The Effect of Delocalized Plans on the Maintainability of Spreadsheets
A Controlled Experiment

Bas Jansen, Felienne Hermans

Report TUD-SERG-2016-014
The Effect of Formula Location on the Maintainability of Spreadsheets

A Controlled Experiment

Bas Jansen
Delft University of Technology
Email: b.jansen@tudelft.nl

Felienne Hermans
Delft University of Technology
Email: f.f.j.hermans@tudelft.nl

Abstract—Spreadsheets are widely used in industry. Companies take important decisions based on information coming from spreadsheets. But, spreadsheets also suffer from typical software maintenance issues. Previous research shows that they contain code smells, lack documentation and testing, have a long live span and during this life span are transferred multiple times from one user to another. To obtain a better understanding of how the maintainability of spreadsheets can be improved, we focus in this paper on spreadsheet comprehension. We analyze the relation between the organization of formulas in a spreadsheet and its maintainability.

To that end, we conduct a controlled experiment with 107 spreadsheet users, divided into two groups. One group receives a model where the formulas are organized in such a way that all related components are grouped closely together, the other group receives a model where the components are spread far and wide in the spreadsheet. All subjects have to perform the same set of comprehension tasks on their spreadsheet.

The results indicate that the way formulas are organized, indeed influences the performance of the subjects in their ability to comprehend and adapt the spreadsheet. Especially for the comprehension tasks, the subjects performed better on the model were the data was grouped closely together. For the adaptation tasks, we found that the way the formulas were constructed influenced the performance of the subjects more than the organization of the formulas itself.

I. INTRODUCTION

The use of spreadsheets in industry is widespread. For millions of employees, spreadsheets form the day-to-day tool to solve business questions, create reports, and deliver support for planning and scheduling activities.

Spreadsheets can be considered a successful end-user programming language. However, it could also be argued that spreadsheets are code [1], as there are similarities with code. Like code, spreadsheets implement concepts like composition, selection and repetition. In spreadsheets, simple objects can be combined into more complex ones by including references to other cells within a cell’s formula (composition), they implement selection with an if-then-else structure and they mimic a loop by the replication of the same formula across many rows or columns.

However, like code, spreadsheets suffer from similar software maintenance issues. They lack documentation [2], contain code smells [3] and clones [4], have an average lifespan of five years and are used or maintained by an average of twelve different users [2], which implicates that during their life span they are transferred multiple times from one user to another.

There is a growing body of research in which methods of software engineering are transferred to spreadsheets. This research include topics about testing [5], reverse engineering [6], [7], code smells, [3], [8], and refactoring [9], [10].

A topic that has not yet been explored in relation to spreadsheets is the concept of delocalized plans or locality and their influence on program comprehension. The existence of a delocalized plan means that the code for a plan is not closely grouped, which makes it more liable for misinterpretation by the programmer. The goal of this paper is to apply this concept on spreadsheets and explore its effect on the user’s ability to comprehend and adjust the spreadsheet.

To address this goal, we set up a controlled experiment with a group of spreadsheet users. We create two different versions of a spreadsheet model. The two models differ in the organization of the formulas within the spreadsheet. The subjects are divided over the two models and asked to perform the same set of comprehension tasks. During the experiment, we measure 1) the correctness, 2) the perceived difficulty, 3) the time to completion, and 4) the number of clicks needed for completion of the task.

The results reveal that indeed, the existence of delocalized plans in spreadsheets influences the user’s ability to comprehend and adjust the spreadsheet. They perform significantly better on the model that contained less delocalized plans.

The contributions of this paper are:

- A definition of the concept of delocalized plans in spreadsheets (Section II)
- Design of a controlled experiment to analyze the ability of subjects to comprehend and adjust a spreadsheet (Section III)
- An empirical evaluation of the effect of delocalized plans in spreadsheets on their maintainability (Section IV)

We organize the remainder of this paper in the following way. In the next section, we provide background information about the concept of delocalized plans in the context of spreadsheets. In Section III we describe the set-up of the experiment followed by a presentation of the results in Section IV and a discussion of the results in Section V. The paper is
concluded with an overview of the related work (Section VI) and the concluding remarks (Section VII).

II. DELocalized PLANs IN SPREADSHEETS

A. Delocalized Plans in Source Code

The concept of locality in source code is a well-researched area. Weinberg [11] defines the concepts of locality and linearity and their effect on program comprehension. **Locality** means that all relevant parts of the program are located closely together. **Linearity** means that decisions in the program are arranged in a strictly linear sequence. The two concepts are related: arranging decisions in a non-linear sequence often causes nonlocality. Locality and linearity can support a programmer with the comprehension and adaption of code.

In their study, Letovsky and Soloway [12] reached a similar conclusion. They explore the relation between program comprehension and delocalized plans. They define a plan as the technique that is used to realize the intention behind the code. A delocalized plan means that the code for the plan is spread far and wide in the source code. In their study, they concluded that delocalized plans are more liable to misinterpretations.

B. Translating the Concept of Delocalized Plans to Spreadsheets

The concept of delocalized plans can easily be translated to spreadsheets. In Figure 1 we have highlighted the formula in cell C32. It calculates the total funding a school receives for its entry level students. The formula in cell C32 is:

\[ = C30 \times C31 \]

Cells C30 and C31 are the direct precedents of this formula and are located close to the formula itself (in the same column in the two rows directly above it). However, the calculation chain does not stop there. Both cells C30 and C31, in turn, also contain a formula and they again refer to other cells. To completely understand the calculation you need to trace back not only the direct but also the indirect precedents. They are illustrated in Figure 1 with the blue arrows. The indirect precedents are located somewhat far away. Nevertheless, they still can be presented in a readable manner on a 13-inch monitor with the zoom factor set to 100%.

In Figure 2 we illustrate the same input, the same calculation, and the same output, but in a spreadsheet where the formulas are organized in a different way. The calculation in this model is located in cell C21 and its formula is:

\[ = 019 / C43 \times C50 \]

If we limit our attention to the direct precedents, we already can observe that they are located much farther apart than in the first example. They are located in different columns (O and C) and also the vertical distance is larger. The precedents are located two rows above, 22 rows below, and 29 rows below the formula. If we also include the indirect precedents, the situation deteriorates. The blue arrows in Figure 2 are directed to all precedents. It is clearly not possible to present them in a readable manner on a single 13-inch screen.

However, in a spreadsheet, there is a third dimension. In the previous two examples, the precedents of a formula were all located on the same worksheet. Nevertheless, it is also possible that precedents are located on a different worksheet. This will require the user to switch between worksheets.

Letovsky and Soloway use the term delocalized plan if the code for a plan is “spread far and wide in the text of the program”. For spreadsheets, we define a delocalized plan as a formula that has its precedents spread widely across the spreadsheet. This could mean that the precedents are located far apart on the same worksheet or that they are located on different worksheets. We define the formula as delocalized if it is impossible to get an overview of all its precedents in a single glance.

Both Letovsky and Soloway and Weinberg argue that source code that tries to avoid delocalized plans is easier to comprehend and adapt. Is this also true for spreadsheets? To answer this question we have designed a controlled experiment. The set-up of this experiment is discussed in the next section.

III. EXPERIMENTAL SETUP

The goal of this paper is to investigate the effect of the existence of delocalized plans in spreadsheets on the user’s ability to comprehend and adjust the spreadsheet model. To address this goal we have formulated the following research...
questions. Does the existence of delocalized plans in spreadsheets influence the user’s ability to:

RQ1 understand a component of a spreadsheet?
RQ2 understand the complete calculation model of a spreadsheet?
RQ3 adapt a spreadsheet?

The distinction between R1 and R2 is the level of abstraction. In the context of source code, we would speak in R1 of code explanation and in R2 of system understanding.

To answer these questions we design a controlled experiment. In the remainder of this section, we explain its set-up.

A. Subjects

To recruit subjects for the experiment we invite the participants of one of our MOOCs1 (Massive Open Online Course), post it on social media and announce it via the mailing lists of EU SPRIG and our own Spreadsheetlab. People who are interested are randomly assigned to one of the two models. In total, we received 107 valid reactions.

We ask the participants to assess their own Excel skills and how frequently they use Excel on a five-point Likert scale. Table I gives an overview of how the participants are distributed over the two models and the average score for their Excel skills and the frequency in which they use Excel.

<table>
<thead>
<tr>
<th>Activity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>Investigating the functionality of (a part of) the system</td>
</tr>
<tr>
<td>A2</td>
<td>Adding to or changing the system’s functionality</td>
</tr>
<tr>
<td>A3</td>
<td>Investigating the internal structure of an artefact</td>
</tr>
<tr>
<td>A4</td>
<td>Investigating dependencies between artefacts</td>
</tr>
<tr>
<td>A5</td>
<td>Investigating runtime interactions in the system</td>
</tr>
<tr>
<td>A6</td>
<td>Investigating how much an artefact is used</td>
</tr>
<tr>
<td>A7</td>
<td>Investigating patterns in the system’s execution</td>
</tr>
<tr>
<td>A8</td>
<td>Assessing the quality of the system’s design</td>
</tr>
<tr>
<td>A9</td>
<td>Understanding the domain of the system</td>
</tr>
</tbody>
</table>

Table III shows that there are no tasks in the experiment that are related to A5 (Investigating runtime interactions in the system) and A7 (Investigating patterns in the system’s execution). Because of the directness of a spreadsheet, there is no clear distinction between coding and runtime. For a user, it is impossible to analyze the behavior of objects during runtime. It is for that reason, that we decide to exclude these two activities from the experiment.

C. The Spreadsheet model

To ensure that all the different comprehension tasks are covered in the experiment we need a spreadsheet of a reasonable size with non-trivial calculations. Furthermore, the experiment should simulate a realistic scenario. For these reasons we choose a spreadsheet that is used in practice by schools to calculate the total funding they receive from the Dutch government, based on the number of students and the number of certificates they issue.

We created two versions of the model, one that adheres to the concept of locality (Model C, precedents are located close to the formula) and one that contains numerous of delocalized plans (Model F, precedents are located far from the formula). Figure 3 is a screen shot of Model C. The blue arrows are directed to the precedents of the formula in the active cell. What we can observe is that the precedents are all located on the same worksheet in the rows above the formula.

In Figure 4 the same formula is displayed, but now in Model F. The precedents are not located on the same worksheet but spread over four different worksheets.

1https://prod-edx-mktg-edit.edx.org/course/data-analysis-take-it-max-delftx-ex101x-0
2Both the questions and the spreadsheet models used in this experiment can be found at: https://figshare.com/articles/The_Effect_of_Formula_Location_on_the_Maintainability_of_Spreadsheets_A_Controlled_Experiment/3167983
TABLE III
COMPREHENSION TASKS USED IN EXPERIMENT AND THEIR MAPPING TO BOTH THE NINE COMPREHENSION ACTIVITIES OF PACIONE AND THE RESEARCH QUESTIONS OF THIS PAPER

<table>
<thead>
<tr>
<th>Question Task</th>
<th>RQ</th>
<th>A1</th>
<th>A2</th>
<th>A3</th>
<th>A4</th>
<th>A5</th>
<th>A6</th>
<th>A7</th>
<th>A8</th>
<th>A9</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1 Explain a calculation</td>
<td>RQ2</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Q2 Find and correct an error</td>
<td>RQ3</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>Q3 Adapt a calculation</td>
<td>RQ3</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>Q4 Explain a key concept of the model</td>
<td>RQ3</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>Q5 Determine relationship between two cells</td>
<td>RQ1</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>Q6 Find dependents of a cell</td>
<td>RQ2</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>Q7 Explain how the spreadsheet can be improved</td>
<td>RQ2</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>Q8 Assess adaptability of the spreadsheet</td>
<td>RQ3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Q9 Assess transferability of the spreadsheet</td>
<td>RQ2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>3</td>
<td>2</td>
<td>5</td>
<td>2</td>
<td>-</td>
<td>-</td>
<td>3</td>
<td>2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. 4. Example of spreadsheet containing delocalized plans

To inspect the formula the user has to switch back and forth between four different worksheets. This in contrast to the first example (Figure 3, Model C), were all precedents are on the same worksheet and because they are located close to the formula itself it is not even necessary to scroll when inspecting the formulas.

D. Experimental procedure

To participate in the experiment the subjects have to register themselves on the web page of the experiment and after registration, they are randomly redirected to a download link for either Model C or Model F. In the instructions we ask them to answer all questions in one sitting. We added a VBA script to the spreadsheet that logs all activities of the subjects. The script is activated when a participant clicks on a random cell in the spreadsheet. From that moment on, each click, activation of a worksheet, or change to a cell is logged with a timestamp. As soon as a subject closes the workbook the associated log file is sent automatically to us by e-mail. Based on the information in the log file we can calculate exactly the time a subject needed to complete a question, the number of clicks that were made, and the number of times the subject had to switch from one worksheet to another.

In the next section, we present the results of our experiment and answer our research questions.

IV. RESULTS

As described in the previous section, we analyzed five different variables: correctness, perceived difficulty, time, clicks, and worksheet changes. During the experiment the subjects had to answer 9 different questions. Table IV gives an overview per question of the results for these variables.

The results are displayed separately for both spreadsheet models. Model C was designed in adherence to the concept of locality, trying to minimize the number of delocalized plans. This in contrast to Model F where the precedents of formulas were spread far and wide in the spreadsheet, leading to a high number of delocalized plans (see for examples Figure 3 and 4 in Section III).

A Shapiro-Wilk test showed that the data did not follow a normal distribution for almost all combinations of questions and models. Therefore, we used the Wilcoxon-Mann-Whitney
test to determine per question if there was a significant difference between the two models. Table V shows that for some questions there is a significant difference between the two models.

For each variable with a significant difference we calculated the effect with the Cliff’s Delta (Table VI). For most variables the measured effect was medium ($0.33 \leq d < 0.47$) [16]. Exceptions are the number of clicks for Question 6 where the effect is large ($d \geq 0.47$) and the correctness, where the effect for Q3, Q4, and Q6 is small ($0.15 \leq d < 0.33$).

In the remainder of this section we will interpret the results and answer or research questions.

A. RQ1: Does the existence of delocalized plans in spreadsheets influence the user’s ability to understand a component of the spreadsheet?

Pacione considers several levels of abstraction with respect to code comprehension. We see this reflected in the set of comprehension activities (Table II). In our experiment the questions Q5 and Q6 are related to code comprehension on the component level (see Table III).

In Question 5 we ask the subjects to determine if one cell is a precedent of another cell. For this question there is no significant difference in performance between the two groups. This in contrast to Question six where there is a significant difference. In this question we ask participants to indicate which cells in the model will change if they change a particular cell that contains the ‘budget factor student value’ (see Figure 5). To be able to answer this questions they have to trace the dependents of this cell. The participants in Model C answered this question significantly faster and needed less clicks. And although there is no significant difference in the correctness of the answer if we use a confidence level of 95%, it is significant if we lower the confidence level to 90%. Notable is the high number of clicks that participants need to answer this question in Model F. In addition of these clicks they also needed to change frequently between worksheets.

For both questions the subjects have to determine relations between cells in the spreadsheet. However in Question 5 we ask the subject if there is a dependency between two cells. To answer this question the subject only has to analyze a part of the calculation chain (namely the branch that connects the two cells). Question 6 is different. In this question we ask the subject to highlight all cells that are depending on another cell. For this question there is no significant difference. In this question significantly faster and needed less clicks. And although there is no significant difference in the correctness of the answer if we use a confidence level of 95%, it is significant if we lower the confidence level to 90%. Notable is the high number of clicks that participants need to answer this question in Model F. In addition of these clicks they also needed to change frequently between worksheets.

In the remainder of this section we will interpret the results and answer or research questions.

A. RQ1: Does the existence of delocalized plans in spreadsheets influence the user’s ability to understand a component of the spreadsheet?

Pacione considers several levels of abstraction with respect to code comprehension. We see this reflected in the set of comprehension activities (Table II). In our experiment the questions Q5 and Q6 are related to code comprehension on the component level (see Table III).

### TABLE IV
RESULTS FOR DEPENDENT VARIABLES CORRECTNESS, DIFFICULTY, TIME, # CLICKS, # SHEET CHANGES FOR BOTH MODEL C AND MODEL F

<table>
<thead>
<tr>
<th>Question</th>
<th>Task</th>
<th>Correctness</th>
<th>Difficulty</th>
<th>Time</th>
<th># Clicks</th>
<th># Sheet Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>Explain a calculation</td>
<td>6.2</td>
<td>3.9</td>
<td>3.2</td>
<td>5.26</td>
<td>7.55</td>
</tr>
<tr>
<td>Q2</td>
<td>Find and correct an error</td>
<td>7.9</td>
<td>8.5</td>
<td>3.7</td>
<td>5.27</td>
<td>5.21</td>
</tr>
<tr>
<td>Q3</td>
<td>Adapt a calculation</td>
<td>5.4</td>
<td>6.8</td>
<td>3.5</td>
<td>3.6</td>
<td>6.64</td>
</tr>
<tr>
<td>Q4</td>
<td>Explain a key concept of the model</td>
<td>5.6</td>
<td>4.1</td>
<td>3.5</td>
<td>3.2</td>
<td>4.23</td>
</tr>
<tr>
<td>Q5</td>
<td>Determine relationship between two cells</td>
<td>8.6</td>
<td>8.9</td>
<td>3.4</td>
<td>3.4</td>
<td>6.42</td>
</tr>
<tr>
<td>Q6</td>
<td>Find dependents of a cell</td>
<td>6.8</td>
<td>5.9</td>
<td>3.5</td>
<td>3.6</td>
<td>3.51</td>
</tr>
<tr>
<td>Q7</td>
<td>Explain how the spreadsheet can be improved</td>
<td>-</td>
<td>-</td>
<td>3.1</td>
<td>3.3</td>
<td>2.28</td>
</tr>
<tr>
<td>Q8</td>
<td>Assess adaptability of the spreadsheet†</td>
<td>6.5</td>
<td>6.2</td>
<td>-</td>
<td>-</td>
<td>1:14</td>
</tr>
<tr>
<td>Q9</td>
<td>Assess transferability of the spreadsheet†</td>
<td>5.3</td>
<td>5.2</td>
<td>-</td>
<td>-</td>
<td>0.50</td>
</tr>
</tbody>
</table>

† In Q8 and Q9 the subjects were asked to assess the adaptability and transferability on a scale from 1 to 10, 1 meaning very difficult, 10 very easy.

### TABLE V
P-VALUES (WILCOXON-MANN-WHITNEY) PER QUESTION

<table>
<thead>
<tr>
<th>Question</th>
<th>Correctness</th>
<th>Difficulty</th>
<th>Time</th>
<th># Clicks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>&lt;0.05</td>
<td>&lt;0.05</td>
<td>&lt;0.05</td>
<td></td>
</tr>
<tr>
<td>Q2</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Q3</td>
<td>&lt;0.05</td>
<td>-</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Q4</td>
<td>&lt;0.05</td>
<td>-</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Q5</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Q6</td>
<td>&lt;0.10</td>
<td>&lt;0.05</td>
<td>&lt;0.05</td>
<td></td>
</tr>
<tr>
<td>Q7</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Q8</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Q9</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td></td>
</tr>
</tbody>
</table>

### TABLE VI
CLIFF’S DELTA PER QUESTION

<table>
<thead>
<tr>
<th>Question</th>
<th>Correctness</th>
<th>Difficulty</th>
<th>Time</th>
<th># Clicks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>-</td>
<td>0.4044</td>
<td>0.4194</td>
<td>0.3000</td>
</tr>
<tr>
<td>Q3</td>
<td>0.2108</td>
<td>-</td>
<td>-</td>
<td>0.3955</td>
</tr>
<tr>
<td>Q4</td>
<td>0.2186</td>
<td>-</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Q6</td>
<td>0.1835</td>
<td>-</td>
<td>0.3865</td>
<td>0.5100</td>
</tr>
</tbody>
</table>

Fig. 5. Budget factor student value in both Model C (left) and Model F (right)
comprehend the relations between components of the spreadsheet. If there are more delocalized plans, the user needs more time and is less accurate.

Users perform better on formula comprehension when the precedents of formula are located close to the formula itself.

B. RQ2: Does the existence of delocalized plans in spreadsheets influence the user’s ability to understand the complete calculation model of the spreadsheet?

In Question 1 and Question 4 it is tested to which degree the subject understands the workings of the model. As stated in the previous section, the model calculates the total funding a school receives given the number of students. The main component for the funding is the student value. It is depending from the number of students but has a different value. In Question 1 we ask the subject to explain why the student value is higher than the total number of students. To answer this question they have to analyze a set of formulas and the relations between them.

For both models the correctness of the answer did not show any difference. However, on average it took the participants in Model F more time and they needed more clicks than the participants in Model C. Also the Model F participants thought that the question was more difficult. In Model C all cells used to calculate the student value could be inspected on a single worksheet without the need for scrolling. In Model F the participants had to switch between two worksheets. It is plausible that the switching between sheets and the fact that all precedents of the formula could not be inspected on a single screen is responsible for the more time that was needed, the higher number of clicks, and a higher level of perceived difficulty.

In Question 4 we also asked the participants to explain the working of the model. An important concept in the model is the cascade factor. It is a weighting factor that ensures that the school gets less funding for students that need more time to finish their education. We ask the participants to explain the cascade factor in their own words. Although there is no significant difference in the time and number of clicks needed to answer this question, the quality of the answers in Model C is significantly higher (See Table V).

Seeing all information on one screen aids the user to better understand the meaning of the different components and to see the relations between them. It supports users in reasoning about the functioning of the model.

In both Question 7 and 9 the subjects are asked to assess the quality of the spreadsheet. In Question 7 they have to suggest improvements to the model, whereas in Question 9 they are asked to assess how easy it is to explain the working of the spreadsheet to somebody else. Both questions are related to the user’s ability to understand the spreadsheet.

It is notable that there is no significant difference, for both questions, in the subject’s assessment of the quality of the spreadsheet, whereas we do observe a difference in the performance when subjects have to execute a comprehension task in Q1 and Q4.

We also received contradicting advice about how to improve the spreadsheet. A subject working with Model C (that consisted of a single worksheet) suggested to: “split the long sheet to several sheets and name them by what they calculate.”, while a subject working with Model F (consisting of multiple worksheets) wrote that: “having all the inputs on one sheet would be helpful (to prevent having to scroll trough tabs all the time)”.

In both groups there were several subjects that suggested that the spreadsheet would be easier to comprehend if named ranges were used (replacing cell addresses by meaningful names). In future work we will research the effect of using named ranges on formula comprehension.

C. RQ3 Does the existence of delocalized plans in spreadsheets influence the user’s ability to adapt the spreadsheet?

In both Question 2 and 3 the subjects are asked to make adaptations to the model. In Question 2 they have to find and correct an error. For this question we do not see any significant difference between the two groups. The subjects knew on beforehand which part of the spreadsheet contained the error. To find the error it was not necessary to trace all the precedents, inspecting the formula was sufficient. This explains why the organization of the formulas in the spreadsheet did not influence the subjects ability to correct the error.

In Question 3 we asked the participants to change the model. Each school receives additional funding for disabled students. In our model it takes into account both the number of students at entry level as the other students. In the question we asked to exclude the entry level students from the calculation. In this case the results indicate that it was easier to make this change in Model F than in Model C. For model F, the participants needed less clicks and submitted a higher number of correct answers. In Model C all components of the formula were located on a single sheet. In Model F the components were split between two worksheets. In order to make the change the subjects needed to change sheets several times. Nevertheless the performance in Model F was better.

There are two possible explanations. Although in Model F, the components of the formula were spread over two different sheets, on both sheets all information was presented on a single (13 inch) screen. Whereas in Model C all information was located on one sheet, but not visible in one glance. We suspect that subjects had to scroll to see the complete picture. So one explanation could be that the ability to see all information in one glance is responsible for the better performance.

Another explanation could be the formula itself. In order to make the change in Model C the subjects needed to make changes in more than one cell, whereas in Model F it was sufficient to make a change in only one cell.
We designed a small second experiment to determine which of these two explanations is more plausible. We provided fourteen employees of the financial staff of Delft University of Technology with an advanced Excel training. During this training we let them do an exercise that was designed to determine which of the aforementioned explanations is the most plausible.

In this experiment for both models it was not possible to see all information in one glance.Scrolling was necessary in both cases. The only remaining difference was the way the formula was constructed. The fourteen employees were randomly assigned two either Model C or Model F. The results for this experiment are summarized in Table VII.

<table>
<thead>
<tr>
<th>Activity</th>
<th>Model C</th>
<th>Model F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Correctness</td>
<td>8.1</td>
<td>10</td>
</tr>
<tr>
<td>Difficulty</td>
<td>3.2</td>
<td>3.4</td>
</tr>
<tr>
<td>Time</td>
<td>8:15</td>
<td>4:40</td>
</tr>
<tr>
<td>Clicks</td>
<td>40</td>
<td>9</td>
</tr>
</tbody>
</table>

Again the performance was better in Model F. All eight subjects that were assigned to Model F completed the exercise flawlessly. In this model it was sufficient to change just a single cell, making the way the formula was constructed the most plausible explanation for the better performance.

When adapting a formula, the way the formula is constructed is more important than the location of its precedents.

This finding also sheds light on a previously unresolved issue regarding the trade-off between the long method and the long calculation chain smell in spreadsheet formulas as mentioned in [3] and [17]. Defining a short formula to avoid the long method smell inevitable leads to a longer calculation chain. Trying to reduce the long calculation chain smell will lead to longer formulas. Based on the findings in this paper it appears that reducing the length of the calculation chain has more impact on maintainability of a formula than the length of the formula itself.

A long calculation chain makes it more difficult to adapt a formula than the length of the formula itself.

In Question 8, subjects are asked to assess how easy it would be to adapt the spreadsheet. The results for this question do not show a significant difference between the two models. This is in line with our earlier finding that the length of the calculation chain has more impact on maintainability of a formula than the location of its precedents.

V. DISCUSSION

A. Threats to validity

There are several threats to the internal validity of this study. The first threat is the subject’s skill level with respect to Excel. To mitigate this risk we asked the subjects to rate how frequently they use Excel and how they assess their skill level on a five-point Likert scale. Table I shows that for both variables there is no significant difference between the two groups.

The second threat is that the subjects were aware of the goals of the Study. To mitigate this we referred to the goals of the study in very general terms in both the invitation to participate and the instruction (for example: ‘We are very much interested in how people are interacting with spreadsheets’). Furthermore, the log file we used to analyze the subject’s interaction with the spreadsheet was sent to us without showing it to the subject.

A third threat to the internal validity is the threat of self-selection. We invited participants of a MOOC about spreadsheets, approached mailing list members of a spreadsheet interest group (EUSPRIG), and members of our own Spreadsheet Lab mailing list. Because of this, it is plausible that all subjects have a more than average interest in the subject of spreadsheets and are not representative of the total population of spreadsheet users. Nevertheless, we decided to approach these possible subjects to maximize the set of participants. And although these subjects have possibly a more than average interest in spreadsheets it is also more likely that they use spreadsheets in their daily activities.

A threat to the external validity of the experiment could be the representativeness of the spreadsheet model that we used in the study. We mitigated this risk by using a spreadsheet model that is based on a model that is used in practice by professionals.

Also, the representativeness of the subjects is an external threat. This risk is reduced by the relatively large sample size of 107 subjects. They varied in their age, cultural background, and Excel skill level and were randomly assigned to one of the two models.

B. Pivot Tables, Charts and VBA code

In this study, we primarily focused on the organization of formulas in a spreadsheet. The concept of locality, however, could also be applied on other constructions in spreadsheets like pivot tables, charts, and VBA code. In future work, we will research the effect of these concepts on the user’s ability to comprehend and maintain the spreadsheet.

C. Long Calculation Chains versus Long Formulas

In previous research on code smells in spreadsheet formulas it is stated that there is a trade-off between the Long Calculation Chain smell and the Multiple Operations and Multiple References smells [3], [17]. Our results indicate that the Long Calculation Chain smell has a higher impact on the user’s ability to comprehend and maintain a formula than the Multiple Operations and Multiple References smells. It is
easier for a user to comprehend and adapt a long formula with a short calculation chain than a short formula with a long calculation chain. This finding contradicts the popular belief that short formulas are easier to comprehend. For example, the FAST Standard Organisation advises in their popular FAST standard to write short formulas: “A formula longer than your thumb likely means that it should be broken into more than one step.” [18].

VI. RELATED WORK

A. Software Engineering Methods and Spreadsheets

There are numerous studies that focus on applying software engineering methods on spreadsheets. Rothermel et al. brought the concept of testing to spreadsheets with the What You See is What You Test paradigm [5]. Other researchers focused on the domain of reverse engineering and came up with methods to extract class diagrams from spreadsheets [6] [7] or to visualize the data flow within spreadsheets [19]. Fowler introduced the concept of smells in code [20], but smells also exist in spreadsheets. They are described in detail in the work of Hermans [21] [22], Cunha [8] and Barowy et al. [23]. From smells it is a small and logical step to refactoring. Hermans defined refactorings for formula smells in spreadsheets [9]. Inspired by this work Badame and Dig developed RefBook, a tool that supports a number of refactorings for spreadsheet formulas [10].

B. Delocalized Plans and Program Comprehension

Weinberg defines several principles for programming language design [11]. According to the author, a properly designed language aids the programmer by keeping relevant information close at hand. In this context, he defines the concepts of locality and linearity. Locality means that all relevant parts of a program are found in the same place, for example on the same page or on a single screen. Linearity means that the decisions in the program are arranged in a strictly linear sequence.

Letovsky and Soloway [12] studied the relation between locality and program comprehension. They state that the goal of program understanding is to recover the intentions behind the code. They define a plan as the technique that is used to realize an intention. In their paper they focus on so-called delocalized plans, meaning that the code for the plan is not closely grouped, but spread far and wide in the source code. They found that in order to understand a program, programmers make reasonable but sometimes incorrect assumptions. They tend to leave these assumptions unverified if the effort required to verify the assumption is great. Typically in delocalized plans, it will take more effort to verify the assumptions. According to the authors, these plans are more liable to misinterpretation than plans whose code is closely grouped. Delocalized plans could also be defined as plans with data flow links spanning widely separate parts of the code. Providing the developer with a data flow analyzer that makes these links explicit could reduce the risk of misinterpretations.

C. Controlled Experiments in Software Engineering

Other research discusses the set-up of experiments to analyze software comprehension. Pacione et al. [13] introduce a software visualization model for supporting object-oriented software comprehension. They evaluate the performance of visualization tools by assessing their performance in typical software comprehension tasks. Based on previous work and evaluation tasks found in software comprehension and software visualization literature, they introduced a framework of nine principal comprehension activities (see also Table II in Section III). In the current paper, we use this framework to make sure that all relevant program comprehension activities were covered in our experimental setup.

A similar set-up of a controlled experiment in software engineering was used by Cornelissen et al. [14]. The authors use a controlled experiment to evaluate the effectiveness of a tool for the visualization of large traces (Extraviz) and also apply Pacione’s comprehension framework to select their comprehension tasks for the experiment.

VII. CONCLUDING REMARKS

The goal of this paper is to investigate the concept of delocalized plans in spreadsheets and its effect on the user’s ability to comprehend and adjust the spreadsheet.

To address this goal, we set up a controlled experiment using two spreadsheets that differ in the organization of the formulas within the spreadsheet. In the experiment, the subjects are divided over the two models and perform a number of comprehension tasks, of which we measure correctness, perceived difficulty, completion time and the total number of clicks.

The results reveal that the existence of delocalized plans in spreadsheets influences the user’s ability to comprehend and adjust the spreadsheet: subjects perform significantly better on the model that contained less delocalized plans.

The contribution of this paper are:

- A definition of the concept of delocalized plans in spreadsheets (Section II)
- Design of a controlled experiment to analyze the ability of subjects to comprehend and adjust a spreadsheet (Section III)
- An empirical evaluation of the effect of delocalized plans in spreadsheets on their maintainability (Section IV)

A. Future work

This paper triggers several angles for follow-up research. Firstly a more extensive experiment is needed to understand the impact of delocalized plans in spreadsheets in more depth. For example, with a larger controlled experiment using multiple spreadsheets, followed by a longitudinal study.

Furthermore, we envision a tool that measures the occurrence of delocalized plans in spreadsheets. Such a tool could support users in estimating the effort needed for maintaining a spreadsheet.

In this experiment references in formulas were presented in ‘A1’ style. However, there are alternatives like ‘R1C1’ notation.
or the use of named ranges for references. In a future study, we will investigate the effect of these different forms of notation on formula comprehension.

REFERENCES


