Learning and Instruction 21 (2011) 614–624

Learning and Instruction

www.elsevier.com/locate/learninstruc

Comparing two types of model progression in an inquiry learning environment with modelling facilities

Yvonne G. Mulder*, Ard W. Lazonder, Ton de Jong

Department of Instructional Technology, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands

Received 19 February 2010; revised 13 January 2011; accepted 23 January 2011

Abstract

The educational advantages of inquiry learning environments that incorporate modelling facilities are often challenged by students' poor inquiry skills. This study examined two types of model progression as means to compensate for these skill deficiencies. Model order progression (MOP), the predicted optimal variant, gradually increases the specificity of the relations between variables, whereas model elaboration progression (MEP) gradually expands the number of variables in the task. The study utilized a between-subject design with three conditions: a MOP condition ($n = 28$), a MEP condition ($n = 26$), and a control condition without model progression ($n = 30$). Consistent with expectations, model progression enhanced students' task performance; a comparison among the two model progression conditions confirmed the predicted superiority of the MOP condition. These results are discussed in relation to the inconsistent findings from prior research. Based on this discussion ways to optimize model order progression are advanced.

Keywords: Inquiry learning; Model-based learning; Model progression

1. Introduction

Computer-supported inquiry learning environments essentially enable students to learn science by doing science, offering resources to develop a deep understanding of a domain by engaging in scientific reasoning processes such as hypothesis generation, experimentation, and evidence evaluation. Computer simulations have long been incorporated in these environments, and are today increasingly being supplemented with opportunities for students to build computer models of the phenomena they are investigating via the simulation. As in authentic scientific inquiry, modelling is considered an integral part of the inquiry learning process as students can build models to express their understanding of the relation between variables (van Joolingen, de Jong, Lazonder, Savelsbergh, & Manlove, 2005; White, Shimoda, & Frederiksen, 1999). Students can check their understanding by running the model; evidence evaluation then occurs by weighting model output against prior knowledge or the data from the simulation. These comparisons yield further insight into the phenomenon and assist students in generating new hypotheses.

The educational advantages of inquiry learning are often challenged by the students' poor inquiry skills. de Jong and van Joolingen's (1998) review showed that many students experience difficulties during simulation-based inquiry learning. For example, students are unable to infer hypotheses from data, design inconclusive experiments, show inefficient experimentation behaviour, and ignore incompatible data. Students also experience difficulties during modelling. Hogan and Thomas (2001) noticed that students often fail to engage in dynamic iterations between examining output and revising models, and merely use output at the end of a session to check if the model's behaviour matches their expectations. A related problem concerns the students' lack of persistence in debugging their model to fine-tune its behaviour (Stratford, Krajcik, & Soloway, 1998).

These findings suggest that students' difficulties with inquiry and modelling both lie at a conceptual level. When

* Corresponding author. Tel.: +31 53 489 4857; fax: +31 53 489 2849.
E-mail address: y.g.mulder@utwente.nl (Y.G. Mulder).

0959-4752/$ - see front matter
doi:10.1016/j.learninstruc.2011.01.003

provided with a simulation, most students manage to design and execute experiments; inferring knowledge from these experiments appears to be the major source of difficulty. Likewise, students are capable of building syntactically correct models, but often fail to relate knowledge about phenomena to those models (Sins, Savelsbergh, & van Joolingen, 2005). As this ineffective behaviour is a serious obstacle to learning, additional support is needed in order for inquiry learning and modelling to be effective.

Mulder, Lazonder, and de Jong (2010) were among the first to identify guidelines for supporting learners during inquiry learning with modelling. Their study compared domain novices' unsupported inquiry behaviour and performance to that of a considerably more knowledgeable reference group (referred to as 'experts'). Using Klahr and Dunbar's (1988) SDDS model as framework, the analyses focused on the processes of hypothesis generation, experimentation, and evidence evaluation. Results indicated that novices and experts were quite comparable with regard to these processes, suggesting that novices predominantly exhibited expert-like behaviour. However, as the novices received no support whatsoever, they induced virtually no knowledge from their inquiry and modelling activities (as also indicated by Dean & Kuhn, 2007; Klahr & Nigam, 2004; Mayer, 2004). Subsequent qualitative analyses provided starting points for the design of learner support. Contrary to expectations, novice learners knew quite well which elements to include in their models, and even their initial models contained nearly all correct elements. Generating the relationships between those elements appeared to be considerably more problematic. Novices generated and tested hypotheses about relations that were so specific that it is highly unlikely that these hypotheses originated from their inquiry or modelling activities, suggesting that they were merely based on guesswork. From these findings, Mulder et al. (2010) concluded that learner support should assist students in identifying the relations between the elements in their models.

Some learning environments such as Co-Lab (van Joolingen et al., 2005), LinkIt (Ogborn, 1998) and Model-it (Krajcik et al., 2000) already offer this support in a rather unobtrusive way by giving learners a choice as to how detailed they want to specify relationships. Learners can opt for a self-generated, full-fledged scientific formula (i.e., quantitative relations), or select less detailed pre-specified, qualitative relations from a drop-down menu (i.e., qualitative relations). According to Löhner, van Joolingen, and Savelsbergh (2003) such qualitatively specified models are more appropriate at the beginning of the modelling process when learners do not yet have a clear idea about the model they are making. It is therefore surprising that participants in the Mulder et al. (2010) study, who received no support, hardly used the possibility to state qualitative relations.

In view of these findings, it might be more fruitful to restrain domain novices' natural tendency to engage in quantitative modelling from scratch by first having them create models that are qualitatively specified, and then enabling them to transfer these qualitative relations into quantitative ones. This form of model progression is generally assumed to be beneficial to novice learners (de Jong, 2005; White & Frederiksen, 1990). The goal of the present study was to empirically assess the effectiveness of this type of scaffolding.

The idea of model progression was coined by White and Frederiksen (1990) who used it to create problem sets that motivate successive refinements to the students' mental models. They distinguished three dimensions on which models may vary: their perspective, their degree of elaboration, and their order. Lateral progressions that represent alternative means of understanding the domain involve changes in model *perspective*. In the domain of electrical circuits, for instance, models describing Kirchhoff's Voltage Law use a different perspective from models describing Coulomb's Law. Upward progressions to more sophisticated models involve changes in a model's degree of elaboration and order. The *degree of elaboration* is determined by the number of variables and relations in a model. The core idea of model elaboration progression is therefore to let students start off with a simplified version of the phenomena; additional variables (and their relations) are introduced step by step over the course of the session so as to expose the students gradually to the full-complex model. The *order* of a model concerns what type of reasoning it supports (i.e., qualitative reasoning or quantitative reasoning). White and Frederiksen postulated that a qualitative understanding needs to be developed before a quantitative understanding should occur. A further distinction is made in the qualitative understanding; the focus should initially be on the students' reasoning about the presence or absence of elements from the phenomena under investigation, and subsequently change to reasoning on the basis of incremental changes of these elements.

Model order progression resembles the type of scaffolding that was advocated on the basis of the Mulder et al. (2010) study. Model elaboration progression represents a viable alternative to provide learners with increasingly sophisticated models about a domain, and a comparison among these two forms of model progression could validate the alleged benefits of the former. Model perspective progression was not included in this comparison because there is no inherent increasing complexity associated with offering different perspectives of the same phenomena. As model perspective progression is not relevant to the purpose of the present study, the remaining part of this section discusses research on the simple-to-complex organization of learning materials.

Gradually introducing learners to increasingly more sophisticated or comprehensive subject matter has long since been recognized as powerful instructional strategy (e.g., Gagné, 1977; Reigeluth & Stein, 1983). Early attempts to reducing complexity during initial learning with computers can be found in the field of software training. Carroll and Carrithers (1984) provided novice learners with a so-called training-wheels system for learning to use a word processor. The key characteristic of this system is that features of the word processor new users typically do not need, but which can be springboards for errors and confusions, were disabled. Carroll and Carrithers reasoned that in a reduced interface learners are prevented from getting caught up in tangles of error and confusion, and as such will spend less time on errors. In their first experiment, the participants were 'learning by doing', they were given an example letter which they had to reproduce with the word processor. Participants who were provided with the training-wheels version of the word processor performed faster and more successful overall than participants who worked with the complete version of the program.

Research on combining a training-wheels system with other support has produced mixed results. Carroll and Carrithers (1984; experiment 2) demonstrated that training-wheels have added value to an instructional manual: learners working with an instructional manual in a simplified version of the domain performed faster and more successful compared to learners working with a manual in the full-complex domain. Results further showed that the training-wheels reduced learners' time spent recovering from errors, which most likely accounts for the instructional efficacy of the training-wheels system (cf. Lazonder & van der Meij, 1995). However, Spannagel, Girwids, Löthe, Zendler, and Schroeder (2008) found that learning to use a spreadsheet program with animated instructions, predominantly led to better performance than learning with text manuals, and that a training-wheels interface did not yield better results for students who learned with animations.

In a more recent study, Löhner et al. (2003) replicated the training-wheels findings in the domain of modelling the temperature inside a house. They compared the performance of a textual modelling group and a graphical modelling group. Participants in the textual modelling group had to build a full-complex model by specifying the relations between variables in precise, quantitative form. Participants in the graphical modelling group, in contrast, only had to indicate whether relations were positive or negative (i.e., specifying the relations qualitatively); the underlying mathematical specifications of the relations were handled by the system. Students using the graphical representation were found to switch quickly from one relation to the next, and try every idea that came up, which might be a viable strategy for the initial stages of a modelling process. Löhner et al. therefore concluded that at the beginning of an inquiry process, novice learners benefit from building qualitatively specified models compared to building quantitative models.

Although these studies demonstrate that starting off in a simple form can be beneficial to learning, they did not take the progression to higher levels of complexity into consideration. Supportive evidence on this matter can be found in the literature on learning with simulations. In a study by Alessi (1995), model progression pertained to the fidelity of the simulation; the general idea behind this form of progression was to go from 'simplified' to more 'realistic' simulations. Alessi assumed that fidelity progression would enhance learning because a simplified simulation supposedly facilitates initial learning whereas high fidelity is expected to be better from a transfer point of view. To validate this claim, three groups were compared that learned procedural knowledge about how to use a multimeter either with a low-fidelity simulation, fidelity progression simulations, or a high-fidelity simulation. The results confirmed some of Alessi's expectations: even though the three groups did not differ on measures of learning while working with the simulations, the high fidelity and progression simulations were found to enhance performance on a transfer task, thus supporting the notion that high fidelity simulations are superior in transfer. More recently, Zacharia and Olympiou (2011) were unable to replicate these findings. Participants who progressed from experimenting with a simulation to experimenting with the real equipment were found to learn as much as those who experimented with the real equipment only. However, participants who only worked with the simulation performed as well as participants in the other two conditions.

Other studies investigated model elaboration progression. Rieber and Parmley (1995) compared performance of students who were presented with either a structured or an unstructured simulation regarding the physics principles of Newtonian mechanics. The structured simulation consisted of a series of four activities in which students were given increasing levels of control over a simulated, free-floating object. This simulation was considered 'structured' because each activity included a controlled number of new subskills, and each successive activity incorporated the subskills of the preceding activity. The unstructured simulation consisted of an open-ended and unstructured activity in which subjects assumed full control over the floating object from the very beginning. Results indicated that students in the structured condition outperformed students in the unstructured condition.

Swaak, van Joolingen, and de Jong (1998) replicated these findings in the domain of oscillatory motion. The type of motion in their study depended on the presence of friction and/or an external force. In case both are absent, the motion is free; if only friction is present, the oscillation is damped; and if both are present, there is forced oscillatory motion. Model progression pertained to the degree of elaboration in that learners were first given a simulation about free oscillatory motion, then a simulation of damped motion, and finally a simulation on forced oscillatory motion. The study's main finding was that students in the model progression condition developed more intuitive knowledge about oscillatory motion than students from a control group who received no model progression. Results further indicated that adding assignments to the model progression has no significant facilitating or deteriorating effect.

Quinn and Alessi (1994), however, found less facilitative effects of model progression. They compared groups that differed in whether the simulation was presented in its most complex form initially or whether it was presented in sections of increasing complexity. The computer simulation was a model of the spread of an influenza epidemic in which the number of people ill with influenza depended on four variables: the number of contacts per person per week, the time to illness, the duration of illness, and the length of the immune period. One group worked with a simulation in which all four variables were present, whereas an other (model elaboration progression) group worked on a simulation in which the variables were introduced gradually. Students in the latter group initially performed better than students who worked with the full-complex simulation, but this effect faded out upon completion of the task.

De Jong et al. (1999) combined model perspective and elaboration progression in a simulation on collisions. They divided the domain into five progression levels. The model's degree of elaboration was progressed in the first three levels; the last two levels offered two alternative perspectives on collisions. Contrary to Swaak et al. (1998), this study showed no main effect of model progression, which was allegedly due to the level of task complexity. Collisions is a relatively straightforward domain that might not be complicated enough for the effects of model progression to show. This explanation was substantiated by the fact that participants in this study had considerable prior knowledge, and therefore might not have needed the first three level of model progression.

To conclude, model progression has been investigated in slightly different configurations, task domains, and for different types of knowledge. These cross-study variations could be the reason why results on the effectiveness of model elaboration progression are inconclusive. Another, perhaps more plausible explanation is that model elaboration progression is a suboptimal way to arrange learning tasks in a simple-to-complex sequence. Various authors have postulated that model order progression better meets the learning needs of domain novices (Löhner et al., 2003, Mulder et al., 2010; Veenman & Elshout, 1995; White & Frederiksen, 1990), but the effectiveness of this type of scaffolding has neither been assessed nor compared to that of model elaboration progression. Both issues were central to the research reported below.

2. Research design and hypotheses

This study aimed to scaffold domain novices on an inquiry learning task with model progression in which the model order is progressed. The study utilized a between-subject design with three conditions. Students in the *model order progression (MOP)* condition had to build increasingly more specific models. They received a full simulation with four variables, and had to model its behaviour in three consecutive phases. Modelling thus progressed from indicating the presence of all elements and relations in the model, through a qualitative specification of these relations, to a quantitative specification. Performance in this condition was compared to two reference groups. In one group, the *model elaboration progression (MEP)* condition, students had to build increasingly more comprehensive models. Their version of the simulation contained four variables that were introduced one at a time. Students' task was to build the model underlying each simulation quantitatively from scratch. The second reference group was not scaffolded by model progression. Students in this *control* condition received the full-complex simulation and had to infer and build the quantitative model that governed its behaviour from scratch and without any externally-imposed structuring.

In line with prior research, gradually introducing learners to increasingly more sophisticated or comprehensive subject matter was expected to enhance performance success (Hypothesis 1). This hypothesis was examined by comparing performance success of both model progression conditions together with the control condition. Prior research further suggests that the effectiveness of model progression depends on which dimension progresses. Progression of model order was predicted to yield higher performance success than model elaboration progression (Hypothesis 2). More precisely, as the identification of model elements was not explicitly supported in either model progression condition, there was no reason to assume any cross-condition differences on this measure. Thus no specific predictions regarding the number of relevant elements could be formulated. However, as MOP was intended to scaffold learners' relation construction—a key problem to domain novices—MOP students were expected to outperform MEP students on the construction of relations in their models.

3. Method

3.1. Participants and design

The initial sample consisted of 90 Dutch high-school students from the science track, aged 15–17. However, as 6 students were absent due to illness during one of the sessions, analyses were performed with 84 participants. A review of school curricula and teacher statements showed that the charging of capacitors, which was the topic of inquiry, had not been taught yet in the students' physics classes. A pretest was administered to substantiate that participants were indeed domain novices; class-ranked pretest scores were used to assign students to either the MOP condition ($n = 28$), the MEP condition ($n = 26$), or the control condition ($n = 30$).

3.2. Materials

3.2.1. Inquiry task and learning environments

All participants worked on an inquiry task about the charging of a capacitor. Their assignment was to examine an electrical circuit in which a capacitor was embedded, and create a computer model that mirrors the capacitor's charging behaviour. Participants performed this task within a modified stand-alone version of the Co-Lab learning environment (van Joolingen et al., 2005) that stored all participants' actions in a logfile.

The learning environment housed a *simulation* of an electrical circuit containing a voltage source, two light bulbs, and a capacitor. Through systematic experimentation with this simulation, participants could induce four physics equations: (1) Ohms law, (2) the junction rule of Kirchoff's law, (3) the loop rule of Kirchoff' law, and (4) the behaviour of capacitors.

The learning environment also contained a *model editor* tool that enabled participants to represent their knowledge of the four physics equations in an executable computer model. As can be seen from Fig. 1, such models have a graphical structure that consists of variables and relations. Variables are the constituent elements of a model and can be of three different types: variables that do not change over time (i.e., constants), variables that specify the integration of other variables (i.e., auxiliaries), and variables that accumulate over time (i.e., stocks). Relations define how two or more variables interact. Each relation is visualized by an arrow connector to indicate the causal link between model elements, and specified by a quantitative formula to indicate the exact nature of this relationship. The model editor also enabled participants to test their understanding by running the model and analysing its output through the *table* and *graph* tool. These tools further allowed students to compare model and simulation output in a single window. Students could use the results of this comparison to adjust or fine-tune their model and thus build an increasingly elaborate understanding of a charging capacitor.

An embedded *help file* tool contained the assignment and offered explanations of the operation of the tools in the learning environment. The help files also informed participants about the specifics of their condition by indicating whether and how their session was divided into phases (see Section 3.2.2). The help files contained no domain information on electrical circuits and capacitors as this knowledge should be inferred from interacting with the simulation.

3.2.2. Variants of the learning environment for the different conditions

All conditions used the same instructional content (i.e., electrical circuits), but differed with regard to the scaffolding mechanisms (see Fig. 2). Participants in the control condition worked with the standard configuration of the environment (as described above) and thus received no scaffolding.

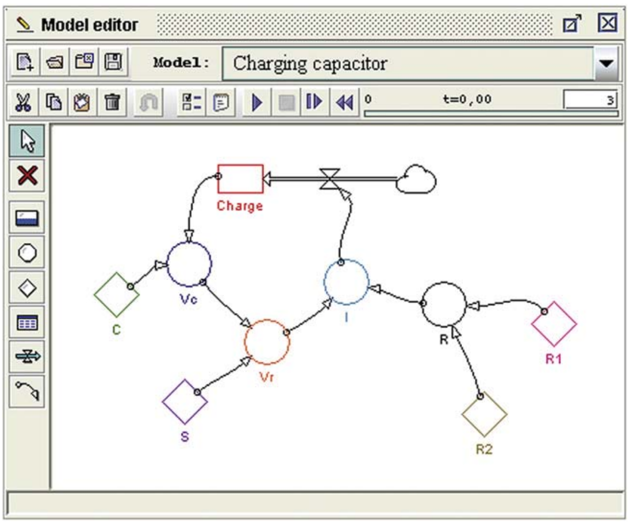

Fig. 1. Screen capture of the model editor tool, which shows the reference model students had to build from their prior knowledge and/or insights gained through experimenting with the simulation. The reference model depicts the charging of a capacitor. The stock element (Charge), which changes over time, is influenced by four constant elements (capacitance [C], power source [S], and two resistances [R1 and R2]) and four mediating auxiliary elements (potential difference across the capacitor [Vc], potential difference across the resistances [Vr], current [I], and resistance total [R]).

Participants in the *model order progression* (MOP) condition received a full-complex version of the simulation, and were asked to induce and build increasingly specific models. Specificity pertained to the relations in the model and progressed in three phases from identifying a relation to quantitatively specifying that relation (cf. Lazonder, Wilhelm, & van Lieburg, 2009; Mulder et al., 2010). In Phase 1, students just had to indicate the model elements (variables) and which ones affected which others (relationships) – but not *how* they affected them. In Phase 2, students had to provide a qualitative specification of each relationship so as to indicate the general direction of effect (e.g., if resistance increases, then current decreases). In Phase 3, students had to specify each relationship quantitatively in the form of an actual equation (e.g., $I = V/R$).

As the first MOP phase had students identify relationships without specifying them, it was technically impossible for the model editor to execute these models. Model runs in this phase therefore activated a software agent that assessed the correctness of the students' model by comparing its elements and relations to a reference model. Results were presented in a bar chart tool with stacked columns that showed the number of correct, incorrect, and incomplete elements as well as the number of correct and incorrect relations. The bar chart tool was available in Phase 1 only. However, students could still add and delete elements and relations during subsequent phases.

In the *model elaboration progression* (MEP) condition, the complexity of the simulation was gradually increased by adding components to the electrical circuit. The simulation in Phase 1 contained a circuit with a voltage source and one light bulb, enabling discovery of Ohm's law. A second light bulb was added to the electrical circuit in the simulation in Phase 2, now introducing the junction rule of Kirchoff's law. The capacitor was added to the simulation in Phase 3, introducing both the loop rule of Kirchoff's law and the behaviour of capacitors. Participants had to induce and build a quantitative model of the circuit in each simulation. Over phases, participants could extend their model to incorporate the new elements. Both the bar chart tool and the possibility to engage in qualitative modelling were disabled in this condition.

3.2.3. Pretest

A pretest consisting of eight open-ended questions assessed participants' prior knowledge of electrical circuits. Four questions addressed the meaning of key domain concepts (i.e., voltage source, resistance, capacitor, capacitance), the other four items addressed the knowledge about the charging of a capacitor in an electrical circuit (i.e., Ohms law, Kirchoff's law (including its two rules: the junction rule and the loop rule), and the behaviour of capacitors). As performance on the test was expected to be low, three simple filler items on the interpretation of numerical data were added to sustain students' motivation during the test. These filler items were left out of the analysis. A rubric was developed to score participants' answers to the eight questions, and one point was allocated to each correct response. Two raters used this rubric

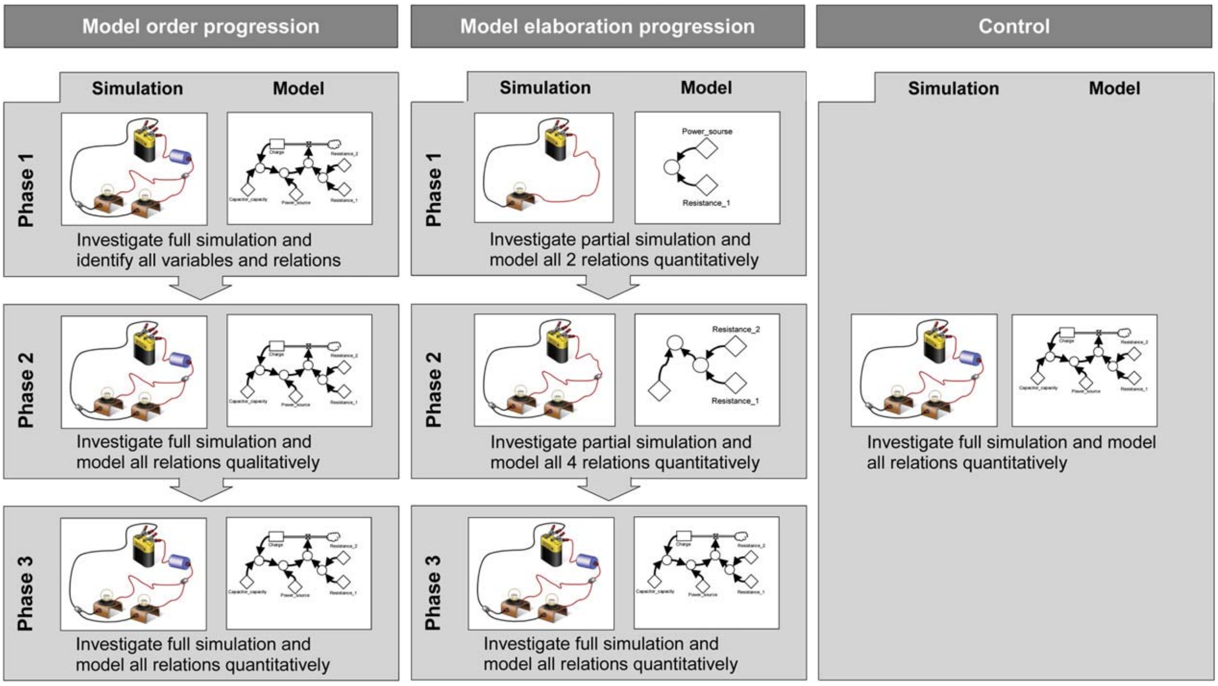

Fig. 2. Schematic overview of the three experimental conditions. Students in the model order progression (MOP) condition investigated a full-complex simulation in all three phases and had to induce and build increasingly specific models. Students in the model elaboration progression (MEP) condition were given an increasingly elaborate simulation in each phase which they had to model quantitatively. Students in the control condition worked with a full-complex simulation and had to induce and build a full quantitative model.

to score a randomly selected set of 24 pretests; inter-rater reliability was .89 (Cohen's κ).

3.3. Procedure

All participants engaged in two sessions: a 50-min introduction and a 100-min experimental session. The time between sessions was one week maximum. To control for differences of the duration of this break, the allocation to condition occurred within each class, so that participants with different inter-session time gaps were equally spread across conditions. During the introductory session, participants first filled out the pretest, then received a guided tour of the Co-Lab learning environment, and finally completed a brief tutorial that familiarized them with the system dynamics modelling language and the operation of the modelling tool.

The experimental session started with a brief reminder that some participants would work in a learning environment where the assignment was split into phases (i.e., the model progression conditions), whereas others would receive a non-divided assignment (i.e., the control condition). Participants were instructed to open the help file tool upon entering the learning environment to find out the specifics of the condition they were assigned to. Consequently, participants were only aware of the details of their own condition. Participants in both model progression conditions were further told that they were free to progress through the phases, but could not return to a previous phase. Towards this end, phase changes were password protected, and participants had to ask the experimenter to unlock the next level. The experimenter did so only if a participant was certain about the phase change and had saved his/her model. After these instructions participants started the assignment. They worked individually and could ask the experimenter for technical assistance only. Participants could stop ahead of time if they had completed the assignment.

3.4. Coding and scoring

All data were assessed from the logfiles. Variables under investigation were time on task and performance success. Time on task concerned the duration of the experimental session.

Performance success scores were assessed from participants' final models. For both model progression conditions, intermediate performance success scores were assessed at the end of each phase. A model structure score was calculated in accordance with Manlove, Lazonder, and de Jong's (2006) model coding rubric. This score represented the number of correct variables and relations in the models. 'Correct' was judged from the reference model shown in Fig. 1. One point was awarded for each correctly named element; an additional point was given if that variable was of the correct type (i.e., constant, auxiliary, or stock). Concerning relations, one point was awarded for each correct link between two variables and one point was awarded for the direction. The maximum model structure score was 38. A previous study (Mulder et al., 2010) found inter-rater reliability estimates of .74 (variables) and .92 (relations) (Cohen's κ).

As the model structure score leaves the quantitative aspects of the model unaddressed, a complementary final model content score was calculated. This score represented participants' understanding of the physics equations that govern the behaviour of a charging capacitor (i.e., Ohms Law: $I = V/R$; resistances connected in parallel: $1/R_t = 1/R_1 + 1/R_2$; the potential difference in the circuit depends on the power source and the potential difference across the capacitor: $\Delta V = V_s - V_c$; and the relationship between the potential difference across the capacitor and the amount of charge that gathers on the capacitor: $C = Q/V_c$). In a correct, fully-specified model these components are correctly integrated as represented in Equation (1):

$$(dQ/dt) = (V_s - Q/C) * (1/R_1 + 1/R_2) \tag{1}$$

One point was awarded for each correctly specified part, leading to a four-point maximum score. A prior study (Mulder et al., 2010) found the inter-rater reliability to be 1.0 (Cohen's κ).

4. Results

Preliminary analyses were performed to check whether the matching of participants had lead to comparable levels of prior knowledge across conditions and to validate the selection of participants. The entire-sample mean pretest score was 1.35 (SD = 1.10), which was deemed sufficiently low to assume that participants can be considered domain novices. The mean pretest scores for each condition are presented in Table 1. Univariate analysis of variance (ANOVA) showed that there were no significant differences in prior knowledge among the three experimental conditions, $F(2, 81) = .24$, $p = .787$. The time on task scores from Table 1 further show that, on average, participants in each condition spent over 90 min working on the assignment. Univariate ANOVA showed that the minor cross-condition differences in time were not statistically significant, $F(2, 81) = 1.34$, $p = .268$.

Performance success was assessed from the participants' final models (see Table 1). A distinction was made between model content and model structure scores. As the model content scores failed to meet the normality assumption, this data was analyzed by a non-parametric Kruskal–Wallis test. Results showed a significant effect for experimental condition on the model content scores of participants' final models, $H(2) = 13.16$, $p = .001$. Post hoc comparisons, using Mann–Whitney U tests with Bonferroni correction ($\alpha = .0167$), addressed the hypotheses. Contrary to expectations, no differences were found in model content scores between the MOP condition and either the MEP condition, $U = 114$, or the control condition, $U = 174$. However, in line with Hypothesis 1, comparison among the latter two conditions revealed a significant difference in favour of the MEP condition, $U = 298$, $r = .33$. In interpreting these results, it should be noted that few MOP participants ($n = 12$) reached the third phase where they could specify their model quantitatively. The remaining 16 MOP participants obtained a model content score of zero, which, as will be discussed below, can often be explained by slow progressing through phases.

Participants' model structure scores were analyzed by MANOVA with both model structure aspects (i.e., variables and relations) as dependent variables. Using Pillai's trace, this analysis produced a significant effect for experimental condition, $V = .21$, $F(4, 162) = 4.74$, $p = .001$. Subsequent univariate ANOVAs indicated that model progression has no effect on the number of correct variables in the students' models, $F(2, 81) = .85$, $p = .431$, but does enhance the quality of the relations between these variables, $F(2, 81) = 9.53$, $p < .001$. Helmert planned contrasts revealed that—in line with Hypothesis 1—the model progression conditions combined had significantly higher scores for relations than the control condition, $t(81) = 2.45$, $p = .006$, $r = .26$, and that—in line with Hypothesis 2—the MOP condition outperformed the MEP condition on this measure, $t(81) = 3.56$, $p = .001$, $r = .37$.

Performance success within both model progression conditions was assessed at three points in time. Table 2 reports descriptive results for each assessment, indicating how the quality of the participants' models developed through time. For statistical analysis of this data it needs to be taken into account that not all participants progressed through all phases. Based on their progression through phases they were classified as either double phase changers (i.e., participants who worked in all three phases, $n = 29$), single phase changers (i.e., participants who worked in Phase 1 and 2, $n = 17$) and no phase changers (i.e., participants who only worked in Phase 1, $n = 8$). As previously mentioned, model quality could not be assessed from the model content score for most of the MOP participants; therefore for the remainder of this results section the model structure score will be the only measure of performance success.

Prior to analysing how the quality of the participants' models developed over time and across conditions, it was examined whether the subset of participants who progressed to subsequent phases were representative of the entire sample in their experimental condition. Logistic regression analyses (using the Enter method) were conducted to determine whether transition to the second and third phase depended on the type of model progression and performance success. The models generated by the logistic regression predicting the phase changes are reported in Table 3. This data indicates that phase change was related to the type of model progression but

Table 1
Summary of participants' performance.

	MOP ($n = 28$)		MEP ($n = 26$)		Control ($n = 30$)	
	M	SD	M	SD	M	SD
Pretest score	1.39	1.20	1.42	1.18	1.23	1.01
Time on task (min)	98.41	5.82	93.60	13.82	95.58	11.66
Performance success						
Model content score[a,b]	0.00	0.00	0.50	0.91	0.07	0.37
Model structure score (variables)[c]	7.29	2.57	6.58	2.50	6.63	1.63
Model structure score (relations)[d]	6.86	3.79	3.42	3.60	3.17	3.24

[a] Maximum score = 4.
[b] As only 12 MOP participants progressed through all phases, a model content score of 0 can often be explained by slow progressing through phases.
[c] Maximum score = 18.
[d] Maximum score = 20.

Table 2
Mean performance success scores in both model progression conditions by phase.

	MOP			MEP		
	n	M	SD	*n*	M	SD
Model structure score (variables)						
Phase 1	28	6.46	2.50	26	4.08	1.55
Phase 2	26	7.04	2.68	20	5.25	1.92
Phase 3	12	6.92	2.47	17	6.82	2.94
Model structure score (relations)						
Phase 1	28	5.57	3.63	26	1.85	2.13
Phase 2	26	6.04	3.68	20	2.20	2.80
Phase 3	12	6.42	3.90	17	2.71	3.47

not to performance success, indicating that the type of model progression influenced the speed of progression through phases. Furthermore, performance success was not a significant predictor of phase change, suggesting comparable performance success slopes for double, single, and no phase changers. This means that the students who progressed to higher phases performed as well as the students who remained in a lower phase. Consequently, analysis of performance success slopes of the double phase changers is likely to be representative for single phase changers' performance success slopes over Phase 1 and 2, and for the no phase changers' performance success over Phase 1.

A mixed-design MANOVA was performed to analyse how model structure scores for variables and relations evolved over the phases in each condition. Using Pillai's trace, MANOVA showed significant multivariate main effects for the between-subjects factor condition, V = .27, $F(2, 26) = 4.88$, $p = .016$, the within-subject factor phase, V = .49, $F(4, 24) = 5.76$, $p = .002$, and a significant Condition × Phase interaction, V = .34, $F(4, 24) = 3.09$, $p = .035$.

Subsequent univariate ANOVAs with the variables aspect of the model structure score as dependent variable indicated a non-significant main effect for condition, $F(1, 27) = 2.32$, $p = .139$, a significant main effect for phase, $F(2, 54) = 14.74$, $p < .001$, and a significant Condition × Phase interaction, $F(2, 54) = 5.28$, $p = .008$. The latter result indicates that the increase in model quality (variables aspect) over phases differed among MOP and MEP participants. Two post hoc comparisons, with Bonferroni correction ($\alpha = .025$), were performed to break down this interaction. The first comparison showed no significant interaction in scores at the end of Phase 1 and 2, $F(1, 27) = .24$, $p = .628$, meaning that both MOP and MEP participants' variables aspect of the model structure score slightly increased during Phase 2. From Table 2 it can be seen that scores in the MEP condition also increased during Phase 3 whereas scores in the MOP condition remained relatively constant. The second post hoc comparison revealed that this interaction was statistically significant, $F(1, 27) = 7.71$, $p = .010$. Upon interpreting these results, it needs to be taken into account that participants in the MEP condition started with a partial simulation that progressed over phases to the full-complex simulation in Phase 3. Therefore, an increase in performance success is inherent to the experimental manipulation in the MEP condition, whereas in the MOP condition it is not.

Univariate ANOVAs with the relations aspect of the model structure score as dependent variable indicated both main effects to be significant (condition: $F(1, 27) = 9.86$, $p = .004$; phases: $F(1.69, 45.56) = 5.54$, $p = .010$), whereas their interaction was not, $F(1.69, 45.56) = 0.14$, $p = .836$ (As the sphericity assumption was violated, the Huynh-Feldt corrected degrees of freedom are reported). This indicates that the relation aspect of the model structure score in the MOP condition was generally higher than in the MEP condition. Furthermore, as shown in Table 2, there was a gradual increase in the relations' aspect of the model structure score over phases. Planned contrasts revealed that this increase was not significant during Phase 2, $F(1, 27) = .47$, $p = .497$, whereas it was significant during Phase 3, $F(1, 27) = 7.12$, $p = .013$. It is interesting to see that, although an increase is inherent to the manipulation in the MEP condition, the model structure score regarding the relations in the model does not show a different slope for the MEP condition compared to the MOP condition (as was the case for the variables aspect of model structure score).

5. Discussion

This study investigated the effects of model progression on students' performance during an inquiry learning task. Model progression in general was predicted to lead to higher performance success (Hypothesis 1). Furthermore, as model order progression was assumed to be more in keeping with domain novices' learning needs, participants in the MOP condition were expected to outperform those from the MEP condition on the construction of relations in their models (Hypothesis 2). Both predictions were generally supported by the results.

Table 3
Logistic regression-analyses (Enter method) testing the dependence of progression speed on condition and performance success.

Predictor	B	SE	Wald	d.f.	sig.	Exp(B)
Dropout phase 2[a]						
Condition	−2.37	1.24	3.64	1	0.056	0.09
Model structure (variables aspect) phase 1	−0.05	0.26	0.04	1	0.835	0.95
Model structure (relations aspect) phase 1	−0.18	0.17	1.12	1	0.289	0.83
Constant	4.18	1.82	5.28	1	0.022	65.09
Dropout phase 3[b]						
Condition	2.01	0.91	4.86	1	0.028	7.46
Model structure (variables aspect) phase 1	0.12	0.27	0.21	1	0.649	1.13
Model structure (relations aspect) phase 1	0.44	0.28	2.51	1	0.113	1.55
Model structure (variables aspect) phase 2	−0.03	0.20	0.02	1	0.882	0.97
Model structure (relations aspect) phase 2	−0.50	0.27	3.46	2	0.063	0.61
Constant	−0.12	1.19	0.01	1	0.919	0.89

[a] $R^2 = .12$ (Hosmer & Lemeshow), .09 (Cox & Snell), .15 (Nagelkerke). Model $\chi^2(3) = 4.86$, $p = .182$.
[b] $R^2 = .21$ (Hosmer & Lemeshow), .24 (Cox & Snell), .33 (Nagelkerke). Model $\chi^2(5) = 12.63$, $p = .027$.

Evidence for Hypothesis 1 comes from the comparison of the two model progression conditions together with the control condition. Participants from both model progression conditions created more comprehensive models – as indicated by their model structure scores – than their control counterparts. Participants' model content scores further indicate that students in the MEP condition created more sophisticated models than students from the control condition. The predicted superiority of the MOP condition on this measure could not be shown, which is likely due to their slow progressing through phases.

Differences in performance success were also examined between both model progression conditions. Consistent with Hypothesis 2, students from the MOP condition had higher model structure scores than students in the MEP condition. Comparison of their final models indicated that MOP and MEP students were equally proficient in identifying which elements are relevant to their models (i.e., voltage source, light bulbs, and capacitor), whereas MOP participants more accurately modelled the relations between those elements. However, the predicted superiority of the MOP condition could not be shown on the model content score.

Existing research on the effectiveness of model progression paint a mixed picture, and the present findings could help explain why this is so. De Jong et al. (1999) previously proposed two conditions for model progression to be effective: a high level of task complexity and low levels of prior domain knowledge. The present study suggests that the type of model progression constitutes a third condition: in the physics domain of electrical circuits, model order progression was found to be more effective than model elaboration progression. The inconsistent results from prior research might therefore be attributable to the application of other, less effective types of model progression such as model elaboration progression (Alessi, 1995; de Jong et al., 1999; Swaak et al., 1998; Quinn & Alessi, 1994).

Even though all cited studies attempted to scaffold students on a science task, there is some evidence that the effectiveness of model order progression extends to different domains. In a recent study, Slof, Erkens, Kirschner, & Jaspers (2010) successfully applied model order progression to a business-economics task. They distinguished three models (a conceptual, causal, and simulation model) that essentially resemble the current study's model order progression phases. Students who consecutively received the three models performed better than students who only worked with one of these models throughout the entire session.

But do students who perform better also learn more? A knowledge posttest might have answered this question, but could not be included in the present study for practical reasons. Yet theoretical and empirical evidence suggests that the performance measures (i.e., model quality scores) that assessed the instructional effects of model progression are indicative of the knowledge students acquired during the experiment. Our students built a system dynamics model that was assumed to represent their knowledge of a charging capacitor. This assumption is based on constructionism, an instructional paradigm in which learning is considered synonymous to the knowledge construction that takes place when learners are engaged in building objects (Kafai & Resnick, 1996). Research has confirmed that the construction of models is associated with cognitive learning (e.g., van Borkulo, 2009), and that the quality of students' models is associated with their reasoning processes (Sins et al., 2005). It thus seems plausible that the superior performance of the MOP participants mirrors higher knowledge acquisition. Still, future research is needed to validate this claim, preferably through an independent measure of learning to supplement performance success measures.

Performance in the present study, although significant, was quite modest. The average model structure scores indicate that even the models created by participants in the MOP condition only partially reflected the contents of the domain. Future research might assess students additional support needs by addressing learning behaviour. A closer look at the quality of students' simulation and model experiments could provide more detailed information as to why elements and relations in the model are correct or not. From the present study it seems plausible that, given that relatively few participants reached the final progression phase, time on task was too short for students to create a full-fledged model. Task performance could accordingly be enhanced by either increasing time on task, or by promoting participants efficiency during the task.

Prior attempts to increase efficiency have tried to accompany model progression with assignments (de Jong et al., 1999; Swaak et al., 1998). These efforts turned out to be unsuccessful; future research should either continue along these lines or explore the effect of other types of additional support. One example would be to embed domain information in the learning environment. Lazonder, Hagemans, and de Jong (2010) found that this type of content support significantly enhances students' inquiry learning performance. Offering designated pieces of domain information in each model progression phase might accordingly improve the efficiency of students' modelling performance.

Efficiency could also be increased by fine-tuning the way model order progression is implemented. Model progression in this study followed the students' learning pace: if learners comprehended a phase, they could progress to the next phase. However, students progressed to consecutive phases with suboptimal models, suggesting that they progressed without full comprehension of the previous phase. This might have compromised the effectiveness of model progression which aims to keep the learning environment manageable by not introducing too many ideas at the same time (Swaak et al., 1998). For students who progressed to phases with suboptimal knowledge, the new phase is unlikely to be manageable. This might also have accounted for the large number of students who never progressed beyond Phase 2 in this study. As such, the instructional effectiveness of model order progression could be further enhanced by prohibiting phase change until a minimal comprehension level is reached. This could be implemented by a restrictive software agent that functions as a gatekeeper at the phase change points based on a model quality benchmark.

Restricting phase-changing nevertheless appears (and probably is) a counterintuitive way to help students progress through all phases. An alternative approach might be to further reduce phase change restrictions. Based on Klahr and Dunbar (1988), inquiry learning is often defined as consisting of three iterative processes: hypothesizing, experimenting, and evaluating evidence. The way students perform these processes is assumed to depend on their knowledge of the task at hand. Model order progression as implemented in this study only enabled students to iterate these processes within each phase. This suggests that model order progression might conflict with the iterative nature of the inquiry learning process; directions confirming this conflict can be found in the results. As model structure scores in the MOP condition were found to increase even in Phase 2 and 3, students apparently generated new or adjusted hypotheses on a Phase 1 level in subsequent phases. Therefore, an adjustment to model order progression, to the extent that students can freely navigate through the order dimension both forwards and backwards (i.e., iterative model order progression), might better suit the iterative nature of the inquiry learning process.

To conclude, this study points to the idea that model progression can foster performance, and that learners benefit most from model order progression. Future research is needed to investigate how model order progression can be further optimized, and two alternative approaches were proposed. One is to introduce a restrictive software agent that functions as gatekeeper at the phase change points, the second alternative is to be less restrictive and allow students to wander across phases in any order they see fit.

Acknowledgements

The authors gratefully acknowledge Anjo Anjewierden, Lars Bollen, and Jakob Sikken for their contributions to the software that was used in this study.

References

Alessi, S. (1995). Dynamic versus static fidelity in a procedural simulation. Paper presented at the Annual Meeting of the American Educational Research Association, San Francisco.

Carroll, J. M., & Carrithers, C. (1984). Blocking learners error states in a training-wheels system. *Human Factors, 26*, 377–389.

Dean, D., & Kuhn, D. (2007). Direct instruction vs. discovery: the long view. *Science Education, 91*, 384–397. doi:10.1002/sce.20194.

de Jong, T. (2005). The guided discovery principle in multimedia learning. In R. E. Mayer (Ed.), *Cambridge handbook of multimedia learning* (pp. 215–229). Cambridge: Cambridge University Press. doi:10.2277/0521838738.

de Jong, T., Martin, E., Zamarro, J. M., Esquembre, F., Swaak, J., & van Joolingen, W. R. (1999). The integration of computer simulation and learning support: an example from the physics domain of collisions. *Journal of Research in Science Teaching, 36*, 597–615. doi:10.1002/(SICI)1098-2736(199905)36, 5<597::AID-TEA6>3.0.CO;2–6.

de Jong, T., & van Joolingen, W. R. (1998). Scientific discovery learning with computer simulations of conceptual domains. *Review of Educational Research, 68*, 179–201, Retrieved from. http://www.jstor.org/stable/1170753.

Gagné, R. M. (1977). *The conditions of learning* (3rd ed.). New York: Holt, Rinehart & Winston.

Hogan, K., & Thomas, D. (2001). Cognitive comparisons of students' systems modeling in ecology. *Journal of Science Education and Technology, 10*, 319–345. doi:10.1023/A:1012243102249.

Kafai, Y. B., & Resnick, M. (1996). *Constructionism in practice: Designing, thinking, and learning in a digital world*. Mawhaw, NJ: Erlbaum.

Klahr, D., & Dunbar, K. (1988). Dual space search during scientific reasoning. *Cognitive Science, 12*, 1–48. doi:10.1207/s15516709cog1201_1.

Klahr, D., & Nigam, M. (2004). The equivalence of learning paths in early science instruction: effects of direct instruction and discovery learning. *Psychological Science, 15*, 661–667. doi:10.1111/j.0956-7976.2004.00737.x.

Krajcik, J., Blumenfeld, P. C., Marx, R. W., Bass, K. M., Fredericks, J., & Soloway, E. (2000). Instructional, curricular, and technological supports for inquiry in science classrooms. In J. Minstrell, & E. H. van Zee (Eds.), *Inquiring into inquiry learning and teaching in science*. Washington, DC: AAAS.

Lazonder, A. W., & van der Meij, H. (1995). Error-information in tutorial documentation: supporting users' errors to facilitate initial skill learning. *International Journal of Human-Computer Studies, 42*, 185–206. doi:10.1006/ijhc.1995.1009.

Lazonder, A. W., Wilhelm, P., & van Lieburg, E. (2009). Unraveling the influence of domain knowledge during simulation-based inquiry learning. *Instructional Science, 37*, 437–451. doi:10.1007/s11251-008-9055-8.

Lazonder, A. W., Hagemans, M. G., & de Jong, T. (2010). Offering and discovering domain information in simulation-based inquiry learning. *Learning and Instruction, 20*, 511–520. doi:10.1016/j.learninstruc.2009.08.001.

Löhner, S., van Joolingen, W. R., & Savelsbergh, E. R. (2003). The effect of external representation on constructing computer models of complex phenomena. *Instructional Science, 31*, 395–418. doi:10.1023/A:1025746813683.

Manlove, S., Lazonder, A. W., & de Jong, T. (2006). Regulative support for collaborative scientific inquiry learning. *Journal of Computer Assisted Learning, 22*, 87–98. doi:10.1111/j.1365-2729.2006.00162.x.

Mayer, R. E. (2004). Should there be a three-strikes rule against pure discovery learning? *American Psychologist, 59*, 14–19. doi:10.1037/0003-066X.59.1.14.

Mulder, Y. G., Lazonder, A. W., & de Jong, T. (2010). Finding out how they find it out: an empirical analysis of inquiry learners' need for support. *International Journal of Science Education, 32*, 2033–2053. doi:10.1080/09500690903289993.

Ogborn, J. (1998). Cognitive development and qualitative modelling. *Journal of Computer Assisted Learning, 14*, 292–307. doi:10.1046/j.1365-2729.1998.144069.x.

Quinn, J., & Alessi, S. (1994). The effects of simulation complexity and hypothesis-generation strategy on learning. *Journal of Research on Computing in Education, 27*, 75–91.

Reigeluth, C., & Stein, F. S. (1983). The elaboration theory of instruction. In C. Reigeluth (Ed.), *Instructional design theories and models: An overview of their current status*. Hillsdale, NJ: Erlbaum.

Rieber, L. P., & Parmley, M. W. (1995). To teach or not to teach? comparing the use of computer-based simulations in deductive versus inductive approaches to learning with adults in science. *Journal of Educational Computing Research, 13*, 359–374. doi:10.2190/M8VX-68BC-1TU2-B6DV.

Sins, P. H. M., Savelsbergh, E. R., & van Joolingen, W. R. (2005). The difficult process of scientific modeling: an analysis of novices' reasoning during computer-based modeling. *International Journal of Science Education, 14*, 1695–1721. doi:10.1080/09500690500206408.

Slof, B., Erkens, G., Kirschner, P. A., & Jaspers, J. G. M. (2010). Design and effects of representational scripting on group performance. *Educational Technology Research and Development, 58*, 589–608. doi:10.1007/s11423-010-9148-3.

Spannagel, C., Girwids, R., Löthe, H., Zendler, A., & Schroeder, U. (2008). Animated demonstrations and training wheels interfaces in a complex learning environment. *Interacting with Computers, 20*, 97–111. doi:10.1016/j.intcom.2007.08.002.

Stratford, S. J., Krajcik, J., & Soloway, E. (1998). Secondary students' dynamic modeling processes: analyzing, reasoning about, synthesizing, and testing models of stream ecosystems. *Journal of Science Education and Technology, 7*, 215. doi:10.1023/A:1021840407112.

Swaak, J., van Joolingen, W. R., & de Jong, T. (1998). Supporting simulation-based learning: the effects of model progression and assignments on definitional and intuitive knowledge. *Learning and Instruction, 8*, 235–252. doi:10.1016/s0959-4752(98)00018-8.

van Borkulo, S. P. (2009). The assessment of learning outcomes of computer modelling in secondary science education. Unpublished doctoral dissertation, University of Twente, Enschede.

van Joolingen, W. R., de Jong, T., Lazonder, A. W., Savelsbergh, E. R., & Manlove, S. (2005). Co-Lab: research and development of an online learning environment for collaborative scientific discovery learning. *Computers in Human Behavior, 21*, 671–688. doi:10.1016/j.chb.2004.10.039.

Veenman, M. V. J., & Elshout, J. J. (1995). Differential effects of instructional support on learning in simulation environments. *Instructional Science, 22*, 363–383. doi:10.1007/BF00891961.

White, B. Y., & Frederiksen, J. R. (1990). Causal model progressions as a foundation for intelligent learning environments. *Artificial Intelligence, 42*, 99–157. doi:10.1016/0004-3702(90)90095-h.

White, B. Y., Shimoda, T. A., & Frederiksen, J. R. (1999). Enabling students to construct theories of collaborative inquiry and reflective learning: computer support for metacognitive development. *International Journal of Artificial Intelligence in Education, 10*, 151–182.

Zacharia, Z. C., & Olympiou, G. (2011). Physical versus virtual manipulative experimentation in physics learning. Learning and Instruction, 21, 317–331, doi: 10.1016/j.learninstruc.2010.03.001.