

Usability Testing: What Have We Overlooked?

Gitte Lindgaard

Human-Oriented Technology Lab
Carleton University
Ottawa, Canada
gitte_lindgaard@carleton.ca

Jarinee Chattratchart

Faculty of Computing, Information Systems and
Mathematics, Kingston University
Surrey, United Kingdom
j.chattratchart@kingston.ac.uk

ABSTRACT

For more than a decade, the number of usability test participants has been a major theme of debate among usability practitioners and researchers keen to improve usability test performance. This paper provides evidence suggesting that the focus be shifted to task coverage instead. Our data analysis of nine commercial usability test teams participating in the CUE-4 study revealed no significant correlation between the percentage of problems found or of new problems and number of test users, but correlations of both variables and number of user tasks used by each usability team were significant. The role of participant recruitment on usability test performance and future research directions are discussed.

Author Keywords

Usability testing, Metrics, UEM, participant recruitment.

ACM Classification Keywords

H5.2. Information interfaces and presentation (e.g., HCI): User Interface

INTRODUCTION

Usability testing is costly. Traditional usability tests are conducted with one test participant at a time. Each additional session adds to the running cost. One way to keep the cost down is therefore to run an optimum number of sessions, i.e. high enough to reveal most – hopefully severe – problems but not too high to keep the running costs manageable. The issue of determining the optimum number of users has attracted a lot of attention over the past 15 years and has generated much debate among practitioners and academics alike.

Our aim is *not* to reinvent the wheel in this debate but to move the discussion beyond the issue of ‘optimum number of users’. We question if the usability field should shift its focus from the number of users to task design, task coverage, and the number of tasks, all of which also

play a major role in usability evaluation. We review the literature briefly first in an effort to show that the assumption of ‘one number fits all tests’ is not warranted in the present state of UEM (Usability Evaluation Method) research and demonstrate why task design merits greater attention. We then examine the data from the nine usability test teams that took part in the CUE-4 study [8], discuss the statistical results from this exercise, and make recommendations based on our findings.

RESEARCH BACKGROUND

Sample Size

“The Magic Number 5” was the name of a panel at the CHI 2003 conference intended to be ‘the last panel of its kind and to lay the issue to rest for once and for all time’ [3]. The number 5 stems from an assertion [16, 13] that 5 users suffice to reveal 80% and 85%, respectively, of all the problems that exist in the interface under evaluation. These claims earned their credibility on the basis of the formulae derived from probability theories as well as from empirical data. Virzi [16] applied a Monte Carlo procedure to three user tests he conducted and showed that there was an asymptotic relationship between the number of users and the proportion of all problems found and that this relationship can be approximated by the formula

$$\text{Proportion of problems uncovered} = 1 - (1 - p)^n,$$

where p is the probability of detecting a problem and n is the number of users. In the following year, using a Poisson distribution and data from five user tests and six heuristic evaluations, Nielsen & Landauer [14] presented a similar formula,

$$\text{Found}(i) = N(1 - (1 - \lambda)^i),$$

where λ is the probability of detecting a problem, N is the total number of problems, and i is the number of users. Both formulae allow the number of users (or heuristic evaluators if heuristic evaluation is used) to be predicted for a known p or λ and a predetermined value of proportion of all problems found, e.g. 0.85. The cost-benefit analysis presented in [14] showed the optimum number of test users to be 7, 15, and 20 for small, medium-large, and very large projects, respectively, and that 3.2 test users would yield the highest benefit-to-cost

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CHI 2007, April 28–May 3, 2007, San Jose, California, USA.
Copyright 2007 ACM 978-1-59593-593-9/07/0004...\$5.00.

ratio. Later, Nielsen [13] claimed that by using the formula with $\lambda = 0.31$, one only needed to run 5 users to find 85% of the problems, explaining why ‘elaborate usability tests are a waste of resources’. This magic number 5 was quickly adopted by many organizations as the optimum sample size for a usability test that would yield an acceptable level of Return On Investment (ROI) [3]. Over time, however, others discovered that the reality was far from what had been claimed. For example, Spool & Schroeder [15] reported that the first five users in their 49 tests of four e-commerce web sites found only 35% of all the problems, yielding a λ value of 0.082. Another example is Faulkner’s [4] well-designed study showing that 5 users might find as little as 55% of all problems. The range of λ values reported in the literature varies from 0.08 to 0.51 [e.g. 14, 15, 16]. Applying a fixed λ value at 0.31 to all usability tests, regardless of the complexity of the interface, could overestimate the proportion of problems found, and hence, underestimate the sample size required.

In addition to the explicit statements made about the λ value, another implicit assumption could explain the counter-evidence reported in the literature. The formulae assume that the number of all problems in the interface being evaluated is known, that it either equals the total number [16], or the average value of total number of problems uncovered in every test from which the values were derived [14]. However, Molich’s series of CUE studies [8, 10, 11, 12] provide evidence suggesting that it is unlikely that anyone knows the figure representing ‘all problems’ that exist in an interface. Both the CUE-2 and CUE-4 studies revealed little overlap in the problems reported by different usability test teams evaluating the same interface:

CUE-2: 75% of the 310 problems was uncovered by a single usability test team; 8 teams evaluated the same web site testing a total of 53 participants [10].

CUE-4: 67% of the 237 problems was uncovered by a single usability test team; 9 teams evaluated the same web site testing a total of 76 test participants [8].

The high percentages of problems reported by only one team suggest that the test teams were nowhere near finding ‘all’ usability problems and that consequently, the magic number 5 may have been grossly underestimated.

Without making the assumption that ‘all’ problems were known or using the problems found in their study as ‘all’ problems found, Jacobsen, et al. [7 cited in 6] suggested that the total number of problems found be estimated from the geometric mean of the number of users and evaluators:

Number of problems found

$$= C\sqrt{(\text{number of evaluators} \times \text{number of users})}$$

The formula was derived from a small database of problems reported by 4 evaluators analyzing the same set of video tapes of 4 users in an evaluation using the think-aloud method. This could be applicable to usability testing because a typical usability test also uses the think-aloud method, one or two evaluators, and more often than not, a small sample of users.

All three abovementioned formulae show some relationship between the number of problems found and the number of test users. This relationship may hold within a particular evaluation or for the same usability test but will it hold across different tests where the value of λ varies widely? If a constant λ is adequate for generalizing (such as the magic number 5), these two variables should correlate across different evaluations of the same interface.

User tasks

Is the number of users worth the paramount attention it has received for nearly 15 years? The outcome of a usability test depends on many attributes such as participants’ characteristics, test objectives, task design, individual differences, problem criteria, skills of usability testers, etc.

Variations in user tasks have consistently been used to explain why different evaluators or test teams found different sets of problems. Hertzum & Jacobsen’s [6] study on evaluator effect revealed that the average agreement between two evaluators assessing the same interface using any one of the three UEMs reviewed (cognitive walkthrough, heuristic evaluation, and think-aloud) ranged from 5% to 65%. Their analysis revealed that this was due to vagueness in the goal analyses, evaluation procedures, and problem criteria.

A goal analysis includes clear and justified evaluation goals and careful selection of tasks. The evaluators in Hertzum & Jacobsen’s review [6] found different sets of problems in the same evaluation because of wide variations in the tasks: some were mentally simulated by evaluators using an inspection method, others tested user tasks in a think-aloud session. Additional evidence showing that task variety may be important was provided by Cockton & Woolrych [2] who conducted two usability tests of the same interface to assess the value of the heuristic evaluation method. The second test was designed to confirm the false alarms indicated by the first test. A different set of user tasks employed in the second test revealed several new problems. Clearly, additional user tasks can uncover new problems. Additionally, results from the CUE-2 study revealed poor agreement among the test teams. The usability test teams in that

study used 51 different user tasks, 25 of which (49%) were used by only one team. To this effect, Molich, et al. [10] stated, ‘the large number of tasks may have affected the number of findings, however not significant as one may think’.

The Magic Number 5 panel at CHI 2003 did not seem to have put an end to the sample size issue. A number for ‘all’ problems in an interface does not seem to provide a viable baseline for UEM comparison. The success of a usability test depends on numerous factors, many of which cannot be controlled. Reports of findings contradicting the original sample size claims and evaluator effect continue to appear in the literature and ‘...usability testing now appears to be a highly variable art in which the results depend on who is testing what by which protocol with which particular subjects’ [3]; a black art indeed!

RESEARCH QUESTIONS

The two main themes of our discussion so far are sample size and user tasks, which influence evaluation results but which can be manipulated for research purposes and for good usability test design. We questioned the claimed relationship between evaluation results and sample size or tasks, which gave rise to two hypotheses tested here:

1. That there is a correlation between number of users and the proportion of problems found.
2. That there is a correlation between number of user tasks and the proportion of problems found.

RESEARCH METHODOLOGY

Rationale

Rolf Molich kindly made the reports from the usability teams participating in the CUE-4 workshop at CHI 2003 publicly available [9]. Of the 17 professional usability teams participating in the CUE-4 study, nine conducted usability tests, using a total of 76 test users; the rest used a variety of inspection methods. We chose to analyze only the usability test data because many confounding variables were well controlled:

1. All usability test teams were highly experienced professionals.
2. All evaluated the same web site.
3. All used the think-aloud method.
4. The evaluations were conducted and completed at about the same time.
5. All received the same instructions prior to the evaluation thereby sharing many usability test attributes such as:
 - evaluation objectives

- problem criteria
- evaluation focus – on the OneScreen reservation system of the web site
- user tasks in which the development team, iHotelier (the supplier of the OneScreen reservation system), was most interested
- reporting format

We employed the factors that were not controlled, namely sample size and user tasks, as independent variables in our study. Other benefits of using the CUE-4 data include the fact that the Hotel Penn web site was, and still is, a fully functional commercial site; the evaluations had good external validity and were professionally conducted.

Obtaining Raw Data

We downloaded the CUE-4 study files from Molich’s web site (<http://www.dialogdesign.dk/cue.html>) containing reports from all 17 teams. As stated earlier, we analyzed only those of the nine usability test teams: Teams A, H, J, K, L, M, N, O, and S.

Data Analysis

It took three major steps to obtain the data required for the statistical analyses:

1. Identify the number of users (participants) in the nine usability tests.
2. Analyze the user tasks and scenarios.
3. Analyze the problems reported by each usability test team.

Number of users

All teams reported the number of users who performed their tasks. These varied from five to 15 as shown in Table 1. All used the think-aloud method and one user per session, with the exception of Team A in which a couple who usually make hotel bookings together participated together in one of the seven sessions.

Team	A	H	J	K	L	M	N	O	S
Users	6	12	7	5	6	15	13	6	6

Table 1. Number of users in each team.

Analysis of user tasks

A review of the nine reports revealed a wide range of user tasks varying in length from one short sentence to a whole page, presented either in tabular or in narrative form and mostly did not report the task-selection criteria. Same task goals were often present in different scenarios, many task goals in one scenario, and differences were found in the questions, the length of instructions, and so forth.

Team A presented 13 tasks in a table. The selection of each task was explained, clearly exhibiting task goal analysis.

Team H gave 9 tasks in a paragraph format without any explanation (task goal analysis).

Team J described 6 task scenarios, one by one, each accompanied by a summary of the results.

Team K gave a list of 12 tasks, mostly with short instructions or a question but no task scenario, e.g. 'Is there a room available for the night of January 4th, 2004?'

Team L listed 10 tasks. They had been given a copy of Team A's tasks but chose to use only nine of these of which two were modified and one was L's own task.

Team M also listed 10 task scenarios with task goal analyses. For each task, the starting point, ending point and success criteria were clearly defined.

Team N used 4 tasks written in a paragraph format. Each had a title, e.g. 'Family Weekend in New York' (task 1) or 'Filling out a Reservation Form and Booking a Room' (task 4).

Team O listed a combination of 16 short task descriptions and scenarios in paragraph format. Tasks 14, 15, and 16 were to be completed 'if schedule permits'.

Team S provided a persona for the user to imagine himself/herself and 8 task scenarios in a paragraph format.

Evidently, the number of tasks listed in the report could not be analyzed statistically in their raw form. For example, the title of Team N's task 4 suggested that there were at least two task goals: *filling a reservation form* and *booking a room*. As this was not the only example of a composite task, it was necessary to define 'user tasks' as well as analyzing the tasks in more detail:

1. A task goal is the highest level in a task hierarchy. It consists of one or more user tasks. The user conducts a task with a goal in mind, for example, *making a reservation, filling in a reservation form, finding information about the hotel*, and so on.

2. A user task belongs to a particular task goal but with a condition or situation attached. A user seeking to accomplish a particular task goal in a variety of contexts will likely interact differently with the system and hence encounter a different set of problems. For example:

Task goal: *Find an available room*

User tasks:

- *Check room availability of a particular room type on a certain date.*
- *Check room availability for the following year*

- *Check room availability for rooms under a budget*

Task goal: *Make reservation*

User tasks:

- *Reserve a non-smoking room*
- *Reserve a family room*
- *Reserve a particular room type*

3. A task token is the smallest unit of task, which cannot be broken down any further. It usually occurs in a very specific context, e.g., *going back to the home page*, and *making a reservation for a family of three from June 28 to July 5*.

Categorization of user tasks

Each task scenario and description in the reports was scrutinized and divided into task tokens. Each task token was given a short, unique description. Some task descriptions did not need to be broken down further. For example, Team K, task 10: 'Cancel the reservation you made for the night of May 5th' or Team L, task 6: 'You were going to leave your two children (age 10 and 15) with your brother, but have decided they might enjoy the trip to New York, and want them to come. Can you modify your reservation to have all 4 of you stay in the room?' Both these examples consist of only one task token, i.e. *cancel a reservation* and *add family members to existing booking*.

By contrast, Team N, task 4 was broken down into five task tokens: (1) *fill in a reservation form*; (2) *complete a reservation*; (3) *cancel a reservation*; (4) *navigate to the reservation page*; and (5) *modify a booking*.

During this process, descriptions that were irrelevant to the Hotel Penn web site or which did not comprise a task were removed. For example, Team O, task 2: 'Whom would you travel with?' and Team J, task 6: '...Using the Internet to find a Marriott hotel near the Penn and make a reservation'.

Each task token was printed and glued to a post-it-note and regrouped by affinity diagramming to merge them into user tasks and task goals. For example, Team S's, 'make a reservation for a family of three from June 28 to July 5', and team M's, 'make a reservation for a family of three in June' were merged into one unique user task called, *reserve a family room*. The task goal of this user task was *make a reservation*.

User tasks results

Many of the tasks were irrelevant to the three issues of interest to iHotelier, but as they were relevant to finding usability problems in the OneScreen interface, we included them in our analysis. There were 41 unique user

tasks in total. Of these, 21 (51%) were used by only one team. The number of task scenarios reported by each team and the number of user tasks in our definition are shown in Table 4.

Analysis of problems reported

The CUE-4 evaluation teams were required to report a maximum of 50 problems and any positive comments they had in a specific table format using the same coding system shown in Table 2. Items were numbered sequentially, described in detail, and assigned to a problem category using the definitions given in the instructions. Some teams included a suggested solution to each problem as well. To distinguish between the teams, items were marked by a team code. For example, M-17 uniquely identified the 17th item in Team M’s table.

Category	Name	Description
C	Positive finding	This approach is recommendable and should be preserved.
P	Minor problem	Caused participants to hesitate for a few seconds.
Q	Serious problem	Delayed test participants in their use of the website for 1 to 5 minutes, but eventually they were able to continue. Caused occasional “catastrophes”.
R	Critical problem	Caused frequent catastrophes. A catastrophe is a situation where the website “wins” over the test participants, i.e. a situation where the test participant cannot solve a reasonable task or where the website annoys the test participant considerably.
A	Good idea	A suggestion from a test participant that could lead to a significant improvement for the user experience.
T	Bug	The website works in a way that’s clearly not in accordance with the design specification. This includes spelling errors, dead links, scripting errors, etc.

Table 2. Problem category definitions

Problem categorization

The nine usability test teams reported between 20 and 50 items as shown in Table 3. In total 348 items were submitted.

Team	A	H	J	K	L	M	N	O	S
Items	50	50	20	27	36	50	30	50	35

Table 3. Number of items in the teams’ report.

The first examination revealed many composite items consisting of two or more issues. Our immediate aim was to generate one list of unique usability problems in the categories P, Q, and R. This process involved six steps:

Step 1

Each item was broken down into smaller problem tokens and new ID numbers were assigned. Each token inherited the problem category given by the team in this first round. For example, item A-02 was broken down into three tokens as shown below.

Report item A-02, category Q (serious):

‘The few that figured out that they got AAA discount didn’t like the fact that you could not see what the discount was in absolute \$ or % without setting up a reservation in both circumstances, writing down the prices and comparing them. There was a strong desire to know the discount [sic]’.

This item also included a suggestion: ‘State the discount, also mention it on the main site under rates packages – many people searched there to find an AAA discount, found nothing and assumed no discount applied [sic]’.

This item A-02 was broken down into three problem tokens:

Token A-02-01, category Q: ‘User could not see the promotional and standard rate simultaneously for comparison’.

Token A-02-02, category Q: ‘The few that figured out that they got AAA discount *didn’t like...*’

Token A-02-03, category Q: ‘Many people who searched rates packages to find the AAA discount, found nothing and assumed no discount applied’.

The 348 items became 526 problem tokens. At this stage, it did not matter whether two tokens pointed to the same problem because they went through a grouping process in the later steps and were finally merged.

Step 2: Round 0 grouping

Each problem token was then printed and glued to a post-it-note. Many tokens, although written differently, appeared to address the same problems or issues. In this round 0, we grouped the notes on sheets of flipchart paper on the basis of their sameness. Two tokens were considered to be ‘same’ if the problem was associated with the same widget, in the same location, had the same cause(s), and was experienced by users completing the

same user task. This grouping was carried out by the second author and two students. The two students had been test users in an independent usability test of the Hotel Penn web site designed and conducted as part of another project at Carleton University. All three were familiar with the problems on the web site.

Step 3: Round 1 grouping

Results from Round 0 were transferred from the flipchart to a spreadsheet in January 2005. Each group of tokens was given a Problem ID and a short description to facilitate subsequent analysis. At the end of this round, the list of unique problems/comments comprised 238 items.

Step 4: Round 2 grouping

The Round 1 list was now revised. It contained positive findings, bugs, and good ideas as well as usability problems. A second grouping was then carried out starting with the original set of problem tokens by the second author in July and August, 2006. Round 2 resulted in a list of 200 unique items. The intra-observer test-retest reliability assessed from the two lists (Round 1 in January 2005 and Round 2 in August, 2006) using Cohen's Kappa yielded a value of 0.84, which represents excellent agreement well beyond chance levels between the two rounds [5].

Step 5: Cleaning up

Positive findings (C), good ideas (A), bugs (T), and problems not relevant to the OneScreen reservation system were then removed from the list, leaving 176 problems categorized as minor (P), serious (Q), and critical (R). This step resulted in a final set of unique and relevant problems.

Step 6: Problem severity

UEMs that uncover a high number of severe problems are more valuable than those that uncover a high number of minor problems. For that reason, we filtered the Round 2 list further, by removing minor problems from the list.

The second author went through all problems and their token members in that list and assigned the most appropriate severity rating to them, primarily based on the usability test teams' categories. However, the severity ratings (or categories) assigned by the teams to the tokens of a problem sometimes contradicted each other. For example, problem 161 consisted of 7 tokens including all three levels of severity (P, Q, and R). Severity ratings were therefore assigned systematically as follows:

1. Assign the original value if there is no conflict, e.g. assign P for a set of (P, P).
2. Assign by majority, e.g. P for a set of (P, P, P, Q).

3. Use the average value if applicable. For example, Q would be the average of P, Q, and R.
4. Where conflicts could not be resolved by the above, the definitions of P, Q, and R (see Table 2) and experience in designing and conducting a usability test of the web site were relied upon.

This step resulted in a list of 70 minor problems (40%) and 106 serious-to-critical problems (60%), hereafter referred to as 'severe problems'. The data from the list of severe problems, were then analyzed.

New problems

Cockton & Woolrych's [2] research indicates that additional tasks in a usability test can uncover new problems. We explored this issue further and devised a metric to compare the performance of each usability team with respect to finding new problems.

For each team, we used the data of the rest of the teams as a baseline. At this stage, we had a full set of severe problems found by each test team. This made it possible to calculate the number of unique problems found by any Team X that did not overlap with problems already found by other teams and the total number of unique problems found by the rest of the teams. The percentage of new problems found was then calculated as $100 \times (\text{number of new problems found by Team X}) / (\text{total number of problems found by 8 other teams})$. These results are shown in Table 4.

Team	A	H	J	K	L	M	N	O	S
No users	6	12	7	5	6	15	13	6	6
No tasks (before)	13	9	6	12	10	10	4	16	8
No user tasks	14	11	5	11	12	10	6	10	8
No task tokens	15	13	5	11	12	11	6	11	9
Problems found (%)	42	43	7	22	27	29	23	24	30
% New problems	12	8	0	4	4	3	2	5	4

Table 4. Data from analyses of user tasks and problems found.

RESULTS

The mean percentage of problems and of new problems found were 27.44 (SD = 10.85) and 4.67 (SD = 3.50) respectively. However, although the data for Teams A, H, and J appeared to be outliers, they were included in the statistical analysis because the CUE-4 study had good

external validity: the participating teams were all usability test experts. Thus, with a larger sample of test teams, one would expect more results resembling these.

A Kolmogorov-Smirnov test of normality performed on both the percentage of problems and of new problems was not significant, indicating that the distribution of the data was normal. Pearson Product Moment correlations were then conducted for the variables of interest. Results of these are shown in Table 5.

Variables	% Problems found	% New Problems found
Number of users	no significant correlation	no significant correlation
Number of user tasks	$r = 0.731, p < 0.05$ ($n = 9$)	$r = 0.821, p < 0.01$ ($n = 9$)
Number of task tokens	$r = 0.823, p < 0.01$ ($n = 9$)	$r = 0.870, p < 0.005$ ($n = 9$)

Table 5. Pearson product-moment correlation test results.

Number of users

Table 5 shows that the Pearson product-moment correlations were not significant for the number of users and the percentage of problems found, or for the number of users and the percentage of new problems found. Hence, there was no statistical evidence supporting a relationship between the number of users and either of the two performance variables. This can be seen clearly in the scatter plots in Figures 1 and 2.

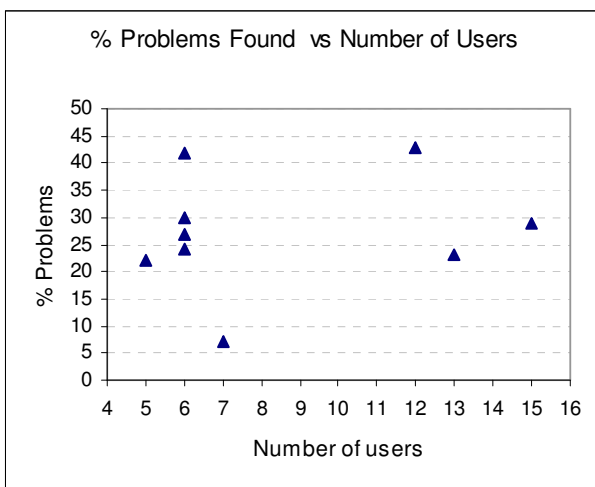


Figure 1. Severe problems found and number of users.

Task coverage

We used the number of user tasks, rather than the number of task tokens, as the primary measure for task coverage in this study for two reasons. Firstly, a high number of

task tokens do not warrant wide task coverage unless the number of user tasks is reasonably high. Secondly, in this study, the number of task tokens of the same test team in a user task was not uniform across all user tasks or across teams. Any conclusion made from an analysis of the CUE-4 data using the number of task tokens alone as a measure of task coverage would therefore not be reliable.

Pearson product-moment correlations were conducted for the number of user tasks and the number of task tokens, and the percentage of problems and of new problems found. The results in Table 5 show that all of these correlations were significant, providing reasonably strong evidence to support a relationship between task coverage and the percentage of severe problems as well as of new severe problems. This can be seen in the scatter plots in Figures 3 and 4. Plots representing the same kinds of data for the number of task tokens yielded similar results.

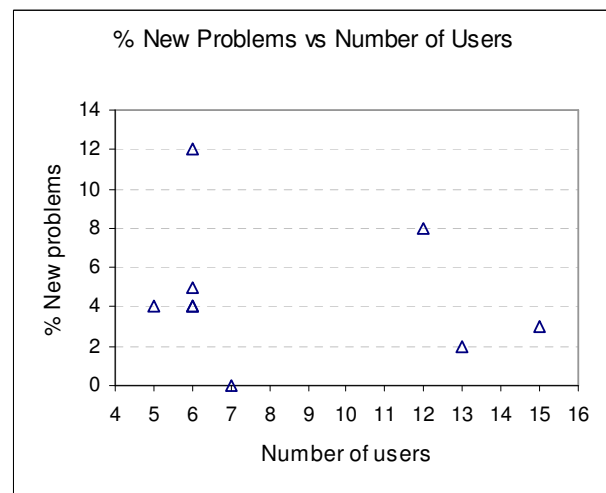


Figure 2. New severe problems found and number of users.

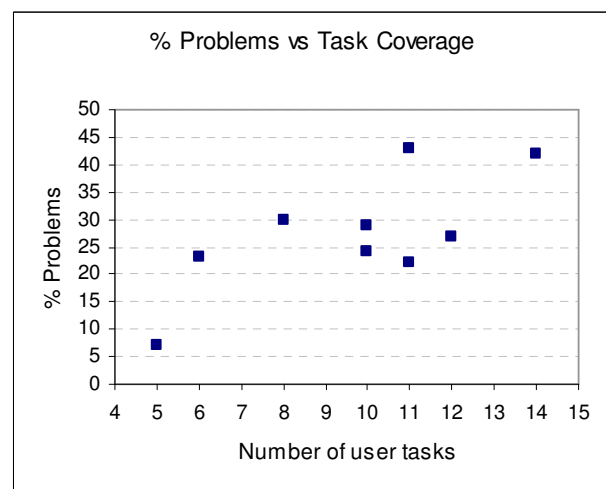


Figure 3. Severe problems found and task coverage.

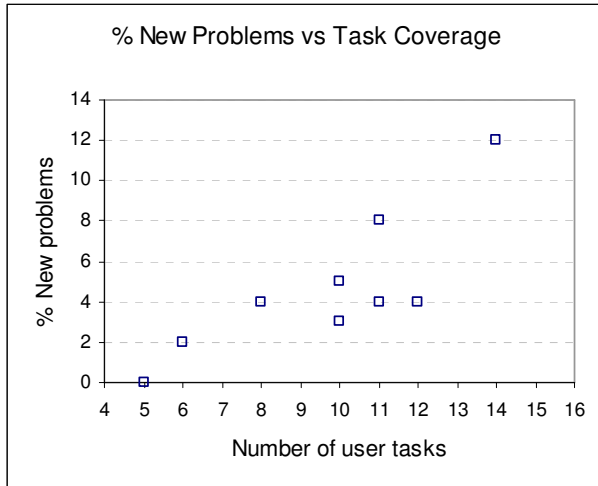


Figure 4. New severe problems found and task coverage.

User participant information

Available information from most of the teams about usability test users included age, gender, internet experience, experience with online shopping, online research and online bookings. There was a good variety in the distribution of age and gender but not in users' experience and familiarity with the Internet, online booking, and online shopping. A heterogeneous sample of test users is more representative of users of the Hotel Penn web site users than a uniform group of test users. The teams' recruitment strategies were examined and rated good, ok, or poor based on this heterogeneity (see Table 6). All three factors (number of users, task coverage, and participant recruitment) were then considered together.

Team L's performance (27%) was much poorer than that of Team A (42%) even though nearly all Team L's tasks were the same as Team A's and both had six test users. This could be due to the difference in their recruitments. According to the criteria applied to assess the 'goodness' of participant recruitment shown in Table 6, Team A's recruitment was superior to that of Team L. Team A recruited a good mix of users while Team L failed to screen out three unsuitable users.

Team J's exceptionally poor performance could be due to the combination of all these three factors: small number of users and tasks and all their users were experts who had used travel web sites before.

The case of Team O is interesting. Its test users traveled extensively, but to the best of our knowledge, half of them had no experience in online bookings. This could explain why O found a slightly higher than average percentage of new problems while its performance was mediocre (24%).

It appears that the role of participant recruitment may be important in that poor participant recruitment seems to

reduce the influence of the number of tasks (e.g. L vs A) on test results while a large sample of user did not seem to improve test results when both participant recruitment and task coverage were good (e.g. A vs H).

Team	Recruitment	Special notes
A	Good	A good mix in online experience
H	OK	A good mix of Internet experience; all but one (a software engineer) were experienced with online booking.
J	Poor	All were experts in online shopping and had used travel sites before. Two users had same attribute values.
K	Poor	All but one were expert Internet users. One was a novice but had experience in online shopping. One was a graphic designer.
L	Poor	A good mix of Internet experience but included a web designer, a usability professional, and an ex-hotel manager.
M	Good	A good mix of Internet and online experience.
N	Poor	All were experts and staff of an IT organization.
O	Unknown	All traveled extensively. Half of the users were inexperienced with online booking. Limited data.
S	Poor	All were experts.

Table 6. Participant recruitment information.

DISCUSSION

The first hypothesis stating that there is a correlation between number of users and proportion of problems found was not supported. Figure 1 does not exhibit any pattern or trend supporting the contention that more problems would be found by increasing the number of users.

All nine usability teams used five or more test users. According to the 5 user claim, 85% of the problems should have been found. Performance should thus not have varied so much between teams. However, the percentage of problems found by the nine teams ranged from 7% to 43% - nowhere near the predicted 85%. The argument is therefore not upheld by the present data.

The 5 user claim is once again questioned. An underlying assumption of the formula on which it is based [14, 16] is that problems are equally likely to be found and that they are found at random. However, this does not appear to happen in practice. Usability tests are preemptive in nature: they are typically guided by a set of design questions defined by developers or clients, and tasks are specifically designed (planned) in advance. Applications vary widely in scope and complexity; users may or may not be representative of the target population for a given application, and usability expertise varies greatly between usability testing teams. It is therefore unlikely that the assumption can be met, and the formula should be used with caution if at all.

One limitation of this work is that there are only nine data points. Then again, the rare richness of this data source does justify the analyses reported here. For now, let us look at the second hypothesis that there is a correlation between the number of tasks executed and the proportion of problems found. It was clearly supported in the above analyses. In addition, the results revealed that the percentage of new problems found by each individual test team did not correlate with the number of users either but it did correlate significantly with the number of task tokens and the number of user tasks. This confirms the important role of task coverage and agrees with some published research discussed earlier.

One interesting finding is that Teams A ($n = 6$ users) and H ($n = 12$ users) performed equally well, uncovering 42% and 43% of the problems from 15 and 13 task tokens respectively. To test whether the 5 user claim may be true in some cases, we identified the overlapping problems found by these two teams. It was only 28%. Thus, over 70% of the problems found by these two teams were found by only one of the teams, which rules out the possibility of the 5-user rule applying to the present data.

It is worth noting that Team J submitted the lowest number of issues (20). It is unclear if, in addition to the issues mentioned earlier, it was unable to devote the necessary time to uncover more problems, or if differences in the teams' skill set may account for its poor performance.

Team M's test design is one of the best. It employed many users and tasks, and a very good mix of test users. Yet its performance was well below that of Teams A and H. By contrast, Team S performed better than expected in light of the poor mix and a small sample of users, and moderate number of tasks. Interestingly, Team S was the only team that gave users a persona to imagine themselves in addition to task scenarios. The persona might have helped their test users place themselves in the real users' shoes and hence carry out the required tasks the way real users would do. As a result, Team S performed better than expected. Indeed, there is evidence

supporting this in a study reporting that participants who pretended to be the target users performed significantly better than those who did not in predicting users' preferences [1]. Taken together, these findings suggest that careful attention to the test design, the characteristics of tasks and users and the number of users and tasks cannot entirely account for the huge variation in the results.

Nevertheless, the statistical results presented in the Results section of this paper provide evidence suggesting that with careful participant recruitment, investing in wide task coverage is more fruitful than increasing the number of users. It pays off to give many sets of user tasks to a small number of users in a usability test rather than giving many users the same limited set of user tasks in a usability test. Apparently, this is true provided care is also taken in the selection of users and tasks. For an optimum ROI, it would thus seem that usability engineers would be wise to strike a balance between the number of user tasks and the number of users when designing a usability test.

FUTURE RESEARCH

Findings from our re-analysis are not surprising but we were surprised to find so little research on how task coverage can potentially improve usability test performance. This paper has raised many interesting questions for future UEM research including the following:

1. What is the extent to which various aspects of user tasks influence performance in a usability test and the number and the types of usability problems identified?
2. What is the role of participant recruitment and its influence on performance?
3. What contributing factors affect usability test performance and how should they be weighted?
4. Are these factors independent of one another?
5. If they are interdependent, what might be the relationships between them?
6. What role do personas play in test design?

It is time that the usability field shifts its focus from sample size to task coverage and other possible contributing factors. We urge UEM researchers to look to other fields such as Information Visualization and Information Retrieval. Research into the taxonomy of user tasks and task selection in the former can provide a good starting point for the first topic suggested above whilst over 50 years of research in the latter can provide a good grounding for evaluation methodology and metrics.

CONCLUSION

We have discussed research into the role of sample size and user tasks in usability testing. We hypothesized that problems found by usability testing would correlate well with these two variables. We gave a detailed description of our re-analysis of the CUE-4 data. The results cast doubt on the role of number of users in a usability test but confirm the important role of task coverage. While this may seem commonsense to researchers and practitioners in the usability field, the point appears to have been under-studied. For more than a decade, the number of users has received much attention whilst the real hero has apparently been overlooked. The paper calls for more research into the role of user tasks on improving usability testing approaches as well as into the importance of recruitment of test users.

ACKNOWLEDGMENTS

We thank Rolf Molich, the CUE-4 organizers and participants for the valuable data, and appreciate the helpful comments of the anonymous reviewers in the preparation of this paper. Part of this research was supported by NSERC/Cognos IRC Grant no: IRCSA 23087-05.

REFERENCES

1. Chattratchart, J. & Jordan, P. W. Simulating 'lived' user experience – Virtual immersion and inclusive design. In *Proceedings of Interact 2003*, Amsterdam: IOS Press (2003), 721-725.
2. Cockton, G & Woolrych, A. Understanding inspection methods: Lessons from an assessment of heuristic evaluation. In *People & Computers XV*, A. Blandford & J. Vanderdonck (Eds.), Springer-Verlag (2001), 171-191.
3. Constantine, L. CHI 2003 Feature: Testing... 1 2 3 4 5 ... Testing... <http://usabilitynews.com/news/article1058.asp>. May, 2003.
4. Faulkner, L. Beyond the five-user assumption: Benefits of increased sample sizes in usability testing. *Behavior Research Methods, Instruments & Computers*, 35, 3, Psychonomic Society (2003), 379-383.
5. Fleiss, J. L. *Statistical Methods for Rates and Proportions*, 2nd Ed., John Wiley & Sons (1981).
6. Hertzum, M. & Jacobsen, N. E. The evaluator effect: A chilling fact about usability evaluation methods. *International Journal of Human-Computer Interaction*, 15, 1, Lawrence Erlbaum Associates (2003), 183-204.
7. Jacobsen, N. E., Hertzum, M., & John, B. E. The evaluator effect in usability studies: Problem detection and severity judgements. In *Proc. HFES 1998*, HFES (1998), 1336-1340.
8. Molich, R. & Dumas, J. S. Comparative Usability Evaluation (CUE-4). *Behaviour & Information Technology*, Taylor & Francis (in press).
9. Molich, R. & Jeffries, R. Comparative expert review, In *Proc. CHI 2003, Extended Abstracts*, ACM Press (2003), 1060-1061.
10. Molich, R., Ede, M. R., Kaasgaard, K., & Karyukin, B. Comparative usability evaluation. *Behaviour & Information Technology*, 23, 1, Taylor & Francis (2004), 65-74.
11. Molich, R., Bevan, N., Curson, I., Butler, S., Kindlund, E., Miller, D., & Kirakowski, J. Comparative evaluation of usability tests. In *Proc. UPA 1998*, UPA (1998), 189-200.
12. Molich, R., Thomsen, A.D., Karyukina, B., Schmidt, L., Ede, M., Oel, W.V. & Arcuri, M. Comparative evaluation of usability tests, *Proc. CHI 1999, Extended Abstracts*, ACM Press (1999), 83-84.
13. Nielsen, J. Why you only need to test with 5 users, *Jakob Nielsen's Alertbox*, March 19, 2000, <http://www.useit.com/alertbox/20000319.html>.
14. Nielsen, J., & Landauer, T. K. A mathematical model of the finding of usability problems. In *Proceedings of INTERCHI 1993*, ACM Press (1993), 206-213.
15. Spool, J. & Schroeder, W. Testing Websites: Five users is nowhere near enough. In *Proc. CHI 2001, Extended Abstracts*, ACM Press (2001), 285-286.
16. Virzi, R.A. Refining the test phase of usability evaluation: How many subjects is enough? *Human Factors*, 34, HFES (1992), 457-468.