed in the first part of the test process to *predict* the expected cumulative number of failures over the planned period of Function Test. A very important property of prediction system is the speed of convergence of estimates. On this respect, we are currently comparing the performances of different estimators (see section 3 and 4).

However, using the predictions provided by the estimators requires insight and knowledge of the process which goes well behind the statistical analyses described here. For instance, suppose that measurement brings to our attention an unexpectedly low number of failures with respect to standard figures. This can be due to an ineffective test process (bad news), or instead to a very good development process (good news).

How the presented estimators are used in project control and management and how historical data through several projects are used to set reference/target measures is outside the scope of the present paper.

2.2 Data Model

As a first step in this investigation, we analysed the typology of data available to lay down an appropriate data model. We could access sets of failure data collected over several projects during the phase of Function Test.

We decided to adopt a data model close to the current logging procedures, as it would be difficult and expensive to change them. As the failure reports are registered on a daily base, we decided to group the failure data into *test intervals* (TIs), each one a day long. A TI in which at least a failure is observed is called a *failed test interval* (FTI), otherwise it is said a successful TI. Quite obviously, anyhow small a test interval is chosen, until this remains larger than a single test there will always be a chance to observe more than one failure within it.

Hence, our model estimates the expected number of failures in two subsequent steps: first we predict N_{FTI}, i.e., the expected number of FTIs; then, from this number, we derive the expected number of failures N_F. To do this, we define a random variable Q to denote the probability that the next TI is failed. Then, over a NTI long period of test intervals, using a valid estimate $\hat{Q}$ of Q, we easily obtain:

$$N_{FTI} = NTI \bullet \hat{Q} \quad (1)$$

Once a value for N_{FTI} is so estimated, the total number of failures clearly will depend on how many failures on average are observed within a FTI. We again introduce a random variable F to represent the number of failures observed within a FTI, and then derive N_F from N_{FTI}, with:

$$N_F = N_{FTI} \bullet \hat{F} \quad (2)$$

As concerns the estimation of $\hat{F}$, we decided to adopt the classical estimator E[F], based on the *sample mean* (also called *arithmetic mean*) of the observed failures over the number of observed FTIs. The choice of the sample mean lies on the observation that, for the projects considered, it soon stabilizes. Furthermore, for large samples it shows the property of consistency and unbiasedness [3].

2.3 Prediction Procedure

The statistical control procedure is based on the following main steps:

- Consider the test intervals assembled in groups of 5 TIs (corresponding to one calendar week of testing) and assign to each group an increasing identification number k with $k = 1,..., \frac{NTI}{5}$.
- After observing the k-th (current) group of failure data derive values of:

  2.1) the cumulative (i.e., from group 1 to group k inclusive) number of FTIs

2.2) the cumulative number of failures

- Using the observations of step 2) derive:

  3.1) an estimate $\hat{Q}$ of Q, based on statistical models (Sect. 3 and 4)

  3.2) an estimate $\hat{F} = \mathrm{E}[F]$ of F

  3.3) predictions of global N_{FTI} and global N_F over a future period testing, based on formulas (1) and (2)

- By use of classical statistical techniques (e.g., confidence interval, relative error) evaluate the accuracy of the estimates obtained at steps 3.1), and 3.2)
- If the estimates $\hat{Q}$ and $\hat{F}$ do not reach the desired level of accuracy, wait for the data relative to another group of 5 TIs, increment k and repeat Steps 2 through 4.
- Check the model, i.e., evaluate if the proposed model and the substantive conclusions fit the data and how sensitive are the results to the modelling assumptions.

In the next sections we describe two different estimators used in step 3.1). Specifically, Section 3 shows a model based on the Classical (frequentist) approach, while Section 4 presents a model based on the Bayesian approach.

3 Using a Classical Approach

A classical approach to derive the probability Q that the next TI is failed, given a sample of NTI, is based on the *maximum likelihood* estimate [3, 4]. The idea underlying the maximum likelihood estimate of a parameter that characterizes a random variable Q, is to choose that parameter value that makes the observed sample values $Q_1, Q_2, ..., Q_n$ the most probable.

In our case the sample to be analysed is formed by sets of test intervals of size n (with n=5, 10, …, NTI), and we want to predict, as early as possible, the proportion Q of FTIs. We can visualize the sample as a sequence of Bernoulli trials with probability Q of failure on each trial (note that in such a way we are assuming independent TIs, which is reasonable for the approach followed in test selection). Thus, if the observed number of failed TIs is f, then the likelihood function l is given by [3, 4]:

$$l(Q) = Q^f (1-Q)^{n-f} \tag{3}$$

The *maximum likelihood* estimate of Q is that value of Q that maximizes the likelihood function l, or its logarithmic form. Solving for Q yields the maximum likelihood estimate:

$$\hat{Q} = \frac{f}{n} \tag{4}$$

It can be proved [3] that such $\hat{Q}$ is an unbiased, consistent, and the minimum variance unbiased estimator of Q.

To complete the statistical control procedure, we associate to each $\hat{Q}$ its *confidence interval*, that is a probability judgement about the accuracy of the estimate delivered. We are dealing with a random variable, so we cannot predict with certainty that the true value of the parameter, *Q*, is within any finite interval. We can, however, construct a confidence interval, such that there is a specified confidence or probability that the true value *Q* lies within that interval. For a given confidence level, of course, the shorter the interval, the more accurate the estimate.

It can be proved [3] that, for a sample of large size *n*, an approximate 100(1-α)% confidence interval for the Bernoulli parameter *Q* is given by:

$$\hat{Q} - z_{\alpha/2}\sqrt{\frac{\hat{Q}(1-\hat{Q})}{n}} < Q < \hat{Q} + z_{\alpha/2}\sqrt{\frac{\hat{Q}(1-\hat{Q})}{n}} \qquad (5)$$

where $\hat{Q}$ is obtained by (4) and values for the parameter $z_{\alpha/2}$ are found in statistical reference tables [4].

Therefore fixed a confidence level (90%) according to the producer exigencies, we associate to each $\hat{Q}$ estimate after *n* TIs, $\hat{Q}(n)$, the relative confidence interval.

The study of the confidence intervals leads us to determine that, after a certain number *n** of TIs, the desired level of accuracy is reached. Therefore we can use the estimate $\hat{Q}(n^*)$, obtained after n* TIs, to make predictions about the number of FTIs after (n*+5), (n*+10), …, NTI test intervals. In other words, after NTI test intervals, the number of failed test intervals N_{FTI} can be simply obtained by $N_{FTI} = NTI \bullet \hat{Q}(n^*)$ (see Eq. (1)).

To complete the prediction procedure, we can now apply Eq. (2) to obtain the global number of failures expected at the end of Function Test.

To assess whether the model inferences seem adequate we check, a posteriori, the obtained predictions against the real outcomes of several projects. Some examples of application are illustrated in Section 5.

The main limitation of this approach lies on the fact that a large amount of data is necessary to derive significant confidence intervals. Consequently the value n* of TIs guaranteeing the desired level of accuracy can result quite high.

4 Using a Bayesian Approach

The method described in the previous section has a broad field of application. We can use it whichever is the behaviour of the product under test. But this is also a limitation of the method, because it is not able to exploit the evidences of historical data collected over other TEI projects and which could contribute to better predict the cumulative number of failures.

For this reason, we investigated other methods which could be correlated to the behaviour of the product under test, with the purpose of reaching more accurate estimates or, more importantly, of anticipating the moment in which the predictions can be trusted. In this section, we report the application of a Bayesian approach.

We chose this kind of approach after an accurate analysis of the failure behaviour of several products. We observed in fact that every realization of the random variable

Q can only take discrete values of the form $\frac{1}{i}$ within an interval $\left[\frac{1}{M},1\right]$ (where M is a maximum fixed value). Precisely, for each i within [1, M], the associated discrete distribution, or probability mass function (pmf) of Q, $p_Q\left(\frac{1}{i}\right)=P\left(Q=\frac{1}{i}\right)$, gives the probability that the next FTI will be observed after *(i-1)* successful TIs. In particular, and more notably, we observed that *for all products* considered the pmf of Q always concentrated for most of its realizations on three same consecutive values, while took very rarely the other possible values. We thought that a Bayesian approach was the most effective way to exploit this knowledge (in the spirit of the approach described in [5]).

In the Bayesian framework [6], probabilities are meant to describe an observer subjective knowledge of yet-unknown events. This knowledge continuously evolves as new events are observed: inferences are drawn by combining the pre-existing knowledge with the new evidence collected through observation.

In our context, the observed behaviour of Q in the projects considered constitutes an important starting point to model a tester's subjective belief about the rate of failure detection during TEI Function Test. We express this knowledge through an appropriate modelling of $p_Q\left(\frac{1}{i}\right)$, the *prior* pmf. During the performance of Function Test, the realization of a sequence of test intervals with and without failures is observed. Thanks to this evidence, the tester's knowledge about *this specific product* evolves and a new distribution for the pmf of Q, called the *posterior* pmf, can be derived.

Denoting by F_n the sequence of observed outcomes (failed/successful) for the first n TIs, the posterior pmf for Q, denoted by $p'_{Q,n}\left(\frac{1}{i}\right)$, then gives $P\left(Q=\frac{1}{i}\mid F_n\right)$, i.e., it is the update of $p_Q\left(\frac{1}{i}\right)$ after having observed the sequence F_n.

Applying Bayes' formula we have:

$$p'_{Q,n}\left(\frac{1}{i}\right)=P\left(Q=\frac{1}{i}\mid F_n\right)=\frac{P_{prior}\left(Q=\frac{1}{i}\right)P\left(F_n\mid Q=\frac{1}{i}\right)}{\sum_{j=1}^{M}P\left(F_n\mid Q=\frac{1}{j}\right)P_{prior}\left(Q=\frac{1}{j}\right)} \quad \textbf{(6)}$$

in which the term $P\left(F_n\mid Q=\frac{1}{i}\right)$ is usually called a *likelihood function*.

With f denoting the number of FTIs observed in the sequence F_n, the likelihood function can be derived as a binomial distribution with parameters n and $\frac{1}{i}$. Substituting, the posterior distribution $p'_{Q,n}\left(\frac{1}{i}\right)$ hence is:

$$p'_{Q,n}\left(\frac{1}{i}\right) = \frac{p_Q\left(\frac{1}{i}\right)\cdot\left(\frac{1}{i}\right)^f\left(1-\frac{1}{i}\right)^{n-f}}{\sum_{j=1}^{M} P_Q\left(\frac{1}{i}\right)\cdot\left(\frac{1}{j}\right)^f\left(1-\frac{1}{j}\right)^{n-f}} \quad (7)$$

In the step 3.1 of our prediction procedure (Sect. 2.3), we use this updated pmf to derive $E_n[Q]$, i.e. the posterior expectation of Q after n TIs. This is taken as the estimator $\hat{Q}$ in Eq. (1) to derive $N_{FTI,n}$, i.e., the predicted number of FTIs expected after NTI test intervals, based on the test outcomes collected during the first n test intervals, *and* on the prior expectation about Q. From $N_{FTI,n}$ the expected number of failures can then be derived in the usual way.

In the next section we provide some examples in which this Bayesian model is compared with a Classical approach.

5 Examples and Discussion

We have so far presented a "textbook", Classical approach for predicting the expected number of failures in Section 3, and an alternative, Bayesian approach in Section 4. The second model was introduced not because it is claimed to be better than already existing methods, but because we hope it can prove more suitable to assess the specific TEI Function Test process. In particular, we think that since it exploits the prior available knowledge about the rate of occurrence of FTIs it needs fewer data to obtain valid predictions.

Indeed, deriving a prior distribution for the probability of interest is in general a difficult task, which also generates some perplexity towards the usefulness and applicability of Bayesian inference methods. In this case, the data available from several projects submitted to the same testing process conducted easily to an empirical distribution in which the rate of occurrence of FTIs concentrates within a strict interval.

In this section we provide a few examples of results obtained from use of the two described estimators. In the following figures we report the results obtained by applying the models to different data sets coming from the Function Test phase of large telecommunication systems. The size of the software varies from project to project (minimum 50 kloc, maximum 150 kloc), and the failures in the data sets considered were classified as priority B[1] (major failures).

In the following diagrams, on the horizontal axis we put the number of elapsed groups of TIs. On the vertical axis we put the cumulative number of failures over

[1] A failure is classified as priority B if it implies:
- large restart with or without reload;
- small restart;
- the function required for the operation and maintenance of the change is out of order;
- traffic disturbance on a single route, or for a few subscribers;
- a dominant PCB (printed circuit board) has a significantly higher failure rate than predicted;
- increased priority from level C for commercial reasons.

completion of the scheduled test period (for confidentiality reasons, we have to omit the actual numbers). We report a dashed curve for the Classical predictions, and a continuous curve for the Bayesian ones. When a prediction becomes acceptable (according to step 4 of the prediction procedure), we fix it and the curve becomes a straight line. The model check (step 6) is presented in the figures below by showing the actual number of failures observed at the end of the test period (the dotted horizontal line) (of course this knowledge is used in no way to make the prediction). The strip marked with vertical segments around the latter signs the zone where the relative error of the prediction would be below 10%.

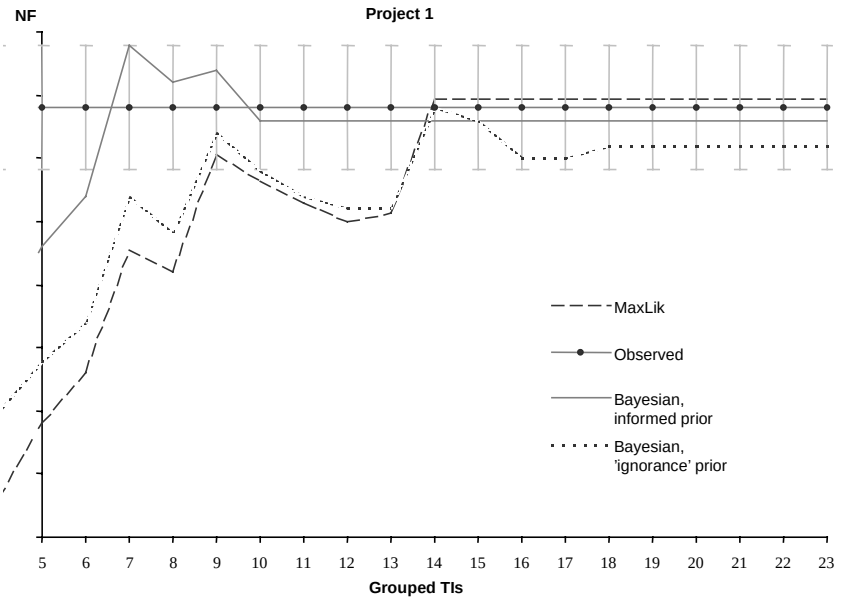

Fig. 1. Prediction results for Project 1

In Fig. 1 we show the results for Project 1. The maximum likelihood method produces a valid prediction after 14 groups of TIs. The light gray curve, labelled Bayesian "ignorance" prior, would be the output from the Bayesian model using as a prior pmf a uniform distribution, i.e., not exploiting any specific knowledge from the test process under observation. We see that with this uninformed prior we gain no particular advantage in using the Bayesian model; on the opposite, the prediction stabilizes only after 18 groups of TIs. However, considering the output from the same model with the informed prior pmf, the prediction is anticipated of as much as four groups relatively to the Classical model, that is a very good result from the manager's point of view. With regard to the outcome of prediction, both models produce accurate estimates.

In the second project considered (Fig. 2), the Bayesian prediction stabilizes three groups in advance with respect to the Classical method, and in addition we see that for this project the estimate produced with the latter is outside the 10% error strip.

But unfortunately the Bayesian model does not consistently work better for any data set. In the third example (Fig. 3), we see that the Classical model reaches first a stable prediction.

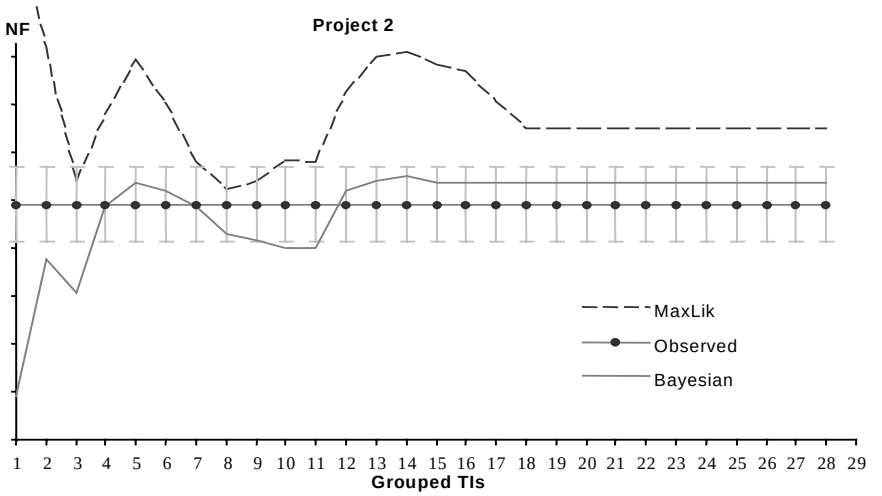

Fig. 2. Prediction results for Project 2

So, the idea is that we can apply both estimators in parallel, and use that one that first gives a valid prediction. We believe that in most cases this should be the Bayesian. However we are still collecting more data to continue the model validation.

The empirically found prior pmf for Q is a very useful result in itself. In particular, we believe it can provide testers with a very rough estimate of lower and upper bounds for the cumulative number of failures expected even before the Function Test phase starts. The quite regular typologies of behaviour showed by TEI projects under Function Test in fact permits us not only to know which will be the interval $\left[\frac{1}{i}, \frac{1}{i+1}\right]$ that includes the actual value of E[Q], but also to establish a good approximation value for E[F]. These numbers used in the formulas (1) and (2) provide us with rough bounds for the cumulative number of failures. For instance, considering a certain class of projects and if the Function Test process is planned to last for 100 TIs, we can estimate prior to starting the test process that the expected cumulative number of failures at the end of the test will be within [38, 50]. With evidence collected over more projects, we expect to be able to decide historical failure effectiveness densities, to be used as reference in process control.

6 Conclusions and Future Developments

We have presented an ongoing experience aimed at introducing statistical control over TEI Function Test processes. So far we have outlined the procedures for failure data collection, and the statistical models for interpreting the data. In particular, we

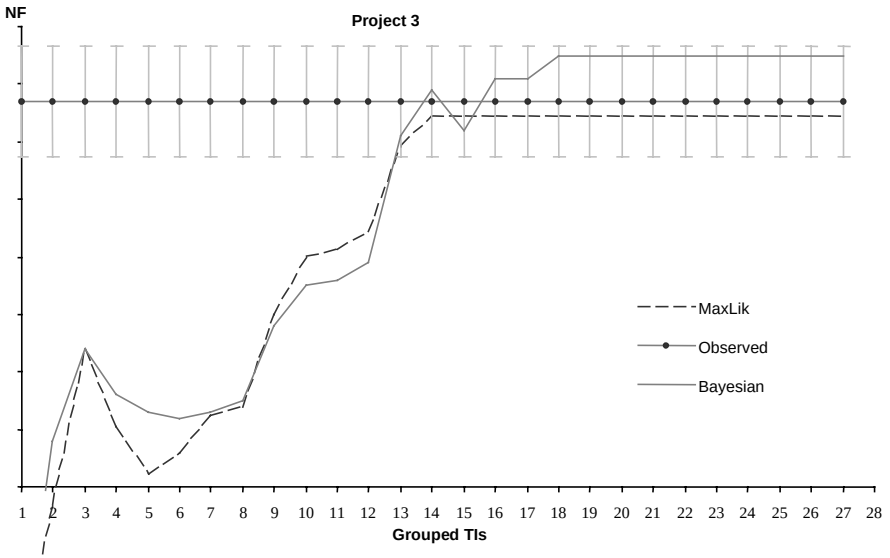

Fig. 3. Prediction results for Project 3

have introduced a Classical approach, based on the maximum likelihood estimate, and a Bayesian model, incorporating empirically defined prior distributions for the rate of detection of failures. We have already examined both models on several projects in a preliminary study, and now plan to introduce the models within the standard test processes, so to make a more comprehensive validation. A key aspect in the definition of the models was not to require any unnecessary additional work from testers, and viceversa to adapt as much as possible the models to the existing procedures.

The objective of introducing statistical control techniques within TEI is to achieve the capability to take decisions and then actions with desirable and predictable effects. We have discussed the application of statistical models to predict the expected number of failures over the planned period of Function Test. From our perspective, such models provide the project management team with an effective and not expensive means to take corrective actions when causes of variation are identified with respect to the Function Test process performance baselines (e.g., minimum and maximum fault density computed on historical data in the same product line), and with respect to meeting TEI slipping through objectives.

The implementation of appropriate corrective actions (such as executing extended Basic Test in parallel to Function Test, or postponing the end date of Function Test) can mitigate the risk of failures slipping through Function Test to first six months in operation, thus reducing rework and maintenance cost. Pilot projects to identify the most effective analysis technique, and to perform cost benefit analysis for the mitigation risk strategies connected to the application of the model are planned. In fact, no extra costs are required to collect data and apply the statistical models, as data collection is executed according to the standard procedure. To allow the easy use of statistical models within projects, a tool to be incorporated in the test environment will be developed.

The pilot projects will be managed according to a *Function Test Measurement Plan*. The Measurement Plan has the purpose to transfer baseline, models, practices approved by the organisation to the projects and then to bridge project's performance to organisation's capability. Very briefly, this plan includes:

- organisation objective and capability baseline,
- analysis technique and tools to be used,
- database where to store project's measurement collection,
- mitigation risk strategies according to objective uncertainty guideline,
- Return on Investment analysis.

References

1. Paul, M.: CMM v2.0 draft C, 22 October 1997
2. Fenton, N.E., Pfleeger S.L.: Software Metrics A Rigorous and Practical Approach. International Thomson Computer Press (1998)
3. Allen, A.O.: Probability, Statistics, and queuing theory with computer science applications. Academic Press (1990).
4. Loyd,E.: Handbook of Applicable mathematics: Statistics. Vol. III, IV, John Wiley & Sons (1987.)
5. Bertolino, A, Strigini, L.: Predicting software reliability from testing taking into account other knowledge about a program. In Proc. Quality Week '96. San Francisco (1996).
6. Gelman, A., Carlin, J.B., Stern, H.S., Rubin, D.B.: Bayesian data analysis. Chapman & Hall (1995)