How Many Individuals to Use in a QA Task with Fixed Total Effort?

Mika V. Mäntylä
Lund University
Department of Computer Science
22100 Lund, Sweden
mika.mantyla@cs.lth.se

Kai Petersen
Blekinge Institute of Technology
School of Computing
37140 Karlskrona, Sweden
kai.petersen@bth.se

Dietmar Pfahl
Lund University
Department of Computer Science
22100 Lund, Sweden
dietmar.pfahl@cs.lth.se

ABSTRACT
Increasing the number of persons working on quality assurance (QA) tasks, e.g., reviews and testing, increases the number of defects detected – but it also increases the total effort unless effort is controlled with fixed effort budgets. Our research investigates how QA tasks should be configured regarding two parameters, i.e., time and number of people. We define an optimization problem to answer this question. As a core element of the optimization problem we discuss and describe how defect detection probability should be modeled as a function of time. We apply the formulas used in the definition of the optimization problem to empirical defect data of an experiment previously conducted with university students. The results show that the optimal choice of the number of persons depends on the actual defect detection probabilities of the individual defects over time, but also on the size of the effort budget. Future work will focus on generalizing the optimization problem to a larger set of parameters, including not only task time and number of persons but also experience and knowledge of the personnel involved, and methods and tools applied when performing a QA task.

Categories and Subject Descriptors
K.6.3 [**Software Management**]: *Software process*

General Terms
Measurement, Economics, Human Factors, Management

Keywords
Effectiveness, Fixed effort Budget, Effort, Review, People

1. INTRODUCTION
Given enough eyeballs, all bugs are shallow, is known as the Linus Law stated by Linus Torwalds [1]. The statement claims that if we increase the number of people performing quality assurance (QA) tasks we find an increasing number of bugs and if we have the possibility to add people endlessly finally all bugs will be found. Whether this statement is completely true is debatable. However, it illustrates the fact that using a larger group of people in a QA task increases the number of defects found in comparison with a smaller group. For example, data by Jones [2] indicates that beta-testing is the most effective QA measure when a high number of sites is available (>1000). Furthermore, research shows that having large groups can be beneficial, e.g. in data of [3] from software inspections, we can see that the number of defects found increases when adding more inspectors even after 20 people. We witnessed in our previous research a similar pattern with manual software testing [4].

ESEM2012, Sep 17-18, 2012, Lund, Skåne, Sweden.

However, the problem with using large groups in QA tasks is the increasing personnel cost, but one can control this problem by limiting the effort budgets for QA tasks. The question to be answered when doing this how to divide the effort. For example, assume we have an effort budget of 10 person-hours for doing a software review. Then how many people should we use? Should we have one person working for ten hours or ten persons working one hour? Questions of this nature have received limited attention in the prior research on software testing and reviews, which focused more on the different techniques and tools to use.

In this paper, we continue our previous work [4] on understanding how many individuals to use in a QA task when having a fixed effort budget. In this paper, a QA task is any task where the primary goal is to find faults in a product under scrutiny. Section 2 presents the relevant prior work. Then, in Section 3, we discuss implications and present extension based on prior work. Section 4 models defect detection as a function of time, by first formulating defect detection with fixed effort budget as an optimization problem, and then applying this optimization problem to experimental data. Finally, Section 5 discusses the results and possible future work. Section 6 presents conclusions.

2. PRIOR WORK
In prior work, Biffl et al. describe how inspection team performance can be statistically estimated from individual inspector performances [3, 5]. For example, assume we have performed an experiment *A* with 40 participants and 10 of them found a particular defect *d1*. Then the detection probability for this defect is 0.25 on average for a single individual picked randomly from that population. Furthermore, if we pick two individuals then what follows from is that the detection probability for the particular defect is $0.4375 = 1 - (1 - 0.25)^2$.

We can also pick individuals from populations using different techniques and combine results as originally suggested by Biffl et al. This idea can be extended to other populations as well, e.g., ones having different time budgets, or having different experience. In Section 4 of this paper we discuss the case of fixed time budgets. To illustrate the case of using different techniques, let us assume we perform an experiment *B* with 40 participants – but using a different technique than in experiment A – and this time 20 individuals find defect *d1* suggesting an average detection probability of 0.5. Then, from this we can calculate the detection probability of a group consisting of one inspector from each population *A* and *B* as $0.625 = 1 - (1 - 0.25)^1 * (1 - 0.5)^1$. In more formal terms, the probability *P(d)* that a group of size *n* finds a given defect *d* is calculated as follows:

(1) $$P(d) = 1 - \prod_{ps \in \{1,\dots,Pops\}} (1 - p_{ps})^{n_{ps}}$$

In the formula above, *Pops* is the number of all populations from which group members are picked, *ps* is the index of a specific population, n_{ps} is the number of group members picked from the population with index *ps*, and p_{ps} is the average defect detection probability for defect *d* of group members picked from the population with index *ps*. Note that the sum over all n_{ps} equals *n*. In this paper, the binomial distribution is used to approximate the hypergeometric distribution when sampling from two populations. In more complex settings, the multinomial distribution could be used to approximate the multivariate hypergeometric distribution (assuming large sized populations).

The expected total number of defects detected can be calculated by summing up the defect detection probabilities of each defect. For example, assume that from the population A, 20 individuals found another defect called *d2*. Then for a single individual picked out of population A the expected performance is to find 0.75 defects (0.25 (d1) + 0.5 (d2)) and for groups of two individuals we expect them to find 1.1875 defects, calculated as the sum of $0.4375 = 1 - (1 - 0.25)^2$ and $0.75 = 1 - (1 - 0.5)^2$. Again, in more formal terms, for a given number of existing defects D, the expected number of detected defects E := Exp(D) can be calculated according to [3, 5] as follows:

$$E = \sum_{d \in D} P(d) \quad (2)$$

In our previous paper [3], we investigated time-restriction in software testing and found that two time-restricted testers with a 2-hour time slot found the same amount of defects as a single tester with no time-restriction, using almost 9.83 hours on average. The results of the study indicate that going slow and being thorough, i.e. using more time, is a good strategy for achieving high defect detection effectiveness for single individual. However, the study also indicates that this is not necessarily true if we consider groups of individuals with time budgests. In fact, we found that if we pool 5 individuals each using 2h testing, i.e., adding up to a total of 10 person-hours effort, the nominal 5-person groups find 71% more defects than a single individual using 9.83 hours of time.

3. IMPLICATIONS AND EXTENSION BASED ON PRIOR WORK

One surprising implication we can deduce from the work and equations proposed by Biffl et al. [3, 5], which they did not mention themselves, is that we should not use the average number of defect detected or any statistical test based on it, e.g. *t-test*, to reason about defect detection performance differences between group. The reason for this is that it uses only the *number* of defects detected per participant and thus ignores the detection probabilities of individual defects. Furthermore, we cannot calculate group performances unless we do know defect detection probabilities for each individual defect. For example, if we consider QA techniques A and B, and experiments show that they both detect 50% of the defects in a given set of four defects, then a classical t-test comparing the average number of defects detected would reveal no difference. Let us further assume that technique A has the detection probabilities of (0.95, 0.05, 0.95, 0.05) for the four defects and that technique B has the detection probabilities of (0.5, 0.5, 0.5, 0.5), i.e., both techniques detect 2 defects on average. Now, if we have a group of two people both using either technique A or B, we find that using technique B is superior. The expected performance of technique B for a group of two people is detecting 3 defects. This can be calculated using equations (1 and 2) in Section 2, i.e., $3 = 4 * (1 - (1 - 0.5)^2)$. However, when using technique A, the expected number of defects found would be $2.19 = 2 * (1 - (1 - 0.95)^2)) + 2 * (1 - (1 - 0.05)^2$. In other words, while the average total number of defects detected by a single individual revealed no differences between techniques A and B, the situation in the group settings changes towards favoring the technique that has smaller variation between the defect detection probabilities of individual defects, in our example technique B.

Our decision problem is related to research about the relationship between review speed (or review rate) and defect detection probability, as studied also by Kemerer and Paulk (see fig 6 in [6]). In [6], one individual reviewing 200 LOC/h (or slower) finds 59.2% and an individual reviewing 400 LOC/h (or faster) 50.0% of all defects. To make the effort budget comparable, the setting would be to compare one person reviewing at a speed of 200 LOC/h against two persons reviewing at 400 LOC/h. Ideally, two fast reviewers together with a defect detection effectiveness of 50% each could find up to 75.0% (i.e., $(1 - (1 - 0.5)^2)$ of all defects. Based on this analysis, using two fast reviewers instead of one thorough reviewer seems to be promising.

We can improve the estimate of the defect detection effectiveness of two individuals by using empirical data. In [3] we studied 13 empirical data sets to determine the average increase in defect detection effectiveness when using two individuals instead of one. We found that across all data sets the number of defects detected by two persons is on average 73.6% (range: 59%-89%) of the theoretical maximum minus the theoretical minimum. Applying this average to the data of Kemerer and Paulk, where two fast reviewers (reading at a speed of 400 LOC/h) have a theoretical minimum of 50% defect detection probability (e.g., if both reviewers happen to find exactly the same defects) and a maximum of 75%, we could predict that two fast reviewers find 68.4% (i.e., 0.736*(0.75-0.5)+0.5) of all defects.

4. MODELING DEFECT DETECTION OVER TIME

The defect detection probability of a given defect *d1* can be understood as a function of time, derived from the defect detection times of individuals performing a QA task. For example if we have 40 reviewers and 10 of them find defect *d1* then the detection probability of *d1* is 0.25 after they all individuals have completed their reviews. At the beginning of the review time is zero (*t=0*) and so is the detection probability of *d1* (*d1p=0*). As time passes, *d1p* changes from 0 to its maximum of 0.25. In Figure 1, we illustrate this by presenting how the defect detection probabilities of two defects (D5 and D36) change over time based on data we got from a previous experiment [7].

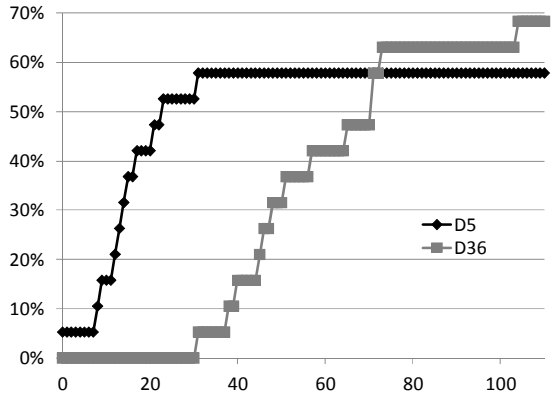

Figure 1. Detection percentages of two defects (D5 and D36) over the course of an inspection

Based on the graph we can now answer questions like the following: How many defects, on average, would 2 individuals find, if they both used half of the given time $2 * {}^{t}/_{2}$? Depending on the graph, the performance of the two individuals using 50% of the time could be worse or better than that of one individual using 100% of the time. In the illustrated case, considering only the two depicted defects, a single individual finds on average 1.21 of those defects when t=100 min. Two individuals using $2 * {}^{t}/_{2}$, i.e. 50 min each, would find 1.35 defects.

4.1 Formalization in mathematical terms

If *n* is the number of individuals working independently on a QA task and *t* is the time used by each individual to perform the QA task, then we can formulate an optimization problem that aims at finding the largest defect detection effectiveness, expressed in terms of the expected number of defects *E(n, t)* found by all individuals *n* in time *t* with fixed effort budget $t * n = c$, where *c* is a constant, as follows:

$$E(n,t) = \sum_{d \in \{1,\dots,D\}} \left(1 - \left(1 - p(d,t)\right)^{n}\right) \rightarrow max \quad (3)$$

with:

- $n \in \{1, \dots, N\}$, with *N* maximum number of individuals,
- $t \in (0, T)$, with *T* maximum duration of QA task,
- $d \in \{1, \dots, D\}$, with *D* total number of defects,
- $p(d,t) \in [0,1]$, average probability of detecting defect *d* at time *t* by any individual,
- $t * n = c$ with *c* is a constant effort budget

Since the probability *p(d,t)* that a defect *d* is found by an individual within a time period of length *t* is a continuous function over time which we cannot derive analytically from a corresponding mathematical formula, we must base the calculations of optimality on empirical data, similar to that shown in Figure 1.

4.2 Application using empirical data

To illustrate our idea, we use a data-set from a previous experiment [7] were inspection techniques, time-controlled reading and usage-based reading, were studied in an experiment involving 19 students who detected in total 31 defects. In that study, no statistically significant differences between the compared techniques were found. Therefore, for our study, we pooled the data and treated it as one data set. It was important for our study that the original experiment recorded the exact time in minutes when each defect was found by each individual. This allowed us to construct figures like Figure 1 and applying the formulas presented in Section 0.

In the original experiment, the time was split into preparation time (40 min used in average) and inspection time (125 min used in average). During the preparation time, the students were instructed to do an overview reading of the inspected document, but also to read instructions on the inspection techniques that were tested. This preparation time would be shorter in the industrial context when inspectors would already be trained in a given technique and familiar with the product. Thus, the long preparation time in the student case represents a situation when beginners come to inspect a product they know nothing about. Since we do not know the values for more realistic preparation time we present the following extremes cases: *case 1* where only the inspection time is considered, and *case 2* where the preparation time is added on top of the inspection time. The maximum inspection time used by an inspector was 125 minutes and the maximum inspection + preparation time was 165 minutes.

Table 1. Defect detection effectiveness (= expected number of defects E)

	Case 1: fixed budget of 125 min (inspection time only)			Case 2: fixed budget of 165 min (inspection+preparation time)		
n	t/n [min]	E(n,t)	t [min]	t/n [min]	E(n,t)	t [min]
1	125	9.16	125	165	9.21	165
2	62	9.78	124	82	6.70	164
3	42	7.89	126	55	3.94	165
4	31	7.25	124	41	1.66	164
5	25	6.43	125	33	0.24	165
6	21	6.33	126	27	0.28	162
7	18	6.21	126	24	0.32	168
8	16	6.02	128	21	0.35	168

We used the maximum inspection times of each case as our base values for the fixed effort budgets, i.e., 125 min for case 1, and 165 min for case 2. Table 1 and Figure 2 show the number of expected defects *E(n,t)* for case 1. From the table, we can see that for case 1 the optimal configuration is to use 2 inspectors who split the time budget of 125 min. Choosing 3 or more inspectors results in declining performance. When we use the multiples of the base time budget of 125 min, as shown in Figure 2, we can see that in all cases the optimal number of inspectors is n+1 when n is the minimum number of inspectors that could be used to consume the effort budget. Also, with effort budgets from 250 min to 500 min, n+2 inspectors perform better than n inspectors.

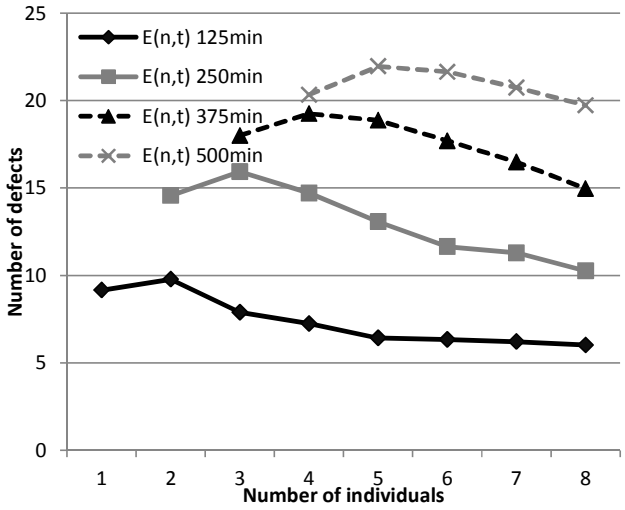

Figure 2. Defect detection effectiveness with fixed budgets of 125, 250, 375, and 500 min (case 1: inspection time only).

Table 1 and Figure 3 show the results for case 2 where both preparation and inspection times are taken under consideration. The table shows how using a single individual is superior when the preparation time is accounted for and having the smallest time budget of 165 minutes. However, from Figure 3, we can see that using the smallest possible number of inspectors is not beneficial for the time budgets of 495 minutes and 660 minutes and in such cases it would be the best to use n+1 inspectors when n is the minimum number of inspectors that could be used to fill the time budget.

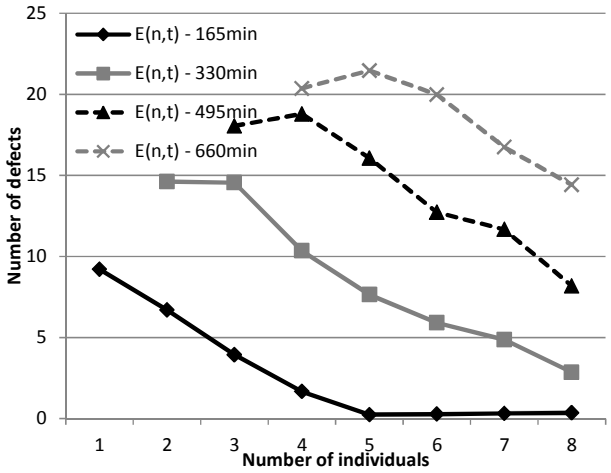

Figure 3. Defect detection effectiveness with fixed budgets of 165, 330, 495, and 660 min (case 2: inspection+preparation time).

5. DISCUSSION AND FUTURE WORK

This paper describes our ongoing work on trying to understand how many individuals to use in QA tasks where the primary goal is defect detection, e.g. inspections or testing, when having a fixed effort budget.

Prior work had indicated that using several fast readers could be more beneficial than using a single thorough reader. In this work, we could not find support for this finding. We would like to point out that our usage of the data is purely illustrative since it did not contain actual data of fast readers, i.e., readers forced to work under defined time-restrictions. To compensate the lack of data on actual fast readers we used the defect detection probabilities of uncompleted reviews. In the future, we plan to fix this shortcoming, by using the same experimental material as in [7], but by applying time-pressure and by forcing the students to perform the whole reviewer in a shorter time and then compare the results with data of previous experiments.

Whether to use multiple readers with shorter individual total time is context dependent and, in this paper, we could see how the number of reviewers that should be assigned changed whether we included or excluded the individual preparation time, and with different effort budgets. In the future, one should look at how much preparation time is actually needed in industrial settings. Furthermore, plenty of mathematical optimization techniques have already been applied in other areas of software engineering such as release planning [8] and they can undoubtedly be applied to this optimization problem as well.

As a next step, the optimization problem presented here will be generalized to include cases where not all inspectors are given an equal time slot, e.g. one could divide 90 minutes into three reviewers by giving one reviewer 60min and two others 15min each. We could also add other dimensions to the optimization problem. For example, higher expertise would make of individuals better in detecting defects, but in industrial setting people with higher expertise would also be more expensive to use. Thus, in such case, the total budget would be monetary and there would be tradeoff in choosing between more and less expensive individuals and the time their use. Furthermore, defect detection techniques could represent yet another dimensions that could be added to the optimization equation.

Furthermore, we aim to generalize the optimization problem to help design QA processes with stages. For example, would it be beneficial to first have one fast reader to find the easy defects, and then have a pair of thorough readers to dig out the defects that are more difficult to find? Such generalization would require that the defect detection rate of a subsequent task depend (partly) on the defect detection effectiveness of the predecessor task. This line of work could lead to giving practical recommendations for designing industry QA-processes based on solid empirical research.

6. CONCLUSION

In this paper we have made three contributions. First, we have described how defect detection probability should be understood as a function of time. Furthermore, we have formulated it as an optimization problem and showed the results of using the formulas on previously collected empirical data set of [7]. Second, based on this we have shown numerous avenues for future work in Section 5. Third, we showed that using number of defects detected per individual and statistical tests relying on such numbers, e.g. *t-test*, should not be used to reason between different techniques in-group settings. This is because defect detection probabilities of individual defects are needed to study group performance. It is still correct to use the number of defects detected per individual if one is only interested in the performance difference between singles. However, we believe this is actually rarely the case as software development and QA are often collaborative activities.

7. ACKNOWLEDGMENTS

This work has been supported by ELLIIT, the Strategic Area for ICT research, funded by the Swedish Government.

8. REFERENCES

[1] E. Raymond, "The cathedral and the bazaar," Knowledge, Technology & Policy, vol. 12, no. 3, 1999, pp. 23-49.

[2] C. Jones, "Software defect-removal efficiency," Computer, vol. 29, no. 4, 1996, pp. 94-95.

[3] S. Biffl and M. Halling, "Investigating the defect detection effectiveness and cost benefit of nominal inspection teams," Software Engineering, IEEE Transactions on, vol. 29, no. 5, 2003, pp. 385-397.

[4] M.V. Mäntylä and J. Itkonen, "The Effect of Adding People and Restricting Time in Software Testing – Power of the Crowds," Submitted to a Journal, Under review,

[5] S. Biffl and W. Gutjahr, "Influence of team size and defect detection technique on inspection effectiveness," Software Metrics Symposium, 2001. METRICS 2001. Proceedings. Seventh International, 2001, pp. 63-75.

[6] C.F. Kemerer and M.C. Paulk, "The impact of design and code reviews on software quality: An empirical study based on PSP data," IEEE Trans.Software Eng., vol. 35, no. 4, 2009, pp. 534-550.

[7] K. Petersen, K. Rönkkö and C. Wohlin, "The impact of time controlled reading on software inspection effectiveness and efficiency: a controlled experiment," Proceedings of the Second ACM-IEEE international symposium on Empirical software engineering and measurement, 2008, pp. 139-148.

[8] G. Ruhe and M.O. Saliu, "The art and science of software release planning," Software, IEEE, vol. 22, no. 6, 2005, pp. 47-53.