Assessing learner-constructed conceptual models and simulations of dynamic systems*

Bert Bredeweg[1] and Jochem Liem[1] and Christiana Nicolaou[2]

[1] Informatics Institute, University of Amsterdam, The Netherlands
B.Bredeweg@uva.nl, Jochem.Liem@gmail.com
[2] Department of Educational Studies, University of Cyprus, Cyprus
chr.nic@ucy.ac.cy

Abstract. Learning by conceptual modeling is seeing uptake in secondary and higher education. However, assessment of conceptual models is underdeveloped. This paper proposes an assessment method for conceptual models. The method is based on a metric that includes 36 types of issues that diminish model features. The approach was applied by educators and positively evaluated. It was considered useful and the derived grades corresponded with their intuitions about the models quality.

Keywords: Assessment, Conceptual modeling and simulation, Dynamic systems, Systems thinking

1 Introduction

Acquiring knowledge by constructing and using models is seeing uptake in secondary and higher education [5]. Recently, the approach is applied in a novel way using *conceptual models* and accompanying tools, which allow modelers to develop and simulate conceptual representations of dynamic systems [9, 2].

To implement modeling in classroom practice, formative and summative assessment techniques [7] for evaluating learner-constructed models are indispensable [19]. Assessment is one of the four vital parameters for science education, together with curriculum, instruction and professional development [17]. However, the assessment of conceptual models is underdeveloped, hampering its usage [4]. This means that there is a lack of criteria of what constitutes a good conceptual model. Consequently, it is difficult to give feedback to learners regarding the models they create. The problem is even more pressing when learning is self-regulated, and (groups of) learners develop their own unique models with different viewpoints, conceptualisations, and levels of abstraction. Comparison between learner-constructed models, and even comparison with a norm model, becomes impractical and inadequate for assessment.

This paper focusses on how assessment of *conceptual* models can be performed. The central idea is that learner-constructed conceptual models are rich

* Research co-funded by EU FP7, Project no. 231526, http://www.dynalearn.eu.

representations, and as such provide evidence of learning. Particularly, the number of correctly modeled ingredients compared to the total number of model ingredients (determined through a catalogue of modeling suboptimalities) can function as a measure of the modeling competence of the learner. This evidence can be identified, enumerated and scored by an assessment method and as such provide the basis for feedback, both formative and summative, and for learners and teachers. Hence, the question guiding the presented research is: What are the main components of an assessment method which can successfully evaluate diverse and different learner-constructed conceptual models?

2 Educational context and relevance

A scientific model is a construct that represents a system, and that consists of a set of objects and their properties, and a number of law statements indicating the behaviors of these objects in terms of their properties [3]. A *conceptual model* is a special kind of model that represents the referents in the domain through particular concepts as distinguished by the modeling language. For instance, it represents an explicit conceptual account of the physical structural and the behavioral features of the system under study, as well as the network of causal relationships underlying the behavior of the systems [10]. *Modeling competence* refers to the ability to construct and improve models [6].

Computer modeling is widely advocated as a way to offer students a deeper understanding of (complex) systems [14]. Consequently, the need for learners to master modeling competencies, e.g. being able to perform proper cause-effect reasoning. However, acquiring this competence is not so easily accomplished. Modeling is complex and both teachers and learners need to be well supported in order to successfully engage in modeling activities [18].

Learning by modeling is a process of engaging learners in (co)constructing models to gain understanding of systems. It is intrinsically related to the constructivist approach to teaching and learning, which is based on the idea that learners, through the use of the appropriate tools, construct their knowledge through building artifacts, here conceptual models. These artifacts encompass evidence of knowledge and skills on behalf of the learner and as such are rich sources of information of their modeling competence.

A model assessment instrument could thus provide valuable support for all stakeholders. However, well-suited methods for assessing conceptual models are sparse [13]. Some of the existing approaches use norm models [12, 18]. That is, the learner-constructed model is compared to a norm model and then scored. However, such approaches do not provide tools that systematically address deviations in learner-constructed models. Deviations that sometimes are erroneous but often also valuable variations on the norm. Moreover, in the context of self-directed learning activities, learners vary on topics, levels of granularity, perspectives and assumptions taken, etc., leading to a significant yet natural variation in the models constructed, which makes the a priory construction of norm models impractical (if not impossible).

Some other approaches use open-ended techniques addressing the model as a whole and evaluate paper-based models (drawings) [1]. The open-ended methods score the models on very general features, such as comparison and abstraction, while drawings models are not dynamic by nature. Both are limited evaluation procedures by design. Important details may be overlooked by the assessor and the scoring may end up being based on irrelevant or incorrect evidence.

In summary, assessment of learner-constructed models is important, yet usable methods are sparse. This hampers the use of modeling as an educational instrument. The work presented in this paper addresses the problem, particularly concerning the assessment of conceptual models.

3 Conceptual models and assessment needs

Our research addresses science education and particularly the challenge of making *learning by modeling* common practice in secondary education. We focus on conceptual models (as opposed to numerical) because they allow learners to directly interact with vocabulary that is necessary for the conceptual understanding they need to acquire. As a modeling tool we use DynaLearn [2], which has been used successfully in different educational settings as a workbench for learners to develop their understanding of how systems work (cf. [15]). The full workbench provides a sequence of workspaces with increasing complexity that facilitates a stepwise approach toward developing conceptual modeling expertise (for details see [11], Ch. 3 & 4).

3.1 Learner-constructed models - Identifying suboptimalities

Consider the learner-constructed model shown in Fig. 1. It was created during a course on conceptual modeling, within an environmental science bachelor, in which learners worked through a series of modeling assignments using DynaLearn (Learning Space 4, LS4). For the final, inquiry-based assignment, learners were asked to choose a system based on their interest, pose a question about that system and develop a model that answers this question. There were no norm models. The only constraint was that at least two processes causing change in the system were modeled. The learners worked in pairs. Model assessment in the context of such a self-regulated learning activity is quite a challenge.

Let us start by interpreting the domain details shown in the diagram. The model represents a field of quinoa being irrigated using salt water. The amount of water absorbed by the quinoa is determined by the concentration of salt in the roots of the quinoa and the salinity of the earth near the roots of the quinoa. As water is absorbed, the quinoa grows and the yield increases.

There are no major issues with the representation of the physical structure of the system, although *Seeds* (and *Saponin*) can be considered superfluous. Quantities, on the other hand, can be improved. *Volume* of *Salt water* is positively influencing *Soil saturation.* However, causal dependencies of type *I-* or *I+* are used for processes, while in the model the dependency seems to be a proportionality, that is *P-* or *P+*. Hence, this can be considered an incorrect causal relation (issue #20[3]) in the model. However, the model makes more sense if the volume quantity is interpreted as the irrigation process. Therefore, this issue is considered to affect the correctness of the model.

Quantity *Root zone salinity* refers to a mixture of notions including an entity and a quantity. As a result, it can be conceptually decomposed (issue #9). The simplest solution is to rename the quantity *Salinity*. Similarly, *Root salt concentration* can be conceptually decomposed (issue #9). The details in the model representing the physical structure of the system can be augmented by explicitly modeling the roots of the quinoa and indicating that these roots contain salt. This salt entity should have a quantity concentration.

The quantity spaces of *Root salt concentration* and *Root zone salinity* can be improved. There is no clear distinct behavior associated with reaching the landmark *Boundary* (issue #14). Consequently, this value and the value *Higher* can be removed. Secondly, the value *Higher* is vague (issue #15). That is, it is context dependent (higher compared to what?). Renaming this value to whatever happens above the value *Boundary*, or removing the value, would resolve it.

Causality has 2 issues. First, quantity *Root zone salinity* is affected by both a positive influence (from *Water uptake*) and a positive proportionality (from *Soil saturation*). Mixing causality types is incorrect (issue #23). Either a quantity is affected by a process directly, or change propagates to this quantity. In this case the proportionality should be removed. Second, when there is no more water in the soil, there can be no more water uptake (which is modeled using a value correspondence between the magnitudes *Zero* of *Water update* and *Zero* of *Soil saturation*). However, for this to occur, *Water uptake* should decrease as *Soil saturation* decreases. This can be modeled using a positive proportionality from *Soil saturation* to *Water uptake*. This is missing in the model (issue #21).

There are 4 issues with inequalities and correspondences, all resulting in inconsistencies (issue #24) when simulating: value correspondence from *Volume* of *Salt water* to *Soil saturation*, from *Volume* of *Salt water* (derivative) to *Soil saturation* (derivative), and the two correspondences from *Water uptake* to *Growth*.

Finally, simulation has 2 issues (Fig. 2). First, quantity *Soil saturation* has no value (issue #32). Second, quantity *Root salt concentration* has the value *Plus* and is decreasing in state 3, but never reaches *Zero*. This is a so-called *dead-end* (issue #34), caused by an inconsistency.

4 Instrument for assessing conceptual models

Within the conceptual modeling community, there is the belief that *"(...) a conceptual model can only be evaluated against people's (tacit) needs, desires and expectations. Thus the evaluation of conceptual models is by nature a social rather than a technical process, which is inherently subjective and difficult to*

[3] Our method identifies 36 issue types, each with a unique number (see Section 4).

formalise" [16]. We argue that it is possible to elevate model assessment from being a social process to one that is largely standardized and objective.

Our approach is based on the notions of *verification* and *validation*. Verification involves determining whether a product satisfies the conditions defined before development [21]. For a software program, knowledge base, or scientific model, such conditions typically include adhering to the syntactical and semantic requirements of the formalism used to develop the product. By contrast, validation determines whether the product performs adequately for its intended purpose and is satisfactory for the end user. As such, verification can be considered the assessment of internal (or internalized) *quality characteristics*, while validation tests external (purpose-oriented) quality characteristics [16].

Appraising internal quality characteristics (verification) should be an *objective* task. For example, conceptual models that allow for inferences (e.g., simulation) have an internal logic that imposes constraints that can be checked automatically. By contrast, validation is more subjective as a result of being domain and goal dependent. For example, different experts may disagree on whether a model is a *correct* domain representation [20] and can cite different resources to support their case. Here, we focus particularly on verification.

We propose *model features* that attest to the quality of a model (Table 1). These features are categorized into two verification categories. First, *formalism features* apply only to conceptual models developed in formalisms that allow for inferences, such as DynaLearn [2]. These features can be assessed using the internal logic of the formalism (e.g., consistency). The second category, *domain features* apply to conceptual models generally, and rely on the human interpretation of the model to be assessed. For example, the model feature *conformance to ontological commitments* requires that a referent in the domain is represented using the correct model ingredient in the formalism (e.g., biomass should be represented as a quantity). We claim both features can be checked objectively. Algorithms can be created to automatically detect them and suggest corrections.

Next step is to determine which model characteristics can be used to actually measure the quality of a conceptual model in terms of formalism and domain features. Correctness, completeness, and parsimony have been proposed as such quality characteristics (e.g. [20]). *Correctness* indicates that a model is free from errors. *Completeness* means that everything of relevance is included in the model. *Parsimony* implies that the model does not include redundancies. The following sections identify model features that attest to these quality characteristics.

4.1 Formalism-based features

Consistency is a prerequisite for the *correctness* of a conceptual model, and requires that ingredients in the model do not contradict each other (in terms of the possible inferences). For example, a quantity cannot be increasing and decreasing at the same time. *No unassigned variables* is a model feature that is important for the *completeness* of a model. An unassigned variable after reasoning (e.g., simulation) is an indicator that information in the model is missing to allow a particular reasoning step to succeed. *Reasoning relevance* means that

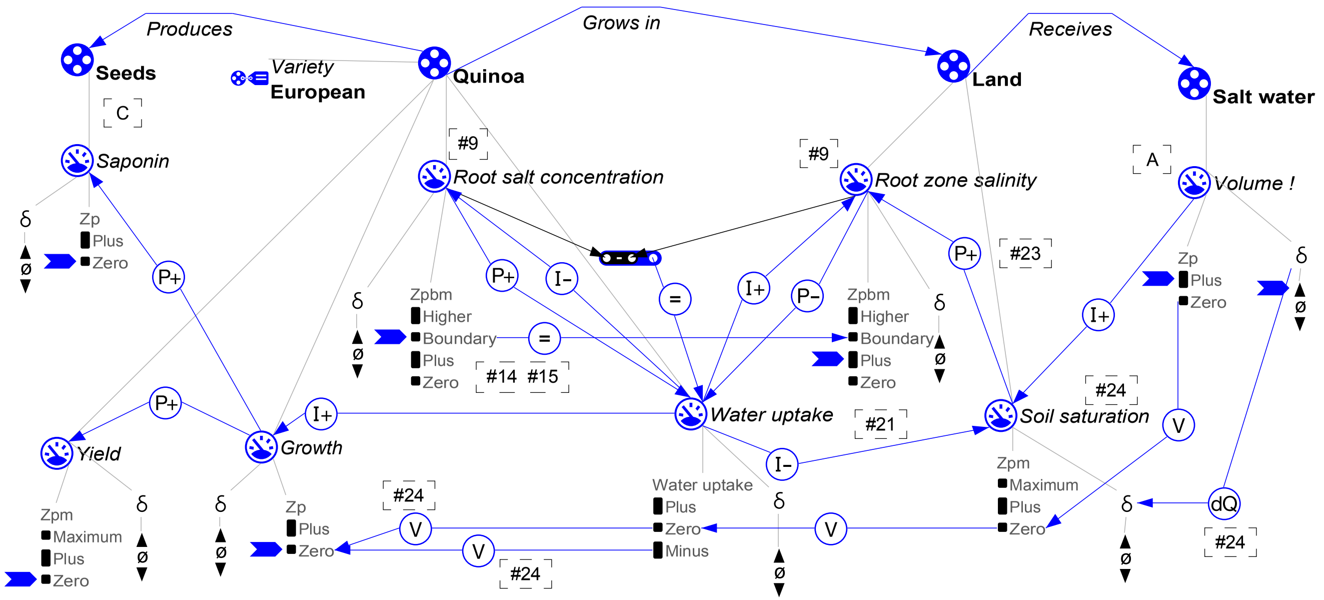

Fig. 1. Learner-constructed DynaLearn [2] model, using learning space 4, modeling the effects of watering quinoa using salt water. The amount of water absorbed (*Water uptake*, WU) by the quinoa is determined by the concentration of salt in the roots (*Root salt concentration*, RSC) of the quinoa and the salinity (*Root zone salinity*, RZS) of the earth near the roots of the quinoa ($RSC - RZS = WU$). As water is absorbed, the quinoa grows (*Water uptake* $I+$ *Growth*) and the yield increases (*Growth* $P+$ *Yield*). Particularly well-modeled is the so-called *equilibrium seeking mechanism* that determines the uptake of water, which consists of two negative feedback loops. The water uptake (if *Water uptake* = *Plus*) decreases the salt concentration in the roots of the quinoa ($I-$), and increases the salinity of the soil surrounding the roots ($I+$). The water uptake decreases as the salt concentration in the root decreases ($P+$), and the water uptake also decreases as the salinity of the soil surrounding the roots increases ($P-$). Note, the layout has been changed by the authors. The model issue numbers (verification) and the validation issues (A: correctness, C: parsimony) are indicated in dashed boxes.

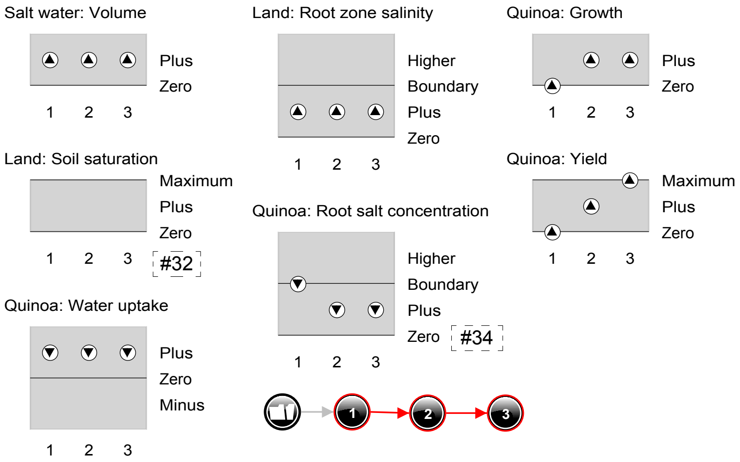

Fig. 2. The state-graph (4 connected circles) and value history (7 squares) of the quinoa model (Fig. 1). The model issues (#32 and #34) are indicated in dashed boxes.

Table 1. Model features that attest to quality characteristics of verification categories.

Verification Category	*Quality Characteristic*	*Model Feature*
Formalism	Correctness	Consistency
	Completeness	No unassigned variables
	Parsimony	Reasoning relevance
Domain representation	Correctness	Conformance to ontological commitments
		Falsifiability
	Completeness	Conceptual decomposition
		No missing representations
	Parsimony	No repetition
		No synonyms

each of the elements in the representation should have a function in terms of the reasoning. If not, the ingredient is superfluous and the model not *parsimonious*. For example, including a quantity without relating it to other quantities.

4.2 Domain representation-based features

Two domain features contribute to the *correctness*. *Conformance to ontological commitments* indicates that referents in the domain are represented using the correct model ingredients. For example, biomass being represented as an entity is an example of a type error. *Falsifiability* is the property of a claim, hypothesis or theory, to be proven false if the 'outcome' cannot be observed in reality. A conceptual model is falsifiable if its simulation results can be shown to be false through comparison with observations. Using vague values, such as 'small' or 'large', is an example of what makes a model unfalsifiable, as it becomes unclear what observations would conflict with the model's simulation results.

Two domain features contribute to *completeness. Conceptual decomposition*, which can be called the *'single concept per model ingredient rule'*, states that model ingredients that represent aggregated concepts should be broken down into multiple ingredients. For example, the use of quantities *Water temperature* and *Air temperature* can be an indicator that *Temperature* is a missing independent model ingredient that should have its own representation. As a guideline, a model ingredient can be considered conceptually decomposed when the represented concept can be found in an encyclopaedia, dictionary or glossary. *No missing representations* means that referents that are important in the domain are represented.[4] For example, given that *Mortality* and *Population size* are represented, there has to be a causal relation between these quantities.

Two domain features contribute to *parsimony. No synonyms* means that a domain concept, such as natality, should only be represented once, and consequently identified using a unique term. Hence, a model in which both *Natality* and *Birth rate* occur breaks this rule. Thesauri can be helpful in determining whether two terms are synonyms. *No repetition* indicates that there are no reoccurring arrangements of related ingredients. Such arrangements should be represented once and reused throughout the model (only at learning space 6 [2]).

4.3 Assessment metric

We have developed a best practice for conceptual modeling in the form of a catalogue of 36 modeling issues, checks to detect them, and modeling actions to ameliorate them (available via ([11] Ch. 5 (p. 99) and App. A.1 (p. 201), section 3.1 gives examples). Each of the issues affects one or more of the model features and thus the overall model quality. The issues are categorized based on whether they affect particular model ingredients, namely (*i*) Structure, (*ii*) Quantities, (*iii*) Quantity spaces, (*iv*) Causality, (*v*) Inequalities and correspondences, (*vi*) Model fragments (only at learning space 6), and (*vii*) Simulation results. For instance, issues #14 en #15 (see Section 3.1) both affect Quantity spaces. Next, we have established a metric that reflects a model's overall quality, based on the best practice (Table 2[5]). The quality metric results in a score between 0 and 100, which, when interpreted as a percentage, can be converted to grades.

How particular assessment categories are weighted is subjective. To minimize the potential for contention about the overall quality metric, we take the position that 50% (or more) of the overall quality measure should be based on objective criteria (hence verification). The other half of the weight is meant for model validation and is equally distributed between how well the model functions as

[4] May contribute to internal and external characteristics. Here the focus is on internal.

[5] Validation is not addressed in this paper. It is assessed using a rubric, see [11].

Table 2. Model assessment categories and weights.

Assessment categories	*Subcategories*	*Weight*
Verification: Model issues (50%)	Structure	10.00%
	Quantities	5.00%
	Quantity spaces	5.00%
	Causality	10.00%
	Inequalities and correspondences	5.00%
	Model fragments	5.00%
	Simulations	10.00%
Validation: adequate domain representation for goal (25%)	Correctness	10.00%
	Completeness	10.00%
	Parsimonious	5.00%
Validation: Communication (25%)	Layout of the model	5.00%
	Documentation	20.00%

a domain representation, and how well the model suites communication. Of course, when deemed appropriate users can change the distribution emphasizing different aspects of conceptual modeling for a particular assignment.

Given the proposed weights, the metric should reflect both those things that have been done correctly and the errors that have been made, as learners need to learn both from their errors, and be motivated by those aspects of modeling that they have done correctly. This results in the following calculation (shown for the Structure ingredients):

$$\text{Structure score} = 100 \times \frac{\#\text{entity} + \text{configuration definitions} - \#\text{structure issues}}{\#\text{entity} + \text{configuration definitions}}$$

When applying the metric, something is counted as an issue if it requires a single correction. As such, repeated reuse of an entity that is not conceptually decomposed counts as a single issue. However, repeated issues of the same type are counted as individual issues. Also, mistakes in smaller models are penalized more heavily, compared to mistakes in larger models. This is done by basing the scores on the ratio between the correctly modeled part and the whole model.

When all the steps needed to grade a model have been taken, the final score can be calculated. For the model in Fig. 1, the results are as follows:

$$
\begin{aligned}
&\text{Structure} && = 100 \times \frac{4+3-0}{4+3} && = 100.0\ (11\%)\\
&\text{Quantities} && = 100 \times \frac{8-2}{8} && = 75.0\ (6\%)\\
&\text{Quantity spaces} && = 100 \times \frac{4-2}{4} && = 50.0\ (6\%)\\
&\text{Causality} && = 100 \times \frac{10+1-2}{10+1} && = 81.8\ (11\%)\\
&\text{Ineq. and corresp.} && = 100 \times \frac{5+2-4}{5+2} && = 42.9\ (5\%)\\
&\text{Simulation} && = 100 \times \frac{3-2}{3} && = 33.3\ (11\%)
\end{aligned}
$$

$$
\begin{aligned}
&\text{Correctness} && = 60\ (10\%)\\
&\text{Completeness} && = 100\ (10\%)\\
&\text{Parsimony} && = 80\ (5\%)\\
&\text{Layout} && = 80\ (5\%)\\
&\text{Documentation} && = 80\ (20\%)
\end{aligned}
$$

The weight for model fragments (not available in LS4) is distributed over the other verification subcategories (except inequalities and correspondences), hence 11 instead 10%, 6 instead of 5%, etc. The causality score is adjusted because a causal relation was missing (issue #21, Section 3.1). Consequently, a causal relation is added to the total number of Causality (10+1). Similarly, mistakes as subtracted: Quantities 8-2 (2x issue #9), Quantity spaces 4-2 (issue #14 & #15), Causality 11-2 (2x issue #23), Ineq. and corresp. 7-4 (4x issue #24), and Simulation (3-2) (issue #32 & #34).

Validation is not discussed here. However, as mentioned before, correctness, completeness, parsimony, layout and documentation are graded using a rubric. The results are shown above, RHS. The final score is 73.3.

5 Evaluating the assessment method

A pilot study was conducted with four evaluators who used the instrument to grade 34 models submitted by the student pairs in the course (two evaluators graded 9 models). The pilot focussed on whether the grades derived using the assessment method are comparable to grades that evaluators proclaim a model deserves. To this end, before having graded any models, the evaluators were asked to *intuitively* grade one set of models assigned to another evaluator[6]. The instruction was to analyse each model for 5 minutes, write down the grade, and proceed to the next model.

The *agreement* between the intuitive and actual grades was calculated. For this the different evaluators are assumed equal, and therefore all assessment method grades are considered of one evaluator (34 grades), and all intuitive grades of another (34 + 10 = 44 grades) (data available via [11] Ch. 5, p. 140). Typical statistical methods for inter-rater agreement (Cohen's kappa and Fleiss' kappa) cannot be used as they require a fixed number of mutually exclusive categories. IntraClass Correlation (ICC) and the Concordance Correlation Coefficient (CCC) can be used. Both were calculated, and both indicate strong agreement of about 0.89 ($r^{ICC} = 0.887$, 99%-confidence interval: $0.765 < r^{ICC}$

[6] One evaluator coincidently graded 2 sets.

< 0.947, $r^{CCC} = 0.885$, 99%-confidence interval: $0.767 < r^{CCC} < 0.945$). Suggesting the method's grades are acceptable.

Evaluators were able to detect model issues easily and only had difficulty in understanding one issue (#9. Ambiguous process rate *quantities*). This suggests that the assessment method is understandable and usable for evaluators. The evaluators required about 45 minutes per model to derive grades. As the model contributed 40% of the final grade, 45 minutes was considered reasonable.

6 Conclusion and discussion

Assessment of learner-constructed models is of great importance for effective development of the modeling competence on behalf of learners, and enabling learning by modeling as common practice in classrooms. Yet, ready to use assessment methods are sparse. We propose an assessment instrument based on a set of model features that attest to the quality of *conceptual* models. The model features address verification, and are categorized as formalism and domain features. The former apply only to conceptual models that allow for inferences, while the latter apply generally. The model features are further categorized as attesting to the quality characteristics correctness, completeness and parsimony.

A pilot study using the assessment method suggests that the derived grades correspond to evaluators' intuition of what a model is worth. The assessment method proved understandable, and the time required to apply it is considered reasonable. A listing of all the issues in a model serves as both an argument why a particular grade was given and as valuable feedback for learners.

As ongoing research we are investigating how the presented approach can be used as a real-time operating instrument, particularly for formative assessment, which requires automated detection of modeling issues. When issues are detected automatically, feedback may also be automated, but can also be left to the teacher. Another interesting future challenge would be to extend the current approach to the assessment of models created by domain experts, such as [8].

References

1. Bamberger, Y. M., Davis, E. A. (2013). Middle-School Science Students Scientific Modelling Performances across Content Areas and within a Learning Progression. International Journal of Science Education, 35(2), 213-238.
2. Bredeweg, B., Liem, J., Beek, W., Linnebank, F., Gracia, J., Lozano, E., Wißner, M., Bühling, R., Salles, P., Noble, R., Zitek, A., Borisova, P., Mioduser, D. (2013). DynaLearn - An Intelligent Learning Environment for Learning Conceptual Knowledge. AI Magazine, 34(4), 46-65.
3. Bunge, M. (1983). Treatise on Basic Philosophy: Volume 5: Epistemology & Methodology I: Exploring the World. Springer. Dordrecht: Reidel.
4. Eurydice: The information network on education in Europe. (2006). Science teaching in schools in Europe. DG for Education and Culture, Brussels.

5. Goberta, J.D., O'Dwyer, L., Horwitz, P., Buckley, B.C., Levy, S.T., Wilensky, U. (2011). Examining the Relationship Between Students' Understanding of the Nature of Models and Conceptual Learning in Biology, Physics, and Chemistry. International Journal of Science Education 33(5), 653-684.
6. Halloun, I. (2007). Mediated Modeling in Science Education. Science and Education, 16, 653-697.
7. Harlen, W., James, M. (1997). Assessment and Learning: differences and relationships between formative and summative assessment. Assessment in Education: Principles, Policy & Practice, 4(3), 365-379.
8. Kansou, K., Nuttle, T., Farnsworth, K., Bredeweg, B. (2013). How plants changed the world: Using qualitative reasoning to explain plant macroevolution's effect on the long-term carbon cycle, Ecological Informatics, 17, 117-142.
9. Leelawong, K., Biswas, G. (2008). Designing Learning by Teaching Agents: The Betty's Brain System. Int. J. of Artificial Intelligence in Education, 18(3), 181-208.
10. Leiba, M., Zuzovsky, R., Mioduser, D., Benayahu, Y., Nachmias, R. (2012). Learning about Ecological Systems by Constructing Qualitative Models with DynaLearn. Interdisciplinary Journal of E-Learning and Learning Objects, 8(1), 165-178.
11. Liem, J.: Supporting Conceptual Modelling of Dynamic Systems: A Knowledge Engineering Perspective on Qualitative Reasoning. University of Amsterdam (2013). *https://jochemliem.files.wordpress.com/2014/01/liem2013-thesisdigital.pdf*
12. Löhner, S., van Joolingen, W. R., Savelsbergh, E. R., van Hout-Wolters, B. (2005). Students Reasoning during Modeling in an Inquiry Learning Environment. Computers in Human Behavior, 21(3), 441.
13. Louca, L., Zacharia, Z., Michael, M., Constantinou, C. (2011). Objects, Entities, Behaviors, and Interactions: A Typology of Student-Constructed Computer-based Models of Physical Phenomena. Journal of Educational Computing Research, 44(2), 173-201.
14. Louca, L., Zacharia, Z. (2012). Modeling-based Learning in Science Education: Cognitive, Metacognitive, Social, Material and Epistemological Contributions. Educational Review, 64(4), 471-492.
15. Mioduser, D. et. al. (2012). Final report on DynaLearn evaluation studies, DynaLearn, EC FP7 STREP project 231526, D7.4.
16. Moody, D.L. (2005). Theoretical and practical issues in evaluating the quality of conceptual models: current state and future directions. Data & Knowledge Engineering, 55, 243-276.
17. National Research Council. (2012). A Framework for K-12 Science Education: Practices, Crosscutting Concepts, and Core Ideas. Washington, DC: The National Academies Press.
18. Sins, P. H. M., Savelsbergh, E. R., van Joolingen, W. R. (2005). The Difficult Process of Scientific Modelling: An Analysis of Novices Reasoning during Computer-Based Modelling. International Journal of Science Education, 27(14), 1695-1721.
19. Songer, N.B., Ruiz-Primo, M.A. (2012). Assessment and science education: Our essential new priority? Journal of Research in Science Teaching, 49(6), 683-690.
20. Teeuw,W., van den Berg, H. (1997). On the quality of conceptual models. In Proc.16th Int. Cont. on Conceptual Modeling (ER97), Los Angeles, California, USA.
21. IEEE standard for software verification and validation. (2004). TR 1012-2004, IEEE.