Simulating Collaboration Processes to Understand and Predict Group Performance

Gwendolyn Kolfschoten
Delft University of Technology
Faculty of Technology, Policy, and Management
g.l.kolfschoten@tudelft.nl

Stephan Lukosch
Delft University of Technology
Faculty of Technology, Policy, and Management
s.g.lukosch@tudelft.nl

Mamadou Seck
Delft University of Technology
Faculty of Technology, Policy, and Management
m.d.seck@tudelft.nl

Abstract

Collaboration and collaboration support are often a mixed blessing. While performing tasks collaboratively is useful to gain commitment, shared understanding and to benefit from the knowledge and different perspectives of all participants, it is difficult for groups to work effectively and efficiently. Furthermore, it is challenging to design collaboration support tools and techniques, as not only the requirements of users interacting with the computer, but also the interaction processes among the users and computer interactions need to be considered. In order to support facilitators in monitoring collaboration processes and to suggest process interventions and tool adaptations, we present a first simulation model of a collaboration process. In this model we simulate the impact of group size. Further, we sketch the research agenda to model and simulate collaboration processes. We aim to use these models to compare and evaluate collaboration processes as well as to predict the outcome of possible process interventions and corresponding tool adaptations.

1. Introduction

Groups that need to collaborate can benefit from collaboration support [1, 2]. Collaboration support can be comprised of tools, processes and services that support groups in their joint effort. In knowledge oriented organizations, there is often a need or demand for collaboration support. Tools and technology for group support exist in a variety of shapes, from complex computer systems (groupware) and group support systems, to simple boxes with cards and pencils. Besides tools and technology, there are several other factors that determine the success of collaboration support. For example, group size has been studied and appears to be an important factor in the success of collaboration. Researchers seem to agree that larger groups benefit more from collaboration support than small groups [3, 4].

Causes are found in the increased amount of conflict [14] and the “span of control” of a (natural) leader. Another key factor is the facilitator [5]. A facilitator of a collaboration process takes the role of process leader, offering the group guidance in their choice of collaborative activities, and instructing and guiding them in the use of collaboration support techniques and tools. The methods a facilitator chooses, but also the style and skillfulness with which an instruction is conveyed has effect on outcomes [6, 7]. The impact of these nuances makes it fairly difficult to predict and reproduce the effects of collaboration support. Large studies are performed to test relatively small interventions [8], and findings in lab settings are not always consistent with findings in field settings [4, 9]. Therefore, designing and supporting collaboration processes is a complex task, which requires expertise from facilitators [10].

The role of a facilitator in collaboration support is described extensively in literature [2, 11, 12]. A facilitator supports a collaboration process with tools and techniques, thus intervening in a collaboration process in order to improve its effectiveness and efficiency. We call the actions of a facilitator to support the group, offering tools and techniques, and guiding the group in using these, “facilitation interventions”.

Facilitation interventions are made at three different levels of abstraction; interventions at the level of the meeting agenda, at the level of activities, and at a micro level encompassing interventions to structure and adapt collaborative behavior [13]. At the agenda level, the facilitator chooses a sequence of activities to support the group in accomplishing their goal and makes interventions to move the group from activity to activity. At the activity level, the task of the facilitator is to give instructions on how to collaborate and on how to use the tools and techniques for each of the activities. These interventions can be captured in thinkLets. A thinkLet is a script for the facilitation of a single collaborative activity [14]. At the micro level, a facilitator makes interventions to motivate the group,
to guard and improve quality, to guard behavioral rules and to keep a productive engaging atmosphere. For this micro level of intervention, we cannot prescribe a detailed set of instructions, as these depend on the characteristics of the group (heterogeneity, history, culture, age, size, etc.) and the position of the facilitator with respect to the group (e.g., part of the team, or paid external facilitator). We can only set "ground rules" that a facilitator can guard and use as a guidance for intervention. In the research on collaboration engineering, a next step is made to formalize the description of facilitation interventions in thinkLets [15, 16]. ThinkLets give the group instructions on how to perform a certain activity. Using thinkLets, we can make facilitation interventions reusable, and reproducible, even by novices [17].

However, while thinkLets give us precise prescriptions of interventions and their expected effects, small modifications of thinkLets can have significant impact on their outcome [6, 7] and would thus require the study and evaluation of an endless set of combinations and modifications of techniques and methods. Furthermore, facilitators are often reluctant to try new facilitation techniques [18], as they cannot predict their effects. This makes it difficult to advance the effectiveness of collaboration support tools.

New collaboration support techniques as well as facilitation interventions often require adapting the collaboration support environment. As a result, such a support environment needs to be open, extensible and adaptive in order to offer the necessary support. First steps towards context-adaptive and extensible collaboration support environment allow users to specify adaptation rules that change the support environment in case a specific situation occurs, e.g., by introducing new categories to cluster ideas or allowing to rate or comment ideas. Specifying such adaptation rules requires understanding the underlying collaboration process and the capability to assess the impact of such an adaptation. Designing such support environment or specifying adaptation rules is therefore challenging [19]. In line with this, the increasing demand of collaboration support tools [20], and the increased understanding that their fit [21], and customization even to the task they support is critical [22, 23], a need emerges to increase our ability to predict the effect of interventions in collaboration.

To address the above issues, we present in this paper an approach to model and simulate collaboration processes and assess the impact of process interventions as well as accompanying tool adaptations. Once collaboration processes can be modeled and simulated at runtime, the effect of alternative tool and process changes can be compared based on process-specific quality criteria and the most promising interventions can be suggested to a facilitator or automatically executed by the support environment. While in the first case, the facilitator is supported in his core tasks, the latter case would represent the first steps towards the facilitator-in-a-box [19].

In the remainder of this paper, we will first further explain the thinkLet concept as a way to formalize the description of facilitation interventions. Next we will move from the thinkLet notation of a specific brainstorming technique to a detailed model of the steps in the intervention, and in the end to an even more detailed overview of the cognitive steps of the activity. At this level we created a computer model to simulate the collaborative activity. We used data from two large studies in the literature to feed our model and to validate it. We will discuss the approach, the future research required and the use and implications of the simulation model for research and practice.

2. ThinkLets

In an effort to reduce the need for professional facilitators, researchers have been coding facilitation practices to enable the separation of the design task of a facilitator and the execution task [14]. In this approach a master facilitator called collaboration engineer, can design and transfer a collaborative work practice to practitioners (domain experts in an organization) to execute a specific domain-related collaboration process on their own, based on a short training. This approach is called collaboration engineering [14]. To ensure predictability and transferability, the collaborative work practices are designed with design patterns called thinkLets [15, 16].

Though practitioners can facilitate collaboration processes based on thinkLets, collaboration processes often still need to be adapted at execution time. Professional facilitators predict the effect such process interventions on extensive facilitation experience and a number of observations of the group interaction. To enable practitioners to also change the collaboration process at execution time, thinkLets are described with background information to explain their effects. However, at this point, there is a limit to the predictability of the thinkLet intervention. We will explain the concepts that provide the foundations
ThinkLets and other facilitation techniques are, in essence, a structured form of communication. A ThinkLet consists of a message that is transmitted with use of a channel/vehicle from a sender (facilitator) who can encode the message and send it to a receiver (participant/group) who can decode this message. From this decoding effort, a meaning can be distilled and, in response, an action can be performed. A response can be either in terms of feedback, i.e. the transmission of a return message, or a communication effect, i.e. any other action in response to the message which will constitute a pattern of collaboration, and a specific result [19].

To realize an intention with facilitation, two types of interventions are required [19]. First, there are static interventions in which one or more commands are given to initiate the key activities of a process. We will refer to this kind of communication as an instruction intervention. Second, there are dynamic interventions intended to adjust the actions performed by the group to resolve a discrepancy between the facilitator’s intentions and the groups’ actions. These interventions depend on emergent conditions. We will call these messages adjustment interventions.

The conceptual design of a thinkLet consists of a set of instruction and adjustment interventions described as rules. These rules are similar to rules mimicking human behavior in avatars [15, 16, 24]. Each rule describes for a role an action that needs to be performed using a capability under a set of constraints to restrict those actions. Further, some thinkLets include conditional rules for frequently-required adjustment interventions because specific discrepancies manifest predictably during the execution of an activity based on the thinkLet.

ThinkLets thus have as their basis a sequence of instructions and interventions that jointly structure the individual behavior of group members to create results that benefit from synergy and productive group behavior. Besides these instructions, thinkLets capture all kinds of information to enable a practitioner to execute the collaborative activity. This includes a description of the effect of the thinkLet on the pattern of collaboration that occurs in the group and an overview of challenges that can occur during the activity and ways to overcome these challenges. In this way, thinkLets are captured as design patterns [15, 16].

ThinkLets are generic in the sense that they can be applied in different types of tasks, and with different tools to fulfill the capabilities. Further, the thinkLet pattern language also consists of a number of modifiers. Modifiers are small variations to the basic pattern of collaboration that a thinkLet produces.

While the design pattern offers a useful instruction for practitioners, it is insufficient to model and simulate the activity. Knoll et al. [25] made a first step in modeling a thinkLet activity with this level of detail at the instruction level. However, besides detailing the instruction we need to understand the behavioral mechanisms that create the synergy as a result of these instructions. An example is the thinkLet FREEBRAINSTORM [26, 27].

The core rules of the FREEBRAINSTORM are listed in the following description:

<table>
<thead>
<tr>
<th>ThinkLet: FREEBRAINSTORM</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Description:</strong> Participants brainstorm in parallel on a single or multiple sub-topics. After each contribution, participants randomly swap contribution sheets. Participants can only see the contributions on the page they are working on, after swapping they are able to see the contributions on the next page that is received.</td>
</tr>
<tr>
<td>1. Allow participants to add only one contribution to each page they receive.</td>
</tr>
<tr>
<td>2. Allow participants to add contributions in parallel, anonymously.</td>
</tr>
<tr>
<td>3. Ensure that participants swap pages after they contributed.</td>
</tr>
<tr>
<td>4. Allow participants to add contributions that match to the contribution specification or comments on contributions of others.</td>
</tr>
<tr>
<td>5. Ensure that participants read the contributions of others for inspiration.</td>
</tr>
</tbody>
</table>

From experience we know that the FREEBRAINSTORM thinkLet creates synergy, shared understanding, reduces barriers to participate, stimulates creativity, etc. However, to understand the precise effect of the interventions made in FREEBRAINSTORM we need to model it in more detail.

3. Collaboration process model

Figure 1 shows a process model of the FREEBRAINSTORM thinkLet. The model indicates the steps of the process. First the facilitator instructs the brainstorming question and the way the tool is to be used. Depending on the technology used, these instructions can be embedded in the tool (e.g in
Group Systems thinktank, instructions can be listed), and the use of the tool can be more or less intuitive, which has an impact on the cognitive effort required to understand and execute the activity. The initial cognitive effort of the brainstorming task further depends on the properties of the task, such as size, complexity and required quality, and on properties of the participants such as their motivation, ability and creativity. Finally there is an impact of the group composition, the size of the group, and for instance their hierarchical distance and the initial shared understanding. Next, the facilitator or the system assigns each participant a page. The participants now processed the task, and start brainstorming. The process further requires that they swap pages and that they read the ideas of others. Both reading and creating new ideas requires cognitive effort. Further there are several cognitive effects of brainstorming described in literature. These include attenuation blocking and evaluation apprehension (considering the relevance of the idea) [3], which is partly solved by anonymity. The effect of anonymity is that people are less hesitant to share ideas. However, this implies that people do consider the impact of their ideas to some extent, which will require cognitive effort. Next, people are confronted by the ideas of others. Stimuli, such as the ideas of others [28, 29] can spark new ideas, but can also cause cognitive interference or spreading, when they interrupt, or lead to distracting cognitive association. Finally, at some point people run dry of ideas, and get a feeling that all ideas are shared. We call this saturation, again this implies that people do some evaluation of the completeness or quality of the set of ideas. When we now list the cognitive activities from a participant perspective we get an even longer list of things that can be modeled to simulate the cognitive process of brainstorming:

1. understanding the problem/question
2. understanding the tool used to capture ideas
3. focus on the task, and start thinking of solutions
4. conceiving and idea
5. explicating the idea (depending on tools used)
6. considering the implications of sharing an idea
7. reading the idea(s) of others
8. conceive new idea (inspired)
9. reflecting on performance with respect to target
10. being distracted

Thus, we moved from 5 rules of FREEBRAINSTORM to 10 cognitive activities. There are probably more effects that we did not yet find in literature. However, with this understanding of the thinkLet we will first attempt to simulate the FREEBRAINSTORM ThinkLet.
4. Simulating collaboration processes

As we saw from the previous discussions, collaboration is an interactive process where complex individual and group dynamics occur. In this section, we describe the simulation model of a collaborative process supported by the FREEBRAINSTORM ThinkLet.

In previous work, discrete event modeling and simulation [30] was shown to be adequate for representing complex human and social behavior. Such models were obtained through the operationalization of psychology and social science theories. In this work, we build a first simulation model of the FREEBRAINSTORM thinkLet. The model is built using the ARENA simulation package. In further research, the DEVS formalism will be used, for a more detailed model specification.

The ARENA model is developed following a process-oriented worldview. Three functional subsystems can be distinguished, as depicted in Figure 2. They represent the roles played by the facilitator, the supporting technology, and the participants.

The facilitator sub-component is responsible for providing instructions and managing the timing of the process. The supporting technology is abstracted as a set of entities containing ideas, and accessible from a repository. The difference between electronic and non-electronic technology can be modeled, for example, by adapting the access time.

The core of the model consists of a number of participant sub-models which can be activated or not, to enable the modeling of different group sizes. A participant is modeled as an entity controlling a limited resource (cognitive capacity), and executing a sequence of time dependent cognitive processes, making decisions, and generating entities (ideas), as a result of the latter cognitive processes and decisions.

At initialization time, participants are created and assigned such properties as an identifier, a cognitive capacity, etc. After this, the participant waits until it is signaled to activate itself, through an instruction produced by the facilitator sub-model.

Following this activation, the participant is delayed to listen to the instructions and further delayed to process the instructions. From this point, the FREEBRAINSTORM thinkLet cycle begins. Based on the Bounded Ideation Theory we modeled the ideation process assuming that the solution space for the brainstorm is limited, and thus, there is an ‘imaginary’ repository of ideas [31]. The ideation routine thus consists in the retrieval of an item from the idea repository (implemented with search and remove modules), a delay to read the idea, and finally, an idea generation delay depending on the number of previously generated ideas and cognitive workload. This sequence results in a new idea added to the idea repository.

Figure 2 depicts a high level structure of the simulation model.

![Figure 2 Simulation Model Structure](image)

Various simplifying assumptions were made. For example, we consider that the time necessary for a participant to generate an idea has a baseline value determined by the individual’s cognitive limitations. In the first experiments, this baseline delay was set to 1 minute (including reading and idea generation time), as this is given as a rule of thumb in the thinkLet description [26].

Furthermore, we consider that the ideation rate is not stable throughout the brainstorm process. As time elapses and ideas are being generated, the potential for new idea generation erodes, as posited in Bounded Ideation Theory [31]. This is operationalized as an exponential decay expressed as a function of the number of ideas already generated, the task complexity, and the group size. The following expression is used in the model to determine ideation time:

\[
t_{\text{ideation}} = \frac{t_{\text{baseline ideation}}}{e^{\left(\frac{\# \text{ ideas}}{c \cdot N}\right)}}
\]

In this expression

- \(t_{\text{ideation}}\) is the time between conceiving new ideas for an individual
- \(t_{\text{baseline ideation}}\) is the baseline delay from the FREEBRAINSTORM script
- \(# \text{ ideas}\) is the current amount of ideas generated by the group
c is a constant characterizing the collaboration process

N is the size of the idea repository as assumed in Bounded Ideation Theory.

Figure 3 shows the evolution of the ideation rate during a 15 minute FREEBRAINSTORM simulation with 6 participants. We can see that proposed formulation of ideation reduces the number of ideas generated per time unit throughout the collaboration process.

![Figure 3 Simulation of decreasing ideation rate](image)

Figure 4 shows the total number of ideas generated plotted against time for the same replication of a FREEBRAINSTORM with a group size of six participants.

![Figure 4 Simulation of number of ideas generated in a FreeBrainstorm](image)

As a preliminary evaluation of the simulation model we used a data sets presented in the literature on brainstorming [3]. GroupSystems Electronic Brainstorming tool was used, in which page swapping is automated. Therefore, we know the FREEBRAINSTORM technique is used in that study. The study reports experiments on the effect of technological support and group size. For our purpose, we will focus on a replication of the experimental results with group size as the dependent variable and the number of ideas as the observable. The base model was calibrated with a group size of two participants. With 100 replications, the result depicted on Table 1 was obtained.

<table>
<thead>
<tr>
<th>Group Size</th>
<th>Average</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>24.6</td>
<td>17</td>
<td>32</td>
</tr>
<tr>
<td>4</td>
<td>43.9</td>
<td>31</td>
<td>54</td>
</tr>
<tr>
<td>6</td>
<td>67.3</td>
<td>48</td>
<td>87</td>
</tr>
</tbody>
</table>

Table 1 Simulation report for number of ideas per group size of 2, 4, and 6

For each group size (2, 4, 6), 100 simulation replications were run. The obtained results are shown on Figure 5, compared with experimental data from the study [3]. As the figure shows, the simulation model manages to give a fairly good prediction of ideation with larger group sizes.

![Figure 5 Comparing simulation and experimental results (from [3])](image)

The simulation model we present shows a very first step in this exciting new research approach. In our example we showed that we can model ideation as a process of idea generation and reading ideas of others, an interactive process, and create an outcome pattern that fits the findings of a study in which ideation is treated as a black box. In order to extend our model with more cognitive steps, we need to further investigate the implications of these effects, and gather data to estimate cognitive effort involved in these steps.
5. Conclusions

We discussed that efficient collaboration support is a critical success factor for many organizations. In order to efficiently and effectively support collaboration with tools and technology as well as process interventions, these have to be carefully designed and chosen. To ensure the predictability and transferability of collaboration processes, these can be designed based on thinkLets. Practitioners can facilitate such designed collaboration processes. However, the process often still needs to be adapted at execution time. Professional facilitators base and predict the effect such process interventions on a number of observations. Others are often hesitant to try new interventions, as they cannot foresee their impact. In both cases, interventions result in tool adaptations that require an open, extensible as well as adaptable collaboration support environment. Though first steps towards such an environment have been made by allowing users to specify adaptation rules, a challenge remains to predict the effect of the tool and process adaption on the cognitive demand of collaborative activities.

In this paper, we introduced an initial simulation model for collaboration processes which is based on thinkLets. We also introduced factors that influence the group performance in such collaboration processes. Based on a model for the FREEBRAINSTORM thinkLet, we performed simulation experiments to predict the effect of group size on ideation in a collaboration process. On a fairly basic experiment, the model showed good replicative power. After being calibrated for a small group size, the model manages to extrapolate the effects of larger groups on ideation. However, this should be considered as a very initial step. It is still early to claim the validity of the model. The purpose of introducing the model is to explore the possibilities offered by simulation to better understand collaboration processes.

In future work, we plan to extend the collaboration process model by explicitly modeling the activities within thinkLets and by using the collaboration process model to predict and evaluate the outcome of process interventions on the group performance. This will require a cognitive model explicitly representing the effect of cognitive workload as a moderator of performance. Finally, we want to integrate the collaboration model and simulation within a group support system. This will create an intelligent collaboration support environment that offers guidance to practitioners, to enable them to flexibly adapt the collaboration process and tool support at execution time to increase group productivity.

References


