Applying Semiotic-Conceptual Analysis to Mathematical Language *

Uta Priss

Zentrum für erfolgreiches Lehren und Lernen, Ostfalia University of Applied Sciences
Wolfenbüttel, Germany
www.upriss.org.uk
orcid: 0000-0003-0375-429X

Abstract. This paper demonstrates how semiotic-conceptual analysis (SCA) can be applied to an investigation of mathematical language and notation. The background for this research is mathematics in higher education, in particular the question of why many students find mathematics difficult to learn. SCA supports an analysis on different levels: pertaining to the conceptual structures underlying mathematical knowledge separately from and in combination with the semiotic relationship between the representation and the content of mathematical knowledge. We believe that this provides novel insights into the structure of mathematical knowledge and presents a method of decoding which lecturers can employ.

1 Introduction

Semiotic-conceptual analysis (SCA) is a mathematical theory for investigating the use of sign systems which was developed by Priss (2017b). SCA provides a different angle compared to linguistic, semantic or usability analyses because it facilitates a separation of conceptual and semiotic aspects but also takes representations and interpretations into consideration. So far only a few applications of SCA have been described (Priss 2017a & 2016). This paper discusses how SCA can be applied to the language of mathematics itself. The purpose for this is to investigate further why many students perceive mathematics as a difficult subject to learn. The work of Priss (2018) which examines ambiguities in mathematical expressions and compares the structure of mathematical concepts with natural language concepts is extended in this paper.

The mathematical details of the formalisation of SCA are described by Priss (2017b) and shall not be repeated in this paper. The conceptual structures of SCA are modelled with formal concept analysis (FCA) whose mathematical details (cf. Ganter & Wille 1999) shall also not be repeated in this paper. In SCA a sign is a triple consisting of a representamen, a denotation and an interpretation so that the representamen and interpretation together uniquely identify the denotation. Or in other words, interpretations are partial functions that map representamens into denotations. Primarily these notions are role concepts which means that anything can be a representamen and so on if it is used in that role. But it is not really useful to consider just any triple as a sign. Therefore, the mathematical definition can be extended by some less formal,

* To appear in Proc of ICCS'19, LNCS , Springer-Verlag

fuzzy usage conditions: a sign should be something that is used in an act of communication; representamens should be physical representations (such as words, images, bits or neuron combinations), interpretations should contain some information about when, where, how, why and by whom a sign is used and denotations should describe meaning. Whoever designs an application decides what exactly the sets of representamens, denotations and interpretations are in that case. Signs can themselves be representamens, interpretations or denotations of other signs leading to a potential "semiotic chain".

Priss (2017b) defines formal contexts for denotations, representamens and interpretations but states that other conceptual structures can also be employed instead of or in addition to lattices. Interpretations can, for example, be modelled as temporal sequences or as possible paths through a computer program. Common structures on representamens are syntax and grammar. At least in the context of this paper where SCA is applied to mathematics, denotations should be modelled with FCA as formal concepts in concept lattices which can have additional features (axioms, functions or relations).

SCA has been elaborated in several publications (Priss 2016, 2017a, 2017b & 2018). The role of language in learning mathematics has been investigated by other researchers (cf. Schleppegrell's 2007 review) but not usually from a semiotic perspective. Other semiotic research in mathematics education (e.g. Presmeg at al. 2018) tends to be not as formal. In our view the mathematical model of semiotics that is most similar to SCA is provided by Goguen (1999). But unlike Goguen's formalisation, SCA separates conceptual and semiotic aspects. Thus, we believe that there is no existing research with the exact same focus as presented in this paper.

2 Differentiating conceptual and semiotic analyses

Denotations can be modelled in some *domain* which consists of all relevant information: conceptual hierarchies, relations, axioms and so forth. For example, set theory is a mathematical domain. In this paper denotations are modelled as concepts which are formalised with FCA. In general, the difference between concepts and denotations is that "concept" is a structural notion whereas "denotation" is a role. In this paper the two notions can be used fairly interchangeably.

In the remainder of this paper the term *conceptual analysis* refers to an analysis of conceptual structures within a domain. The term *semiotic analysis* implies that different parts of a sign (representamens, denotations and interpretations) contribute to an analysis. Using this terminology, it is possible to investigate separate aspects of signs: their underlying conceptual structures and their semiotic aspects, such as, how well structures amongst representamens correspond to denotational structures. For example, proofs in analytic geometry being simpler than the corresponding proofs in Euclidean geometry is primarily a conceptual difference and not a semiotic one. In FCA, formal contexts and concept lattices are two different conceptual structures. But whether vectors in linear algebra are represented graphically or as n-tuples is not a conceptual but a semiotic difference. Typical questions for a conceptual analysis focus on denotational structures, what kinds of concepts can be constructed in a domain and the complexity and efficiency of such structures. Typical semiotic questions focus on synonymy and polysemy, complexity and efficiency of representamens compared to denotations, how representamen relations correspond to conceptual relations, how suitable and complete the conceptual domain is for the representamens and on the usability of signs.

Complexity and efficiency are relevant for both conceptual and semiotic analyses. From a conceptual view, compound concepts are more complex than atomic concepts which could be axioms or object and attribute concepts[1] in FCA. Reducing complexity is inverse to increasing efficiency because having more atomic concepts is less efficient but makes it easier to express compound concepts. For example, defining an equivalence relation as a "binary relation that is reflexive, symmetric and transitive" is not very complex because it relies on previously defined terms "binary", "relation", "reflexive", "symmetric" and "transitive". A definition via a set of formulas without relying on previously defined terms would be more efficient and more complex. Similarly, a programming language with a large set of predefined functions yields programs that are short and simple compared to programs, for example, written in assembly language. In FCA, a clarified, reduced lattice has maximum efficiency but if all concepts were atomic then they would have minimum complexity.

Complexity and efficiency can also be viewed semiotically. Synonyms are signs whose different representamens are mapped into the same denotation. Synonymy should be kept to a minimum because it is inefficient. Polysemy means that one representamen is mapped into different but related denotations under different interpretations. Polysemy occurs for example, when the value of a variable in a computer program is changed. Polysemy increases efficiency because it means that the set of representamens can be even smaller than the set of atomic concepts because interpretations disambiguate the representamens. An example of this is that natural languages can express a much larger number of concepts than they have words or word combinations because the contexts in which the words are used disambiguate them.

Correspondence between grammatical/syntactic relations and conceptual relations refers, for instance, to the ordering, sequence, topology and distances in a graphical representation compared to such structures in the denotational domain. Usability investigates how adequate representamens are for denotations (including structural and qualitative features). But the denotational domain should also be adequate for the representamens, for example, sufficiently complete. Different users can be modelled as different interpretations. In this paper, interpretations are not mentioned very often because Priss (2017a) argues that mathematical texts mainly employ only a single interpretation corresponding to the "correct" meaning of mathematical signs as considered by experts. Interpretations are relevant when it comes to explaining the view of students. Their interpretations are often different from expert interpretations and may be an indication of misconceptions. With respect to experts, interpretations may be useful if one wants to analyse personal preferences of using and interacting with mathematical signs.

3 Conceptual analysis of mathematical signs

This section discusses mathematical signs from a conceptual viewpoint. Priss (2018) argues that many people find mathematics difficult to learn because mathematical con-

[1] The object concept of an object is the smallest concept which has the object in its extension. Vice versa an attribute concept is the largest concept which has an attribute in its intension.

cepts are formal and structurally very different from natural language concepts which are associative. Associative concepts depend heavily on the words used to express them and their situational contexts. For example, it is impossible to obtain a universal agreement about what "democracy" means. Priss (2017a) provides a definition of "formal concept" that is slightly wider than the usual FCA notion and essentially refers to concepts that have a necessary and sufficient definition. Apart from mathematical concepts other formal concepts are for example phylogenetic notions such as "Centaurea cyanus" because it provides necessary and sufficient conditions for whether something is a common cornflower or not. This paper addresses the question as to how such formal mathematical concepts can be modelled with FCA. The aim is not to apply FCA to mathematics which has been done numerous times before by Ganter & Wille (1999) and many others. Instead the aim is to obtain an understanding of what the concepts that underpin mathematical language themselves are like – by modelling them with FCA. For example, what is the extension and intension of a concept for "$=$" modelled with FCA?

We argue that denotations of mathematical signs are abstract concepts corresponding to mathematical equivalence classes because there can be many mathematically equivalent representamens for a mathematical sign. For example, after disambiguating a representamen "5" with respect to whether it is an integer, real number and so on, it can be assumed that there is a shared abstract notion of "5" which is independent of its representamen and situational context. Thus a single interpretation can be assumed for all representamens of the integer "5". This is contrary to "democracy" where such a shared abstract notion does not exist. It should be mentioned that mathematical concepts do develop over time. For example, the number "0" was historically invented (or discovered) much later than the natural numbers. Nevertheless we would argue that changes occur before a precise mathematical definition has been established. Once a mathematical concept has a formal definition, the meaning is exactly what is stated by the definition and only changes if the definition is changed.

Fig. 1 shows two examples of formal concept lattices that represent mathematical concepts. The purpose of Fig. 1 is to create lattices where FCA concepts represent mathematical concepts and where attribute implications might correspond to mathematical implications. The formal objects of lattice A are all pairs of the numbers 1, 2, 3, and 4 and all triples of the numbers 1 and 2. The intensions use a placeholder notation: \$1 refers to the first element of the vector, \$2 to the second and so on. For example, the vector (2,2) fulfils the equation $\$1 = \$2$ because $2 = 2$. The lattice contains some attribute implications which are correct mathematical statements, for example $\$1 \prec \$2 \Longrightarrow \$1 \leq \$2$ (where "$\prec$" is the predecessor-successor relation). But there are also implications which are not generally true, such as $\$1 + \$2 = \$3 \Longrightarrow_? \$1 = \$2$ and only appear in the lattice because of the very limited set of formal objects.

In theory, if it was possible to create lattice A for all pairs, triples and so on of natural numbers and all operations on them, then the attribute implications would be extensionally proven mathematical implications. But because such a lattice would be infinite and it is not clear what operations to include, this is not possible. Extensional proofs tend to only be possible if they pertain to finite sets or if there is some sort of construction mechanism that loops through an infinite set as is the case for mathematical induction. One interesting observation is that as soon as one enlarges the set of formal objects in lattice A many concepts emerge which are neither object nor attribute concepts. Many relationships in a larger version of lattice A appear to be coincidental. It is also questionable whether it is at all useful to combine pairs and triples in one set of formal objects because the resulting implications may not be interesting. In SCA the notion of *signature* is used to describe the length and datatypes of vectors that are used as formal objects. In lattice A the signature is length 2 or 3 with datatype natural numbers.

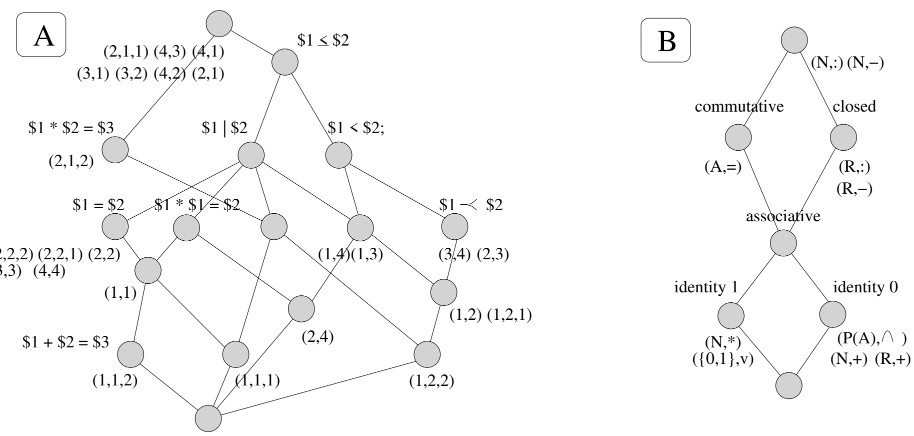

Fig. 1. Conceptual analysis of mathematical signs

Lattice B in Fig. 1 is constructed in a similar manner. In that case the signature of the formal objects describes a pair of a set with an operation. The letters "N" and "R" refer to the sets of natural and real numbers; "A" is a set and "P(A)" its powerset. The attributes are represented by words but the corresponding definitions should be clear. For example, "commutative" corresponds to $\forall_{x,y \in \$1} : x\$2y = y\$2x$. The formal object $(A, =)$ is unusual because it refers to a set A with the equals sign as a truth-valued operation. In that case the definition of relational symmetry $\forall_{x,y \in \$1} : x\$2y \iff y\$2x$ coincides with commutativity. This raises questions about adding some further restrictions about the formal attributes or objects in order to avoid combining notions such as commutativity and symmetry in one lattice.

A difference between the two lattices is that in lattice B, quantifiers are used in the intensions. Furthermore, the placeholder $2 refers to operations and is thus somewhat more abstract. In lattice A an algorithm could be provided that iteratively generates the set of formal objects even if it is infinite. But in lattice A it is not clear whether the set of attributes is in any way complete. In lattice B it is impossible to discuss an infinite set of formal objects because that would involve a set of sets in the first component of the pairs. Furthermore, there could be other algebraic structures with operations which nobody has yet thought about. Nevertheless the set of attributes in lattice B is complete if it is meant to correspond to the axioms of commutative monoids. Priss (2017b) suggests the use of the notion "open set" (from linguistics) for a set for which it is basically known whether an item belongs to it or not but which cannot be listed in its entirety. The set of common cornflowers that currently exists on this Earth is such an open set. Thus, it appears that some mathematical concepts have an extension or intension that is "open" in some manner because an infinite version of the set of objects in lattice B and the set of attributes in lattice A cannot be provided.

In both lattices the relationship between formal objects and attributes requires proofs but in lattice B the proofs are somewhat more complex. It is conceivable that in lattice B there could even be relationships between formal objects and attributes which are unknown because nobody has yet been able to construct a proof. This may not be the case for elementary group theory, but it is the case in other mathematical domains. In cases where it is not possible to provide an algorithm for a set of formal objects, mathematicians tend to rather look for a structural description of the elements instead of generating a list. For example, it is known how many different structural types of monoids or groups exist for sets of a certain size. For expert mathematicians all of this is familiar. But first year university students experience a transition from school mathematics which relies somewhat on associations and intuitions to a purely formal mathematics. They need to learn about the nature of formal, mathematical concepts and what exactly is formal and precisely defined and what is not yet known, not provable or open.

4 Semiotic analysis of mathematical signs

In addition to encountering the nature of formal mathematical concepts for the first time, first year university students also need to develop a tolerance for the ambiguity of mathematical notation. The textbooks used in schools tend to be more carefully designed and select a certain fairly consistent subset of mathematical notation. But at university level, students need to get accustomed to how mathematical language is used in a wider context of applications where different notations may be used. For example, the modulo operation can be written as: a mod n, $a\%n$, a (mod n) and use $=$, $\equiv$ or $\equiv_n$. Notations such as using $1/n$ for inverse elements in modular arithmetics lead to examples such as $1/2 = 1$ (in $\mathbb{Z}_3$) which require students to unlearn some aspects of their school knowledge. Some notations always depend on context. For example an edge AB in a directed graph is different from an edge AB in an undirected graph (where it equals BA) even though both use the same notation.

But SCA considers more than just analysing notations. It involves first specifying the extensions and intensions of the denotational conceptual structures as presented in the previous section. Then, second, it involves considering all 3 components of the signs. This section provides an overview of using SCA for analysing mathematical signs as discussed in the literature and adds a further small example. Priss (2018) argues that mathematical signs are incomplete, polysemous, make frequent use of strong synonymy (contrary to natural language) and because of frequent use of visualisations raise questions of iconicity. Incompleteness is often strongest when it comes to defining the basics because these tend to require some meta-level formalisations or are at a higher level of difficulty[2]. Polysemy is discussed below. Strong synonymy means that two representamens have exactly the same denotations (Priss 2017b). In natural languages, synonyms such as "car" and "automobile" tend to be similar but not exactly the same whereas mathematically equivalent denotations belong to strong synonyms. The stylistic use of synonyms is different in mathematics as well because in natural languages exact repetitions are often considered bad style and students may be taught to rephrase by using synonyms. In mathematics, however, a sign can only be replaced by a strong synonym but not by non-strong synonyms. For example, the words "function" and "relation" cannot usually replace each other in mathematical sentences. Iconicity of visualisations refers to relationships such as left/right, inside/outside, near/far, above/below and distance which may be present in visualisations. For an iconic sign there is some similarity between such relationships in its visualisation and in its denotation. It can be challenging for students to determine which relationships in visualisations matter and which do not. For example, in a Hasse diagram of a concept lattice it matters which nodes are above other nodes and connected via an edge but the length of the edges does not matter (Priss 2017a). Formulas often lack the iconic quality of left/right corresponding to first/later because formulas may require the reader to jump across different elements of the formula instead of the usual left-to-right sequence (Priss 2018). Other semiotic aspects of mathematical language pertain to syntactic rules (e.g. precedence rules), and the special use of variables (Priss 2018).

A means for identifying polysemy of mathematical notations is by first disambiguating and then translating them into a formal language which specifies how the signs are to be calculated or operated with. Mathematical signs often contain a hidden task relating to what is meant to be done with the signs which is clear to expert mathematicians but not explicitly notated. There is some evidence that at least school children sometimes perform nonsense calculations if they are incorrectly associating a certain task with a certain type of problem (e.g. Puchalska & Semadeni 1987). The following table shows the steps of disambiguating and identifying tasks of the polysemous meanings of the equals sign (as discussed by Priss (2016)):

nr	example	disambiguation	task
1	$1+2=?$		calculate(1+2)
2	$1+2=4$	"=" : Expr × Expr → { true, false }	eval_truth(=(1+2,4)))
3	$1+1+1=1+2=3$	abbreviation for $1+1+1=1+2$, $1+2=3$ and $1+1+1=3$	eval_truth(...)
4	$1+1+1=1+2=3$	{1+1+1,1+2,3} is an equivalence class	observe equivalence

Example 1 presents an operational use of the equals sign. It conveys a request to perform a calculation similar to a function call in programming languages with the sum of 1 and 2 as input and the result 3 as output. Because example 2 is false its equals sign can only be interpreted as a truth-valued function that maps a pair of expressions $(1+2,4)$ into the set of values "true" or "false". Example 3 is an abbreviation for two or three statements of example 2. But it could also be an indication of equivalence as represented by example 4. The equals signs in example 4 cannot be considered truth-valued functions because if $=(1+1+1,2+1)$ yields "true" then it would result in

[2] For example, definitions of "set", "$\Longleftrightarrow$" or the Peano axioms for natural numbers are challenging for first year students.

"true $= 3$" which is meaningless. Example 4 is semiotically similar to $2 < 3 < 4$ and could also be interpreted as a graph where "$=$" or "$<$" are vertices and transitivity is implied similar to a Hasse diagram. The denotations of the four examples are similar. Thus, it is a case of polysemy. Specifying exactly what is meant by mathematical signs as in the table highlights subtle differences in meaning.

5 Conclusion

SCA provides insights into the nature of mathematical concepts and notations which mathematicians (apart from teachers) may not be aware of. SCA can be employed as one tool amongst others for decoding hidden difficulties in the mathematical subject matter. A conceptual analysis of mathematical signs shows that mathematical concepts are structurally different from natural language concepts and that there may be a divergence between extensional and intensional aspects of the mathematical domain. A semiotic analysis demonstrates that mathematical language can be ambiguous, context-dependent and full of synonymy and polysemy. Mathematical representamens hide a fair amount of detail about their underlying concepts and the tasks that are to be performed with them. Similar looking representamens can have totally different meanings.

One method of helping students to understand the intricacies of mathematical signs is to have students translate mathematical notations into a computer programming language. This is, for example, advocated by Priss & Riegler (2017). The reason for this is that in programming languages the structures and datatypes of all elements always need to be explicitly provided. Furthermore, a certain functional abstraction is required in order to implement tests for features such as commutativity where the operation itself is an input to a function. A disadvantage of teaching mathematics by programming is that the students need to learn a programming language in addition to the mathematical notation. But there are some programming languages (such as SETL or Python) which support expressions that are fairly close to mathematical language.

References

1. Ganter, B.; Wille, R. (1999). *Formal Concept Analysis. Mathematical Foundations.* Berlin-Heidelberg-New York: Springer.
2. Goguen, J. (1999). *An introduction to algebraic semiotics, with application to user interface design.* Computation for metaphors, analogy, and agents. Springer, p. 242-291.
3. Presmeg; et al. (eds.) (2018). *Signs of Signification.* ICME-13 Monograph, Springer 2018.
4. Priss, U. (2016). A Semiotic-Conceptual Analysis of Conceptual Learning. In: Haemmerle; Stapleton; Faron-Zucker (eds.), Graph-Based Representation and Reasoning, Proceedings of ICCS'16, Springer, LNCS 9717, 122-136.
5. Priss, U.; Riegler, P. (2017) *Automatisierte Programmbewertung in der Mathematik-Ausbildung.* In: Automatisierte Bewertung in der Programmierausbildung, Waxmann.
6. Priss, U. (2017a). *Learning Thresholds in Formal Concept Analysis.* In: Bertet; et al. (eds.), Formal Concept Analysis. Proceedings of ICFCA'2017, Springer, LNCS 10308, p. 198-210.
7. Priss, U. (2017b). *Semiotic-Conceptual Analysis: A Proposal.* International Journal of General Systems, Vol. 46, 5, p. 569-585.

8. Priss, U. (2018). *A Semiotic-Conceptual Analysis of Conceptual Development in Learning Mathematics.* In: Presmeg; et al., Signs of Signification, ICME-13 Monograph, Springer 2018.
9. Puchalska, E., & Semadeni, Z. (1987). *Children's reactions to verbal arithmetical problems with missing, surplus or contradictory data.* For the learning of mathematics, 7(3), 9-16.
10. Schleppegrell, M. J. (2007). *The linguistic challenges of mathematics teaching and learning: A research review.* Reading & Writing Quarterly, 23(2), 139-159.