



HAL
open science

Surveying teachers' conception of programming as a mathematics topic following the implementation of a new mathematics curriculum

Morten Misfeldt, Attila Szabo, Ola Helenius

► To cite this version:

Morten Misfeldt, Attila Szabo, Ola Helenius. Surveying teachers' conception of programming as a mathematics topic following the implementation of a new mathematics curriculum. Eleventh Congress of the European Society for Research in Mathematics Education, Utrecht University, Feb 2019, Utrecht, Netherlands. hal-02417074

HAL Id: hal-02417074

<https://hal.science/hal-02417074>

Submitted on 18 Dec 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Surveying teachers' conception of programming as a mathematics topic following the implementation of a new mathematics curriculum

Morten Misfeldt¹, Attila Szabo² and Ola Helenius³

¹Aalborg University, Department of Learning and Philosophy; misfeldt@learning.aau.dk

²Stockholm University, Stockholm Education Administration; attila.szabo@stockholm.se

³University of Gothenburg, National Center of Mathematics Education, ola.helenius@ncm.gu.se

In this paper, we investigate mathematics teachers' conception of the relationship between mathematics and programming. The context of the investigation is a recent curriculum reform in Sweden that makes programming a compulsory element of the national mathematics standards. Following up on an in-service training initiative, we conducted a pilot survey (N = 133) exploring – among other things – the teachers' conception of the relationship between mathematics and programming. The results suggest that the teachers, on average, feel that there is a strong, but not very strong, relationship between the two subjects. Furthermore, the results suggest that mathematics teachers are interested in working with programming but that they do not feel well prepared for taking on that task. These results are used to discuss the mathematical potential of the different ways in which compulsory programming can be introduced in schools.

Keywords: Programming, mathematics, computational thinking, curricular changes.

Introduction: Programming as a new element in compulsory school

Programming, computing, computational thinking (Wing, 2006), technology as a subject, and similar topic formulations are currently being included in the curriculum of compulsory schools in many countries. The purpose of these changes is ostensibly to support students in becoming technologically literate and able to participate in the society of the future, as well as to promote STEM careers. Different countries are taking different routes to implement this change. In Great Britain, the main idea has been to develop a computer science curriculum as a new topic, whereas other countries such as France, programming is included in the mathematics curriculum. European approaches to computing in schools differs (Vahrenhold et al., 2017) but most countries have technological skills and literacy as increasingly important objectives of K–12 teaching students, and they share the challenge of building the necessary teacher capacity (Wilson et al., 2010).

The specific relationship between mathematics and programming have been explored by scholars such as Papert (1980) and Dubinsky and Harel (1992) and we know that the potential for learning mathematics through programming depends on a number of other contextual factors (Misfeldt & Ejsing-Dunn, 2015). Investigations of the relation between programming and mathematics, have typically taken aim at the potential of using programming as a vehicle for developing mathematical competence. The current situation is almost the converse; programming is included in the national curricula and standards and mathematics, is in some cases used as a vehicle to meet that goal.

In Sweden programming became a compulsory part of mathematics (and science) in autumn 2018. Therefore, several in-service activities have been developed and were already started in spring 2018 and all three authors of this paper have been involved in the development. As part of this work, we

decided to conduct a survey of how Swedish mathematics teachers have experienced the introduction of programming as a topic in their teaching. This paper is our first report on the data from this survey¹, and we will continue by describing the Swedish situation and then go on to the theoretical understanding that underpins our work, as well as our method. In the last part of the paper, we present descriptive statistics on the teacher's conception of the relationship between mathematics and programming in his/her class. We end the paper by discussing how to implement programming in relation to mathematics in light of the presented results.

An in-service professional development program in Sweden

The structure of the Swedish national curricula of both primary and secondary mathematics (grades 1–9) as well as upper secondary school builds on certain special “central content,” such as algebra, numbers, statistics, and geometry as well as five to seven competencies (e.g. problem solving, reasoning and communication). The grading criteria mostly relate to these competencies. Programming has been included in school mathematics through the revision of the curricula, in which it is referred to in connection with the specified central content.

The Swedish National Agency of Education is responsible for the implementation of programming in grades 1–12, while Swedish schools are governed by municipalities, companies, or nonprofit organizations. The National Agency has arranged courses in programming for teachers. These courses are not related to mathematics teaching but are simply oriented towards developing the teachers' own programming competency. To attend to the didactical question of how to include programming in mathematics teaching, the National Agency also decided that specific professional development (PD) modules should be designed².

The context for the creation of these modules is that Sweden has recently carried out a large-scale PD program involving all teachers of mathematics in grades 1–9 and in upper secondary school. Boesen, Helenius and Johansson (2015), describes the program and effect and Lindvall, Helenius and Wiberg (2018) describes the program theory that was used in the initial design. Briefly, the program consists of several different PD modules that teachers may choose among. These modules were designed by researchers and teacher educators. Each part of the module follows a cycle in which teachers first study the PD material individually (60 minutes per part), then meet in groups and discuss the PD material and plan lessons together (90 minutes), after which they carry out those lessons with their regular classes (one lesson), and then again discuss the material and the lessons (60 minutes). Schools are monetarily compensated for teachers to attend two such modules, but the modules are free to use for teachers or schools that want to do more PD. As support for the implementation of programming in school mathematics, the National Agency decided that a special programming module should be designed. The first and third author of this paper were part of the

¹Since the data collection is ongoing, this paper can be regarded as a pilot study. More results and insights will be explored further in the continuing work.

²The digital material supporting the professional development program can be found here: <https://larportalen.skolverket.se/#/moduler/0-digitalisering/alla/alla>

design team. We describe the conceptual framework underlying this module below, but one interesting difficulty was that the level of programming competency among teachers was unknown. From reviewing teacher education, based simply on our background knowledge of Swedish teachers, we knew that their programming knowledge and experience were probably rather basic and probably lower the earlier in the school system that the teachers taught. However, at the time, the time frame available for the design of the module did not allow more research into this question. The future effects of such a large-scale programming reform will be interesting to follow up, and as an aid to better understand such effects, we carried out a survey among 133 teachers. The intent of this survey was to establish a knowledge base for understanding the situation among teachers now and to form a basis for further longitudinal investigations. The present paper describes the design of this survey by relating it to some of the ideas we developed when constructing the PD module and presents some preliminary survey results.

Conceptual framework and ideas about the relationship between programming and mathematics

When designing the PD module, a view of mathematics that we found helpful was the Piagetian-based work by Vergnaud, in particular his work on representations (1998). Briefly, mathematical ideas (concepts, theorems, etc.) are, on one hand, related to situations where some particular mathematics makes sense and, on the other hand, related to representations, such as language or other semiotic systems used to signify mathematical matters. Vergnaud's reasoning can be extended to programming, but here, situational phenomena are instead represented in terms of programming code. A real-world situation hence can be modelled either by programming code or by mathematical concepts. Moreover, for understanding, analyzing, or improving the code, it may often make sense to use mathematics, and conversely, if a mathematical model of a situation already exists, that can often be translated into code. This creates an interesting and quite complex dynamic among real-world situations, the world of code and the world of mathematics.

This prompted us to consider two different programming situations in relation to mathematics. Type 1 is where the programming concerns some concrete phenomenon – to take a classic example, a model of a turtle moving in some particular pattern. For type 1, mathematics is not necessarily a part of the modelling of, for example, the turtle's movements. This modelling can be done purely in programming code. Therefore, for mathematics lessons based on type 1 programming, the teacher must assume the responsibility of extracting or inserting mathematics into the lesson. Using the work of Misfeldt and Ejsing-Dunn (2015) on programming and mathematics, we described three intersecting points or potentials: (1) viewing students as producers of code; (2) supporting abstract thinking; and (3) developing algorithmic thinking. Awareness of such potentials could, we hypothesized, assist teachers when helping students to mathematize their work in type 1 situations.

The type 2 situation, is when the programming part concerns something that is already mathematical, as in using programming to build a mathematical tool, solve a mathematical problem, or to explore mathematical phenomena. In type 2 situations, the mathematics is already relatively fixed, but an important responsibility of the teacher, particularly when working with pupils who are programming novices, is to make sure that the pupils are properly introduced to elements of

programming that can be useful in approaching the mathematical problem at hand (Guin & Trouche, 2002).

In order to investigate the teachers' ideas about the relation between mathematics and programming empirically, we have chosen to focus on how the teachers conceptualize this relation. We use the notion of *conceptions* to signify what the teachers think are important aspects of the relation between the two areas of knowledge. Our construct of conception is built upon the same constructivist post Piagetian approaches as our mathematics education approaches (Papert, 1980; Vergnaud, 1998), and as a consequence, active articulation and representation of the relation between the mathematics and programming plays a critical role in shaping individual conceptions. We explore the conceptions through questions focusing on importance of the relation, preparedness of the teachers to teach programming and the qualities of the relations between the two topics, based in the three intersection points from Misfeldt and Ejsing-Dunn (2015).

Methods and instruments

To understand how the change in curriculum and the work in the PD modules are experienced by Swedish mathematics teachers, we designed a survey addressing various aspects of these issues.

Participants

The participants were teachers from school years 1–12 in municipal schools in Stockholm. Some months before the study, the participants attended seminars organized by the Education Administration that aimed to facilitate the introduction of programming into school mathematics. Prior to these seminars, the participants – as well as the vast majority of Swedish mathematics teachers – had no formal education in and very little experience in programming. When attending the seminars, teachers submitted their email addresses, and these addresses were subsequently used to contact them about the survey. Answering the survey was optional, and the teachers were guaranteed anonymity. After the responses were gathered, the teachers who responded received the opportunity to participate in the present study. They were informed that their answers and personal data would be stored according to the current GDPR regulations. Currently, 133 teachers have chosen to participate in the study, 77% (102 teachers) from primary school and 23% (31 teachers) from upper secondary school – however, not all teachers answered every question of the survey.

The survey

The survey was developed by the authors of this paper; the included statements are based on the revision of the Swedish curricula with regard to programming in mathematics, on feedback from teachers at seminars about introducing programming, and on authors' experience of designing PD programs for mathematics teachers. The statements deal with 17 topics, representing the categories: a) teachers' prior experiences of teaching mathematics and programming; b) teachers' conceptions of how programming should be implemented in mathematics; c) teachers' conceptions of how programming could develop pupils' understanding of mathematical concepts and procedures, and pupils' problem-solving competency. In order to achieve an appropriate level of reliability and validity of the responses (expected to be above 100) – in concordance with studies about survey scales in social sciences, that point out that the “optimum number of alternatives is between four

and seven” (Lozano, Garcia-Cueto, & Muniz, 2008) – we decided to use a four-alternative Likert scale. By using an even scale, i.e., a scale without a neutral middle category – and by taking into consideration that the participants had little or no previous experience of programming – we aimed to discern participants’ responses in more distinct ways. Thus, participants were asked to answer to every statement on a scale of 1 (meaning “not at all”) to 4 (meaning “to a great extent”).

The survey underwent a substantial a priori testing with a group of teachers with similar levels of experience in programming in mathematics as the participants, thereby confirming that included statements were compatible with the aims of this study. In this paper, we are focusing teachers’ answers to 12 statements – displayed at the next section – related to mathematical thinking and mathematics as a topic and a set of learning objectives on one side, and programming activities and objectives on the other. The first eight statements focus on teachers and pupils’ mathematical knowledge and potential related to programming, and on their understanding of mathematical concepts, procedures and algorithms in the context of programming. The last four statements deal with teachers’ enthusiasm and preparations related to programming, their views on programming and mathematics as different subjects and on learning programming by tinkering.

Results

We performed a simple descriptive analysis in this paper, thus, the paper does not present or confirm any hypotheses. Rather, we generated an overview of the replies and present our questions and their relationship to the framework.

In the survey, we gathered data about a number of things as described above. We asked the eight questions shown in Table 1, and the distributions of the answers were more or less the same.

Question	Not at all	2	3	To a great extent
I benefit from my mathematical knowledge when I am programming.	8	32	59	32
My pupils benefit from their mathematical knowledge when they are programming.	7	36	57	28
I can do more things related to mathematics when using programming.	12	34	60	22
My pupils can do more things related to mathematics when using programming.	11	39	62	15
I can obtain a better understanding of mathematical concepts by using programming.	20	43	48	14
My pupils can obtain a better understanding of mathematical concepts by using programming.	19	33	57	16
I can obtain a better understanding of mathematical procedures and algorithms by using programming.	15	33	54	27
My pupils can obtain a better understanding of mathematical procedures and algorithms by using programming.	9	26	64	29

Table 1: The distribution of the number of answers to respective questions about the relationship between programming and mathematics.

As seen in Table 1, the most typical answer was category 3. The answers to every question have more or less the same distribution; almost half of the teachers answered 3 and less than one fifth of them answered 4 (“to a great extent”). The rest of the teachers, less than one third of them, were divided between 1 (“not at all”) and 2 (that might be interpreted as “to a small extent”).

Do you feel enthusiastic about teaching programming in mathematics?	23	29	55	23
Do you feel well prepared to teach programming in mathematics?	40	42	43	6
Programming and mathematics are two different disciplines that should not be taught within the same subject.	52	38	24	18
It is important that students learn to modify and adapt given programs according to certain criteria (tinkering).	12	29	64	20

Table 2: The distribution of the number of answers associated to respective statements about teachers’ motivation for and self-esteem regarding teaching programming in their mathematics classes.

According to Table 2, the answers associated to the teachers’ levels of preparation to teach programming in mathematics respectively to their opinion about the relationship between programming and mathematics as disciplines, are distributed towards less affirmative categories. All together, we take the results shown in Table 1 and Table 2 as signs of two issues that both need further statistical inquiry in order to be considered stable results: (1) Swedish mathematics teachers do see a clear relationship between mathematics and programming – however, this does not mean, that the teachers view these topics as coinciding; (2) despite admitting the benefits of programming, Swedish mathematics teachers are not well prepared for teaching programming in mathematics. Rather, the teachers express that they and their students can use their mathematical competencies while programming and feel that it is a good idea to do this. Yet, simultaneously, they do not currently feel well prepared to take on the task of teaching programming. In the following section, we will discuss these results.

Discussion: approaches to including programming in compulsory schools

Many countries, municipalities, and schools are currently struggling with how to introduce programming as a body of knowledge in the school system. Mathematics tends to play an important role in a number of these attempts. There are many ways in which this can be done in terms of curriculum structure. Programming can be included in mathematics and science curricula, as in the case we have explored here. Of course, this can also be done in different ways, and different teachers will have different competencies in this regard. Another approach is to develop programming as a topic in its own right. However, such a topic can have different “flavors” and foci and different relationships to the more traditional school topics. The Swedish case is an example of a country addressing this problem by imbedding programming into mathematics (and science), and the data from our case suggest that there are a number of synergies and potentials with such an approach. In general, the teachers who answered our survey do see a relationship between programming and mathematics, and they do want to build on that relationship in their teaching. This tells us that the route of integrating programming into mathematics seems feasible in Sweden. However, we should be aware of the fact that both the sample size and current sampling strategy challenge the generalizability of the results. Accordingly, these results say nothing about whether

combining mathematics and programming is the best way to address the integration of programming in Swedish schools or even if this integration is a good idea in the first place. Yet, the relationship between the topics and the motivation for taking on the task seems to be there, even though the capacity for teaching programming is not yet in place. We are not free from the problem of training teachers to take on this specific task.

The introduction of programming in the mathematics curriculum poses an interesting implementation problem. What are the critical choices and concerns that we need to consider in this respect? In this paper, we have addressed mathematics teachers' conception of the relationship between mathematics and programming, but this is not the only relevant concern. Stakeholders such as end users (pupils and teachers), implementation plans, the potential for diffusion, etc., are important concerns. We also believe that the discussion of whether to teach programming in relation to other topics or as a topic in its own right could benefit from a broader "implementation framework," such as the one discussed by Century and Cassata (2016). To mention just one concern, it is critical to have sound implementation of programming in relation to mathematics education, and that could benefit from being viewed through the lens of implementation research; we can look at teacher's resistance to programming as part of mathematics. This resistance is rare but might exist, as seen in Table 2. If programming is included as a part of mathematics at all levels, no mathematics teachers can avoid teaching it. Accordingly, programming – that is, not a priori natural tool for mathematics education – might also be viewed in the light of instrumental distance and double professional geneses from the part of the teachers (Haskejian, 2014). In that sense, this structure is more fragile for the minority with negative views, whereas the development of an entirely new topic might make it easier for teachers to choose whether they want to engage in the teaching of programming.

Conclusion

In this paper, we have presented an initial descriptive analysis of Swedish mathematics teachers' experience regarding the relationship between mathematics and programming. The data collection is still ongoing, and the sample size is expected to increase. Furthermore, we are not yet in a situation where we can conduct a strong statistical analysis of the data, so the stability and generalizability of the results are currently limited. However, the results that we do have point in the direction of a meaningful relationship between mathematics and programming that can be used to support the teaching of programming within the subject. Yet, they also suggest that this will not be easy and that Swedish teachers – despite a positive attitude towards working with programming in mathematics – do not feel that they are prepared to take on this task. This result indicates that it might be a good idea to build the capacity for teaching programming from within the topic of mathematics. However, the capacity problem seems to persist in the sense that it is indicated that teachers need to be prepared and trained to teach programming. Furthermore, it might be the case that some teachers do not at all see the relationship between mathematics and programming, nor the relevance of teaching programming in their mathematics classes, and such teachers might also experience the task as very challenging. However, we assume that further data collection and statistical analysis will allow us to better understand the detailed clustering of teachers' attitudes on the subject of programming.

References

- Boesen, J., Helenius, O., & Johansson, B. (2015). National-scale professional development in Sweden: theory, policy, practice. *ZDM Mathematics Education*, 47(1), 129–141.
- Century, J., & Cassata, A. (2016). Implementation research: Finding common ground on what, how, why, where, and who. *Review of Research in Education*, 40(1), 169–215.
- Dubinsky, E., & Harel, G. (1992). *The Concept of function: aspects of epistemology and pedagogy*. Washington, DC: Mathematical Association of America.
- Guin, D., & Trouche, L. (2002). Mastering by the teacher of the instrumental genesis in CAS environments: necessity of instrumental orchestrations. *Zentralblatt für Didaktik der Mathematik*, 34(5), 204–211.
- Haspekian, M. (2014). Teachers' instrumental geneses when integrating spreadsheet software. In A. Clark-Wilson, O. Robutti, & N. Sinclair (Eds.), *The Mathematics Teacher in the Digital Era* (pp. 241–275). Dordrecht: Springer Science and Business Media.
- Jones, S. P., Bell, T., Cutts, Q., Iyer, S., Schulte, C., Vahrenhold, J., & Han, B. (2011). *Computing at school. International comparisons*. Retrieved from <https://www.computingschool.org.uk/data/uploads/internationalcomparisons-v5.pdf>
- Lindvall, J., Helenius, O., & Wiberg, M. (2018). Critical features of professional development programs: Comparing content focus and impact of two large-scale programs. *Teaching and Teacher Education*, 70, 121–131.
- Lozano, L. M., Garcia-Cueto, E., & Muniz, J. (2008). Effect of the number of response categories on the reliability and validity of rating scales. *Methodology: European Journal of Research Methods for the Behavioral and Social Sciences*, 4(2), 73–79.
- Misfeldt, M., & Ejsing-Duun, S. (2015). Learning Mathematics through Programming: An Instrumental Approach to Potentials and Pitfalls. In K. Krainer & N. Vondrová (Eds.), *Proceedings of the Ninth Congress of the European Society for Research in Mathematics Education* (pp. 2524–2530). Prague, Czech Republic: Charles University in Prague.
- Papert, S. (1980). *Mindstorms: children, computers, and powerful ideas*. New York: Basic Books.
- Vahrenhold, J., Nardelli, E., Pereira, C., Berry, G., Caspersen, M. E., Gal-Ezer, J., Kölling, M., A. McGettrick, A., & Westermeier, M. (2017). *Informatics Education in Europe: Are We All in the Same Boat?* New York: Association for Computing Machinery.
- Vergnaud, G. (1998). A comprehensive theory of representation for mathematics education. *The Journal of Mathematical Behavior*, 17(2), 167–181.
- Wilson, C., Sudol, L. A., Stephenson, C., & Stehlik, M. (2010). *Running on empty: The failure to teach K-12 computer science in the digital age*. New York: Association for Computing Machinery.
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33–35.