Facilitating Model Construction during Inquiry Learning with Self-Generated Drawings

Master’s thesis Frank Leenaars

First supervisor: Prof. Dr. W.R. van Joolingen

Second supervisor: Dr. L. Bollen

University of Twente
Facilitating Model Construction during Inquiry Learning with Self-Generated Drawings

Frank Leenaars

The construction of models during an inquiry learning task can be an effective way to enhance learners’ insight into the domain being studied. Because of the difficulty of creating models from scratch, the learner needs support during this activity. This study introduces a new form of support for the model construction process: self-generated drawings. Findings show that most learners create realistic drawings of described real world systems. Learners who represent situations realistically also identify more important aspects of these situations than learners who represent them schematically. Access to simulations during the construction of these drawings leads to enhanced insight into the effects of variables that can be manipulated. On the other hand, participants with access to simulations thought of fewer important factors that were not explicitly available in the simulation than participants without this access. Finally, automated clusterings of participants’ drawings with the DBSCAN and LSDBC algorithms were compared and it was found that the LSDBC algorithm produced more useful clusterings.

1 Introduction

1.1 Inquiry learning and inquiry modeling

Inquiry learning, also known as (scientific) discovery learning, is a form of learning during which “students are exposed to particular questions and experiences in such a way that they “discover” for themselves the intended concepts” (Hammer, 1997, p. 489; see also De Jong & Van Joolingen, 1998; Kuhn, Black, Keselman & Kaplan, 2000). This approach to learning is rooted in constructivist theories that emphasize knowledge construction over knowledge transmission (Applefield, Huber & Moallem, 2001; Jonassen, 1991): learners acquire knowledge about the world by actively constructing their own mental models rather than by ‘copying’ information imparted by teachers. Over the course of many studies done since as early as the 1950s (see Mayer, 2004 for examples), it has become increasingly clear that pure, unguided, discovery learning is an ineffective instructional method compared to guided discovery learning. Students need guidance and support to overcome the problems they encounter during different stages of discovery learning (e.g. hypothesis generation, experiment design and data interpretation) and general regulation of learning (De Jong & Van Joolingen, 1998). This paper will explore a form of support for a variant of discovery learning called inquiry modeling.

The emphasis on mental model construction in constructivist theories is echoed in the focus on external model construction in science education literature on modeling. Hestenes (1987, p. 444) believes that “problem solving in physics is mainly a modeling process” and that “students should be taught that the key to solving a typical physics problem is the development of a model from the given information” (p. 446). Penner (2000) also argues that engaging students in the development of models should be a crucial component of science education. Inquiry modeling is a form of inquiry learning, in which learners use a modeling tool to express their ideas about a domain (Löhner, Van Joolingen, Savelbergh & Hout-Wolters, 2005). When learners are asked to create a model of a system, they have to identify the important components and
relations that make up this system. Learners can find out more about the system by performing experiments. These experiments are usually done with computer simulations, such as in the SimQuest environment (Van Joolingen, King & De Jong, 1997; Van Joolingen & De Jong, 2003). Although this is a very useful activity, especially from a constructivist perspective, it is also a difficult task to create a model from scratch in this way. Therefore learners need support from the modeling environment during this activity. This paper presents a study on a new form of support for inquiry modeling: facilitating model construction with self-generated drawings. The next section explains how learners could benefit from creating drawings during the inquiry modeling process by giving an overview of the functions of external representations in general and in an inquiry modeling task in particular.

1.2 Supporting inquiry modeling with self-generated drawings

Drawings and models created by learners both are examples of self-generated external representations. Zhang (1997, p. 180) defines external representations (ERs) as “the knowledge and structure in the environment, as physical symbols, objects, or dimensions, and as external rules, constraints, or relations embedded in physical configurations.” Cox (1999) differentiates between presented ERs (e.g. textbook diagrams) and self-generated ERs (e.g. sketches made by the learner). Like framing the same problem in different ways can affect peoples’ decisions (Tversky & Kahneman, 1981), so can different external representations of the same information cause people to use this information in different ways. Multiple examples of this effect will be given in the following three subsections, discussing presented ERs, self-generated ERs and the use of ERs in an inquiry learning context respectively.

1.2.1 Presented ERs

External representations are involved in all sorts of activities. Sometimes they simply serve as memory aids (e.g. laundry lists, lecture notes, etc.), but during many tasks they play a much more important role. For example, the sort of ER used (e.g. graphs, tables or lists) during decision making activities can have a large impact on the decision making strategies used (Kleinmuntz & Schkade, 1993). Zhang (1997) demonstrated the effects of working with different presented ERs of the same underlying structure by performing a clever experiment that let participants play four seemingly different games that were all isomorphs of Tic-Tac-Toe. Participants’ use of different strategies while playing different versions of the game was caused by differences in representation that were not present in the underlying abstract structure of the games played. This shows how different representations of the same information can cause different behaviors.

In earlier work Zhang & Norman (1994) introduced a framework for the study of tasks that require the processing of information distributed across the internal mind and the external environment, called distributed cognitive tasks. The distributed nature of these tasks lies in their components being both internal (e.g. mental models) and external (e.g. ERs). When analyzing representations within this framework, emphasis is placed on the importance of the form of the ERs used in a task. When a different ER is used to represent the same abstract structure, the difficulty and nature of the task can change.

When multiple external representations (MERs) are used during a learning activity, analyzing each ER individually is not sufficient to form a complete understanding of the learning environment. The relationships that exist between these different ERs are important as well. Ainsworth (2006) describes the DeFT (design, functions, tasks) framework for learning with multiple external representations. The framework can be used to analyze a situation in which MERs are used in learning, by looking at:

1. The design of the learning system that is used (number of ERs, information represented in these ERs, form of the ERs, sequence in which the ERs are presented and the relations between the ERs);
2. The functions that the MERs serve (e.g. do they allow computational offloading or do they re-represent information?);
3. The cognitive tasks that must be undertaken by the learner to use information from multiple external representations.

Although this framework only offers very general guidelines for the analysis of MER learning systems, it is valuable in ensuring that no relevant aspects of a system are overlooked during its evaluation.

1.2.2 Self-generated ERs

An important aspect of ERs, that is not explicitly included in the previously outlined framework, is whether the ER used is presented or self-constructed. Cox (1999) discusses the virtues of self-constructed ERs and views the externalization process as a form of self-explanation. An important property of self-constructed graphical ERs is that they are weakly expressive of abstraction. This means it is difficult or impossible to represent certain abstract relations with them, which forces learners to make relations that may have previously been abstract more concrete. For example, it is impossible to represent just the abstract relation of a fork and a plate being next to each other in a drawing. The fork has to be drawn either to the left or the right of the plate, making the relation more concrete (Mani & Johnson-Laird, 1982). When creating a drawing representing “a window in a wall” multiple choices have to be made about properties and relations that were implicit in the description, such as the size and location of this window relative to the wall.

Van Essen & Hamaker (1990) present research regarding whether encouraging elementary students to create drawings during problem solving improved performance. They found that fifth graders did better on an arithmetic word problem test when they were instructed to create drawings of these problems. Creation of drawings depended on the perceived usefulness of such a drawing, e.g. as a memory aid or for making problem characteristics more concrete. Finding the answer to a word problem was facilitated by the creation of a correct drawing in approximately 10% of the solutions. The drawings that were created also often reflected the interpretational errors that were made by students.

Drawing generation can also be used to support the learning of a text. Van Meter (2001) discusses the mixed findings as to its usefulness and offers the hypothesis that support during the construction of drawings is essential. An experiment shows that guidance during the drawing process, in the form of illustrations to be compared to learners’ own drawings and prompting questions, leads to better performance on a posttest.

1.2.3 ERs in inquiry modeling

The findings discussed in the previous subsections are very relevant for inquiry modeling, where both presented (e.g. textual descriptions and simulations results) and self-generated (e.g. drawings and models generated by the learner) ERs play important roles.

During the design of a learning environment, choices have to be made regarding the simulation interface and the presentation of simulation. Even when the same presented ERs are used in a learning environment, the way they are integrated affects the way students learn with them. Van der Meij & De Jong (2006) compared three versions of the same learning environment and found that students learned more from working with integrated, dynamically linked representations than from working with separate representations, either linked or not. Choices also have to be made concerning the sort of models the learner can create. Löhner, Van Joolingen & Savelsbergh (2003) found that the modeling tool used (textual or graphical) in an inquiry modeling task had an effect on the modeling process and its results. Students that used a graphical modeling tool ran more simulations with different models and created more complex and better models. These findings show the importance of selecting appropriate ERs in an inquiry modeling task.
An important function of self-generated drawings in the inquiry modeling process is that they help the learner identify those components and relations that are important in the studied system, because she has to make concrete those aspects of the system that were left undefined in the text (e.g. the size and location of the aforementioned window). Other possible functions of a self-generated drawing are those mentioned by Löhner et al. (2003): serve as an extended working memory, provide a layout of the problem space and (when students work in groups) provide a means for communication.

1.3 Simulation based learning
Constructivist theories consider the learner to be an active agent in the learning process. Simulation based learning fits nicely with this approach to learning, as information is not merely offered to learners, but they have to acquire their knowledge in an active, constructive way (Njoo & De Jong, 1993). Computer simulations are particularly suited for inquiry learning, because they offer learners a rich environment with low transparency (Swaak, Van Joolingen & De Jong, 1998). This means learners are able to gather a lot of information about the underlying model of the simulation by performing experiments, without being able to inspect this model directly. This means learners are required to infer knowledge about the system from experimental results.

One problem with learning from simulations is that it is hard. Many studies show that students are not able to handle their own learning process and need to be supported (e.g. De Jong, Van Joolingen, Swaak, Veermans, Limbach, King & Gureghian, 1998; De Jong & Van Joolingen, 1998). This problem has received a lot of attention (e.g. Swaak at al., 1998; Van der Meij & De Jong, 2006) and progress in creating useful forms of support is being made.

Another possible problem with simulation based learning, that has not received much attention in the literature, stems from the fact that the model on which a simulation is based, is necessarily a simplified version of the real system. This means that learners will not be able to modify all the variables that play a role in this system. Limiting the number of variables that can be manipulated in the simulation is useful in ensuring that the learner is not overwhelmed by the system’s complexity, but there is a risk that learners will have trouble thinking of other important factors in the system, that they cannot manipulate in the simulation. Learners’ mental models will be restricted by aspects of the system that are explicitly represented in the simulation. This phenomenon, known as the “focusing effect”, is known to play a role in reasoning and decision making (Legrenzi, Girotto & Johnson-Laird, 1993).

1.4 Automatic recognition of drawings
Although Cox (1999) argues that the externalization process of creating a drawing based on a textual description is useful in itself, because it forces the learner to make certain components and relations in the described system more concrete, it is likely that learners could benefit from self-generated drawings even more if they receive support during this activity. E.g. Van Meter (2001) found that students learned more from the creation of drawings if they were supported by comparing their drawing with other pictures. To support learners during creation of their drawings, the learning environment should, at least to some extent, understand what the learner is drawing.

Recognizing objects in a drawing can be done with a two step process: first, a drawing is partitioned into separate objects; secondly, sketch recognition techniques are used to identify the different objects. Dividing an image into separate parts is known as image segmentation and many different approaches to this task exist (for an overview, see Pal & Pal, 1993). One of these approaches is clustering, defined by Biçici & Yuret (2007, p. 1) as “the process of allocating points in a given dataset into disjoint and meaningful clusters”. Density based clustering is a clustering technique that locates clusters based on the density of the data points in different locations of a drawing. Because density based clustering methods are able to partition a drawing into an arbitrary number of clusters of arbitrary size, they
may be well suited to locate objects in drawings when the number or size of objects is not known beforehand.

A drawing, created with a graphics tablet, consists of a collection of points that are defined by their properties:

- x and y coordinates,
- creation time,
- color,
- pressure (the pressure on the pen at the time the point was drawn) and
- stroke membership (those points drawn between putting down and lifting up the pen belong to the same stroke).

The distance between two points can be defined using any number of these properties. E.g. the Euclidean distance, a common distance measure, can be calculated using x and y coordinates as well as the creation time and pressure information. If color and stroke information is to be used, another distance measure has to be chosen, since these variables are categorical.

After the drawing is partitioned into separate clusters in this way, information about the relative location of the different clusters and sketch recognition techniques could be used in combination to identify the objects represented by each cluster.

1.5 Research questions

The goal of this study is to answer a number of questions related to supporting the inquiry modeling process with self-generated drawings. These questions will be introduced by an example of a typical inquiry modeling activity, in which a student is asked to create a system dynamics model of the temperature in a house. The situation is described in the following short case text:

“A house is heated by a radiator that is controlled by a thermostat, but energy leaks from the house to its environment through the walls and windows. Whenever the temperature drops under a specified value (e.g. 1 °C below the preferred room temperature), the radiator is turned on by the thermostat. The radiator is turned off again when the temperature gets too high (e.g. 1 °C above the preferred room temperature).”

Besides the textual description of the system, the student also has control over a simulation of the house. This simulation shows the changes in temperature in the house during the day and lets the student control a number of parameters of the model. In this case the student can change the preferred room temperature, the average outside temperature during the day, the volume of the house and the number of windows.

To support the student in constructing a model of this system, the learning environment first asks the student to create a drawing of the house. Multiple aspects of the way the student represents described systems are interesting. Does she draw a realistic or a schematic representation? Or perhaps she combines aspects of both representation styles by adding formal elements to realistic depictions of the situation. This leads to the first research question:

1. How do learners commonly represent real world systems in a drawing?

Drawing a sketch helps the student to discover the components and relations in the system that should be a part of the model. When drawing a window for instance, the student could realize that its size is not its only relevant property for the amount of energy leaking from the house. Other relevant aspects could be curtains hanging in front of it, the material of which it is made, its position in the wall, and the side of the house it is on. It is possible to ask the student to create a drawing of the system before she has had access to the simulation or during or after it. If the student creates the drawing during or after access to the simulation, the drawing may not only be based on the case text, but also on the simulation. And although computer simulations provide insight into the effects of changing certain variables, a possible drawback of their use is that it could restrict the learner’s mental model of the described system to
those parameters that can be manipulated in the simulation. Therefore we ask the following questions:

2. Do students with access to a simulation show a better understanding of important variables and relations in a described system than students without access to a simulation?

3. Do students with access to a simulation think of fewer important factors in a system, that are not explicitly represented, than students without access to a simulation?

Finally, if the modeling environment is to support the student during the task of creating a model, based on her drawing, the environment should to a certain extent understand what the student has drawn. Thus the final research question is:

4. Can existing cluster and sketch recognition techniques be used to successfully identify objects in the students’ drawings?

2 Method

2.1 Participants

Thirty-seven undergraduate students at the University of Twente, who had taken advanced physics courses at the highest level of high school (Dutch VWO), participated in an experiment that lasted 30-45 minutes. Students that needed it received “participant credits” for their participation, but most received no compensation.

2.2 Materials

Participants worked with a laptop and drawing tablet in an integrated drawing and simulation environment. The drawing and simulation environment participants worked in is shown in Figure 1.

Figure 1. The drawing and simulation environment participants worked in.
environment that was created specifically for this study. Participants could use pencil and eraser tools to create drawings and a labeling tool to make notes on certain aspects of their drawing. The pencil tool responded to the pressure that was applied to the pencil, drawing a thicker line if the participant pressed harder on it. The simulation tool was based on SimQuest (Van Joolingen & De Jong, 2003). Figure 1 shows this environment. Detailed logs of all the user’s interactions with the software were automatically saved.

The environment offered three tutorials, which taught participants to use the drawing tablet and the software, followed by two cases. Short descriptions of these tutorials and cases will be given in the following subsections.

2.2.1 First tutorial: connect the dots
The first tutorial consisted of a simple ‘connect the dots’ worksheet that familiarized participants with the pencil and eraser tools. To simplify sketch analysis, the eraser tool allowed only stroke wise deletion.

2.2.2 Second tutorial: background information
The second tutorial served two functions: offer some writing practice to participants and collect background information. Participants were asked to give handwritten information about their gender, their major, how long they had been in college and their grades for math and physics in high school.

2.2.3 Third tutorial: labeling
This tutorial introduced the labeling tool and prepared participants for the two cases. The labeling tool allowed participants to add labels to any part of their drawing. Text could be typed in these labels and they could be moved around after they were created. Four labels are shown in the drawing in Figure 1.

Participants were shown a picture of water flowing from a faucet into a leaky bucket and asked to add labels to those parts of the picture that were relevant if one wanted to calculate how much water was in the bucket after a specified time. For instance, the discharge of the faucet and the size of a hole in the bucket are relevant parts of the picture. This task, identifying and labeling relevant components of a system, was also an important activity during the two cases.

2.2.4 First case: toy car
Participants received the following instructions (translated from Dutch original):

“A toy car is connected to a table leg with an elastic string. The car contains a small engine that produces a constant forward force on the car. The engine is switched on and the car starts to move away from the table leg, causing the elastic string to be pulled tight. Because the string pulls the car backwards, the car may start to oscillate, but it is also possible that it slowly comes to a halt without oscillating. Make a sketch of this situation and use labels to identify those parts of your drawing that have an effect on whether or not the car will oscillate. Decide whether the variables mentioned below are relevant, and try to think of as many other relevant variables as possible.”

Below these instructions were either some suggestions for relevant factors (text only condition) or a simulation that allowed participants to manipulate these same variables (simulation condition). These variables were:

- mass of the car,
- force produced by the car’s engine,
- length of the relaxed elastic string and
- force constant of the elastic string.

In the simulation condition participants could use sliders to set values for these variables within a given range. The results of the simulation were plotted in a time-distance graph. Finally participants were asked to specify what the effects of the relevant variables were. E.g. would a greater mass of the car lead to more or less (chance of) oscillation?
2.2.5 Second case: temperature regulation
The second case had the same structure as the first case. Participants received a case description very similar to the one in the example in section 1.3 and were asked to identify those parts of the system that had an effect on the frequency with which the radiator was switched on and off. Below the description were again either some suggestions or a simulation with the following variables:

- preferred temperature in the house,
- average outside temperature,
- volume of the house and
- total surface area of the windows.

When the simulation was run, the indoor and outdoor temperatures during the day were plotted in a time-temperature graph. The outdoor temperature varied around the average outside temperature and was based on a sinusoid with a 24 hour period. Results of this simulation can be seen in Figure 1.

2.3 Procedure
Each participant was randomly assigned to either the text-only (n = 19) or the simulation group (n = 18). After receiving some brief instruction from the experiment leader about using the drawing tablet, participants went through the three tutorials and two cases. There was no time limit for the tutorials and cases and participants were told they could go to the next task when they felt they were done with the current one.

2.4 Analysis

2.4.1 Representation of the situation
Drawings were classified as either realistic or schematic representations. Drawings were only classified as schematic when it was impossible to recognize that a car (first case) or a house (second case) was being represented. For instance, this was the case when the toy car was drawn as a block with arrows representing the different forces acting upon it. The number of drawings that contain formal elements, such as the arrows representing forces, was also examined.

Also of interest was how much of the situation was represented by participants. E.g. do participants draw the ground in the first case and the radiator in the second case? Whether or not participants represent these aspects of the situation could have an effect on which relevant aspects of the situation they find. For instance, when participants draw the ground in the first case, they could realize that it is not necessarily flat (nothing is said about this in the case description), which would have an effect on the movement of the car.

2.4.2 Effects of simulations
Participants’ understanding of the different variables and relations playing a role in the cases was measured by scoring them on four different aspects in both cases. For each of these aspects, participants received either 0 or 1 point(s).

For the first case, participants were scored on correctly assessing the effect of each of the four mentioned variables (mass of the car, force produced by the engine, length of the relaxed elastic string and force constant of the elastic string) on whether or not the car would start to oscillate.

For the second case, participants were scored differently, because the effect of three of the four mentioned variables (preferred inside temperature, average outside temperature, volume of the house, total surface area of the windows) was not straightforward. Only the volume of the house had a clear effect on the frequency with which the heating was turned on and off. Participants received points for mentioning that the difference between preferred and outside temperature was relevant (as opposed to either one of these individually), for mentioning the heating was always on or off in some situations and for mentioning that the frequency of the heater’s on/off cycle depended on both the warming up and cooling down period.

Further details of this scoring system can be found in Appendix 1. Using this system, each participant received a score ranging from 0 to 4 points for both cases. Differences between the scores of participants in the text-only and simulation
conditions were determined by using an independent two-sample t-test. This scoring system was believed to be quite deterministic. To make sure this was so, one third of the participants’ drawings were scored by a second rater. Inter-rater reliability was calculated using Cohen’s kappa.

The number of other relevant factors participants thought of in each case was measured by simply counting the number of new factors that were mentioned. The difference between participants in the text-only and simulation conditions was again determined with an independent two-sample t-test.

2.4.3 Clustering
To find out how well current clustering techniques are able to differentiate between the different objects represented in the drawings, two different density based clustering algorithms were implemented and used to analyze the drawings. The first algorithm used was the DBSCAN algorithm (Ester, Kriegel, Sander & Xu, 1996; Sander, Ester, Kriegel & Xu, 1998) and the second algorithm was the LSDBC algorithm (Biçici & Yuret, 2007). Several parameter values and distance measures were tried for both algorithms.

3 Results
3.1 Representation of the situations
The toy car in the first case was represented realistically by 35 out of 37 participants (95%). Arrows representing the forces that played a role in this situation were added by 17 participants (46%). None of the participants chose to represent the car with a free body diagram of just the block and the forces acting upon it; all participants represented the table leg and elastic string explicitly. The only other aspect of the situation that was frequently represented by participants was the ground. This was drawn by 12 participants (32%).

The house in the second case was represented realistically by 29 out of 37 participants (78%). Arrows were used to represent air flow or heat transfer by 9 participants (24%). Much more variation in the way participants represented the situation existed in the second case than in the first case. Some participants drew a house with a see-through wall on the front side, others drew solid houses; some drew multiple stories, other drew just one room; some drew the house in 3D others in 2D. Although the eight participants that represented the house schematically were evenly divided over the simulation and text-only conditions (four in each group), there was a relation with the number of other factors these participants mentioned. They mentioned significantly fewer other factors than participants that represented the situation realistically (t = 2.3; p = 0.03; standardized mean difference (SMD) = 0.7).

3.2 Effects of simulations
Table 1 shows the means of the scores and number of other factors mentioned for each condition in both cases. For the first case the difference between the scores of the text-only and simulation groups was significant (t = 7.1; p < 0.0005). As hypothesized, participants in the simulation group scored higher (SMD = 1.52) than participants in the text-only group. Inter-rater reliability was high (kappa = 0.84). The difference between the number of other relevant factors participants mentioned was also

<table>
<thead>
<tr>
<th>Scores and Number of Other Factors Mentioned</th>
</tr>
</thead>
<tbody>
<tr>
<td>First case: toy car</td>
</tr>
<tr>
<td>Score</td>
</tr>
<tr>
<td>Text-only</td>
</tr>
<tr>
<td>Simulation</td>
</tr>
</tbody>
</table>

Note. Values are means of the scores and number of other factors mentioned for the text-only and simulation conditions in both cases (standard deviation in parentheses). The minimum and maximum scores were 0 and 4 respectively.
significant ($t = 2.7; p = 0.006$). Participants in the simulation group thought of fewer other relevant factors than those in the text-only group (SMD = 0.82).

For the second case the difference between the scores of the text-only and simulation groups was also significant ($t = 2.6; p = 0.007$). Participants in the simulation group scored higher than those in the text-only group (SMD = 0.79). Inter-rater reliability was very high (kappa = 0.96). There was a trend towards participants in the text-only group mentioning more other factors than those in the simulation group, but this was not significant ($t = 1.6; p = 0.06; SMD = 0.89$). Closer inspection of the log files showed that the three participants in the simulation group that mentioned the most other

![Figure 2. Clustering results. Separate clusters are indicated by different colors. The top row shows results from the GDBSCAN algorithm using a Euclidean distance measure only taking the x and y coordinates into account. The middle two rows show results from the GDBSCAN algorithm using a distance measure that also takes time into account. The bottom row shows results from the LSDBSCAN algorithm using the same distance measure as the middle rows.](image-url)
relevant factors, did this before using the simulation. Only after they had created a drawing of the situation and labeled those aspects of the drawing they believed were relevant, did they start using the simulation to gain more insight into the effects of the variables that could be manipulated. This means that at the time they were thinking of the other relevant factors, they had the same information as those in the text-only group. When these three participants are added to the text-only group, the difference regarding the number of other factors does become significant ($t = 3.4; p = 0.001; \text{SMD} = 1.1$).

3.3 Clustering

3.3.1 DBSCAN

The first algorithm tested was the DBSCAN (Density-Based Spatial Clustering of Applications with Noise) algorithm (Ester et al., 1996). This density based clustering algorithm takes a collection of data points and two parameters as input and returns a cluster id for each point. A detailed description of the inner workings of this algorithm is outside the scope of this paper, but a brief overview will be given to explain the meaning of the two parameters.

Clusters of points consist of core points and border points. For a point to be considered a core point, it has to have at least $\text{MinCard}$ (first parameter) neighbors. Two points are considered to be neighbors when the distance between them is smaller than $\text{Eps}$ (the second parameter). Clusters are found by selecting a point that is not yet part of a cluster and checking if it is a core point. If this is the case, this point and its neighbors are given the same cluster id. Then all the neighbors that are core points as well are expanded (i.e. their neighbors are checked and, if they are core points, expanded).

Clusters in self-generated drawings are usually collections of lines, which are essentially very stretched out groups of points. For this reason the value chosen for the $\text{MinCard}$ parameter is always 1, so that all points in these lines will be considered core points and these stretched out groups will be considered part of the same cluster.

Two different distance measures were used with the DBSCAN algorithm. The first distance measure is based on the Euclidean distance and only uses the x and y coordinates of the points. The first row of Figure 2 shows the results of the DBSCAN algorithm using this distance measure for three different drawings. It is clear that these clusterings are not very useful. In the first drawing, the table, the elastic string and the car are all in the same cluster and in the third drawing the table and the car are both divided over many different clusters. The clustering of the second drawing is more useful, but still not great. The second distance measure is also based on the Euclidean distance, but takes into account the time at which the points were drawn as well. The two middle rows in Figure 2 show the results of using the DBSCAN with this distance measure. The clusterings in the first drawing on the second row and the second and third drawings on the third row are much more useful than those in the first row. However, the differences in the value for $\text{Eps}$ in the second and third row show the main problem with the DBSCAN algorithm: the optimal $\text{Eps}$ value is different for each sketch. A smaller $\text{Eps}$ value leads to better clustering results in the first drawing, but to very bad results for the second and third drawings. A larger $\text{Eps}$ value is much better for these drawings, but leads to a bad clustering of the first drawing: the table, elastic string and car are again considered to be a part of the same cluster. The second clustering algorithm that was tried, the LSDBC algorithm, offers a solution to this problem and will be discussed in the next section.

3.3.2 LSDBC

The LSDBC (locally scaled density based clustering) algorithm, like the DBSCAN algorithm, takes a collection of data points and two parameters and returns a cluster id for each point. This algorithm starts by finding the densest region in the collection of points. This is done by finding the point that is closest to its $k$-th (first parameter) nearest neighbor. For instance, if $k = 5$, the point that is closest to its fifth nearest neighbor is considered to have the highest density and is used as a starting point for finding a cluster. The $\text{Eps}$ value of each point is equal
to the distance between that point and its fifth nearest neighbor. The cluster is then expanded from this point in a similar way as in the DBSCAN algorithm, as long as the Eps values of the points that are added are not much larger than the Eps value of the original seed point of the cluster. The allowed increase in Eps value is defined by $\alpha$, the second parameter.

Optimal values for $k$ and $\alpha$ were experimentally found to be 10 and 20 respectively. The bottom row in Figure 2 shows the clustering results of the LSDBC algorithm using the same distance measure as the middle rows. Even though the parameters were kept constant for these three drawings, the results were as good as or better than DBSCAN results with optimized parameters.

3.3.3 A new distance measure
A problem with including time difference in a measure of distance is that different drawing directions lead to different clustering results. Figure 3 shows two pair of strokes that would look the same on paper, but are drawn differently. A time based distance measure would likely consider the first pair of strokes to be part of the same cluster, but not the second pair. The reason for this is the large distance in the horizontal dimension between the end point of the top stroke and the start point of the bottom stroke in the second pair. At the same time the end points of both strokes are considered to be far apart in the time dimension. To deal with this problem a new distance measure was created.

This distance measure calculated the difference between the $x$ and $y$ coordinates of two points in the same way as the previously mentioned distance measures, but the time difference was calculated differently. If stroke A was drawn before stroke B, the time difference between a point in stroke A and a point in stroke B was considered equal to the time difference between the last point in stroke A and the first point in point B. Using this distance measure the second pair of strokes would be considered to be part of the same cluster. Analysis of participants' sketches for the first case, the toy car, shows an improved clustering result with this distance measure in 3 out of 37 drawings, with no worse results for the other drawings.

4 Conclusion and discussion
The goal of this study was to examine three different aspects of supporting the inquiry modeling process with self-generated drawings. First, the way learners represent real world systems was examined. Secondly, the effect of access to a simulation on learners' understanding and creativity was analyzed. Finally, an exploratory examination of using clustering techniques in preparation for further automated sketch recognition was performed.

Analysis of the way the situations described in the two cases were represented, shows that most participants chose to represent these situations realistically. An interesting finding was that those learners that represented the house in the second case schematically thought of fewer relevant aspects of the situation that were not explicitly mentioned. This finding is in line with expectations based on the externalization process (Cox, 1999). For example, when students don't draw a window in their schematic representation, they do not have to make decisions about such properties as its size and location. Therefore they may be less likely to think of these aspects of the situation as being relevant. Since representing the situation realistically or

![Figure 3](image-url)
schematically was not an independent variable in this study, no causal links between representation style and number of relevant factors found can be assumed. Further study regarding this possible effect is suggested, because it would have implications for the best way to ask learners to represent a situation, if finding relevant aspects is the goal of this activity.

The higher average score for participants in the simulation group than for those in the text-only group indicates that access to the simulations leads to greater insight into the effects of the mentioned variables. This is of course the expected result and the reason simulations are a valuable tool in inquiry learning environments. On the other hand, participants in the simulation group thought of fewer other factors that were relevant in the situation. This result can be explained as an occurrence of the focusing effect, as described in Legrenzi et al. (1993). In this case it is a particularly strong effect, since the variables that could be manipulated in the simulations were also explicitly mentioned in the text-only condition. It is likely that manipulating variables during experiments with the simulation caused participants to focus on these variables more than merely reading about them as suggestions in the text-only condition. Particularly interesting was the finding that the three participants in the simulation group that mentioned the most other relevant factors in the second case, did so before using the simulation for the first time. These students may have hit upon the best way to combine self-generated drawings and simulations: first make a drawing of the situation and think about the relevant aspects, and only then use the simulation to gain further insight into the effects of certain variables. This gives learners the best of both worlds: the improved insight gained from working with the simulation, without the constrained creativity caused by the focusing effect.

The analysis of clustering results in this study is of a very exploratory and qualitative nature, but some useful results have been found. The LSDBC algorithm performed better than the DBSCAN algorithm across the board, even when the parameters of the DBSCAN algorithm were optimized for each drawing and the parameters of the LSDBC algorithm were kept constant. A new distance measure is introduced that takes stroke membership into account when calculating the time distance between points, leading to slightly improved clustering results. Clustering could possibly be further improved by using the labels participants added to their drawings as seed points (e.g. Patrick, 1973). The text in these labels could also be useful during a later object recognition phase, although this would require a dictionary with words for each case. In the first case for instance, if a label contains the word ‘car’ it is likely that the cluster to which it points contains the toy car. The finding that most participants represented the situations realistically rather than schematically means that sketch recognition techniques could be used to identify objects in most drawings.

When application of these clustering and sketch recognition techniques leads to accurate identification of objects, different forms of support for the modeling process are possible. One relatively simple way to support learners is to give feedback about important aspects of the situation that are not yet represented in the learner’s drawing. For example, an initially obscured list of important aspects of the situation could be displayed which is updated each time the learner draws an important component of the system. In the temperature regulation case, these could be objects like a window or the sun. When the learner draws one of these objects, this specific element of the list would then be shown and checked off. This may stimulate learners to find all the important aspects of the situation to complete the list. Bollen, Van Joolingen & Leenaars (2009) introduce a more extensive form of support for the modeling process: modeling with inaccurate drawings. In this approach the drawing itself is transformed into a model of the situation. This could be accomplished by either translating this drawing into a formal model or by adding information to it in such a way that it becomes a computational model. The latter option could be used to create an animated version of the drawing.
This could for example show the toy car from the first case moving forwards and backwards, while the elastic string is alternatively extended and relaxed. Arrows added to the drawing by participants could be used to identify the different forces or processes that play a role in this model.

The findings presented in this paper open the door to further research where self-generated drawings are actually used during an inquiry modeling task. This could show whether learners that were previously asked to create a drawing of the situation and identify the important aspects of this situation create better models than learners who were not asked to do this.

Acknowledgements
I’d like to thank my supervisors, Wouter van Joolingen and Lars Bollen for their advice and enthusiasm. I’d also like to thank Annika de Goede for being a first-rate second rater.

References


Appendix 1: scoring system

First case: toy car
Two out of four of the mentioned factors have an effect on whether or not the car will start to oscillate: the mass of the car and the force constant of the elastic string. Both of these factors are positively correlated with the chance the car will oscillate.

<table>
<thead>
<tr>
<th>Factor</th>
<th>Effect on oscillation</th>
<th>Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mass of the car</td>
<td>The mass of the car has an effect. Larger mass means the car is more likely to oscillate.</td>
<td>Not relevant: 0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Negative correlation: 0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Positive correlation: 1</td>
</tr>
<tr>
<td>Force produced by the engine</td>
<td>The force produced by the engine has no effect on whether or not the car will oscillate. It does have an effect on the amplitude of the oscillation.</td>
<td>Relevant: 0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Not relevant: 1</td>
</tr>
<tr>
<td>Length of the relaxed elastic string</td>
<td>Length of the relaxed elastic string has no effect on whether or not the car will oscillate.</td>
<td>Relevant: 0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Not relevant: 1</td>
</tr>
<tr>
<td>Force constant of the elastic string</td>
<td>The force constant of the elastic string has an effect. Larger force constant means the car is more likely to oscillate.</td>
<td>Not relevant: 0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Negative correlation: 0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Positive correlation: 1</td>
</tr>
</tbody>
</table>

Second case: temperature regulation
All four of the mentioned factors have an effect on the frequency with which the radiator is switched on and off. However, the effect of these factors is not as straightforward as in the first case: there is a lot of interaction between the relevant variables. The volume of the house is an exception to this rule: larger volume leads to a lower frequency.

<table>
<thead>
<tr>
<th>Factor</th>
<th>Effect on frequency</th>
<th>Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preferred inside temperature</td>
<td>By itself the preferred temperature and the average outside temperature don’t mean much. However, the difference between these two is relevant.</td>
<td>Participant mentions difference between these two as relevant: 1</td>
</tr>
<tr>
<td>Average outside temperature</td>
<td>The volume of the house has an effect. A larger volume leads to a lower frequency.</td>
<td>Not relevant: 0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Negative correlation: 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Positive correlation: 0</td>
</tr>
<tr>
<td>Total surface area of the windows</td>
<td>The total surface area of the windows has an effect, but this effect is not straightforward. A larger surface can lead to both a higher and a lower frequency in different conditions.</td>
<td>No points awarded</td>
</tr>
</tbody>
</table>

Participants were awarded points for mentioning two other aspects of the situation:
- The heater is always on or off in certain situations: 1 point
- The frequency of the heater’s on/off cycle depends on both the time it takes to warm up the house and the time it takes for the house to cool down. This is especially relevant when considering the effect of the total window surface. A larger window surface often means it takes more time for the house to warm up, but less time for it to cool down: 1 point