Exploring aspects of computational thinking in mathematical modelling projects: Insights from a project week in a German secondary school

Explorando aspetos do pensamento computacional em projetos de modelação matemática: Ideias decorrentes de uma semana de projetos numa escola secundária alemã

Lukas Bayer
University of Koblenz
Germany
bayer@uni-koblenz.de

Martin Bracke
University of Koblenz
Germany
bracke@uni-koblenz.de

Jean-Marie Lantau
Kant-Gymnasium Boppard
Germany
lantau@kant-gymnasium-boppard.de

Mareike Theobald
University of Koblenz
Germany
theobald@uni-koblenz.de

Abstract. This study investigates the role of computational thinking (CT) within mathematical modelling projects in secondary school mathematics education. We conducted a four-day mathematical modelling project with 14 students from grades 9 to 11 during a project week before summer vacations at a secondary school in Germany. Our observations revealed that various CT aspects, such as data collection, pattern recognition, and abstraction, naturally emerged in students' modelling activities, with these aspects being closely tied to the specific nature of the modelling problems. These findings suggest that mathematical modelling projects offer rich opportunities to develop CT skills in students. Furthermore, our research highlights how fostering CT can enrich the

Quadrante: Revista de Investigação em Educação Matemática 33(2) 130-150
https://doi.org/10.48489/quadrante.36870
Received: July 2024 / Accepted: December 2024

modelling process and assist students in the mathematical problem-solving process. By illustrating the synergy between CT and mathematical modelling, this study underscores the potential of integrating computational thinking into mathematics education to prepare students for the challenges of the digital age.
Keywords: computational thinking; mathematical modelling; mathematics education; secondary education; problem solving; STEM education.

Resumo. Este estudo investiga o papel do pensamento computacional (PC) em projetos de modelação matemática na disciplina de matemática no ensino secundário. Conduzimos um projeto de modelação matemática de quatro dias com 14 alunos do 9º ao 11º ano, durante uma semana de projetos, antes das férias de verão, numa escola secundária na Alemanha. As nossas observações revelaram que vários aspectos do PC, como a coleta de dados, o reconhecimento de padrões e a abstração, emergiram naturalmente nas atividades de modelação dos alunos, com esses aspectos estando intimamente ligados à natureza específica dos problemas de modelagem. Esses resultados sugerem que os projetos de modelação matemática oferecem oportunidades profícuas para desenvolver capacidades de PC nos alunos. Além disso, a nossa investigação destaca como o incentivo do PC pode enriquecer o processo de modelação e ajudar os alunos no processo de resolução de problemas matemáticos. Ao ilustrar a sinergia entre o PC e a modelação matemática, este estudo sublinha o potencial de integrar o pensamento computacional na educação matemática para preparar melhor os alunos para os desafios da era digital.
Palavras-chave: pensamento computacional; modelação matemática; educação matemática; ensino secundário; resolução de problemas; educação STEM.

Introduction

Although the relationship between computational thinking and mathematics has been an active area of research (Ye et al., 2023), work in the field of computational thinking and mathematical modelling seems to be lacking.

Ang (2021) compared different modelling approaches and identified several possibilities for aspects of computational thinking (abstraction, decomposition, pattern recognition and algorithms) to occur, stating that those ideas would have normally been observed as mathematical problem solving. He concluded that the ability to make use of computational tools is a key aspect that differentiates computational thinking and mathematical modelling and also benefits the latter.

Villa-Ochoa et al. (2022) investigated a mathematical modelling course where future teachers were tasked with developing a mathematical modelling project, highlighting the work of one group that focused on the use of digital tools. They analysed the developing process and identified multiple instances of computational thinking aspects present in the modelling project.

Regarding further empiric research, Teck et al. (2023) looked at the development of computational thinking during a mathematical modelling project with seven students from a private school in Malaysia. They rated the students' abilities in different aspects, like decomposition, from 0 to 2 based on observations of the modelling process, interviews with the students as well as a test deployed before and after the modelling and found improvements over the course of individual and group modelling tasks.

This study aims to further contribute to the empirical body of literature by highlighting the aspects of computational thinking found in mathematical modelling projects with secondary school students and identifying opportunities for computational thinking and mathematical modelling to benefit each other.

Theoretical background

Mathematical modelling

In this article, we understand mathematical modelling as the process of translating a real life problem into a mathematical model and solving the problem by using that model (Greefrath & Vorhölter, 2016).

Over the years, the theoretical background of mathematical modelling has evolved and a series of "modelling cycles" have emerged that illustrate the underlying process, with the modelling cycle of Blum & Leiß (2007) being one of the most prominent ones. Based on that cycle, Greefrath (2011) looked at the influence of digital tools on modelling and developed a modelling cycle that takes into account a computer model when working with digital tools. To use a digital tool, the mathematical model has to be first translated into a computer model. Then the computer can produce a computer solution, which can be translated back into a mathematical solution and then be interpreted in the real context.

Modelling projects, as regarded in this study, are characterized by open problems from authentic and real-life contexts. To enable students to carry out such projects, several universities in Germany implemented so called "modelling weeks" and "modelling days", which were carried out for example since 1993 at the university of Kaiserslautern, even partly with support of industry partners (Bock & Bracke, 2015; Greefrath & Vorhölter, 2016).

The aim of modelling weeks or days is to provide an opportunity for students to solve mathematical modelling problems independently. Both modelling weeks and days have a similar structure, starting out with the presentation of "authentic problems" (Bock et al., 2017), from which the students can choose, before going into a free working phase in groups. The students will than work over the course of a few whole school days or a week on the given modelling problem before presenting a possible solution, usually repeating the steps of the modelling cycle several times. The steps do not necessarily happen in order and individual "modelling routes" (Borromeo Ferri, 2007) can be observed.

Using authentic problems often comes with a form of product orientation of the modelling process. Although presentations can be seen as a form of product, this generally includes algorithms, computer programs or prototypes that can be used by the client, that is, the problem poser. The client and the product can then act as an external control to provide another form of support when carrying out modelling projects (Bock et al., 2017).

Computational thinking

Since the influential article of Wing (2006) multiple authors proposed possible definitions and frameworks for the term *Computational Thinking (CT)*. One of the most cited definitions by Cuny et al. (2010, as cited in Wing, 2010) describes CT as "the thought processes involved in formulating problems and their solutions so that the solutions are represented in a form that can be effectively carried out by an information-processing agent".

Since then, further definitions and multiple frameworks have been proposed to define, what thought processes and aspects are part of CT. Dong et al. (2019) consider *pattern recognition, abstraction, decomposition,* and *algorithms* to be the cornerstones of CT. Selby and Woolard (2013) additionally view the aspect of *evaluation* as part of CT and replace pattern recognition with the broader term of *generalisation*.

Weintrop et al. (2016) consider the implementation of CT in science and mathematics classrooms and defined a taxonomy of twenty-two CT practices, which are categorized into *data practices, modelling & simulation practices, computational problem-solving practices,* and *systems thinking practices.* The taxonomy comprises a wide variety of different practices such as *collecting data, designing computational models, debugging* or *programming*.

Following a review of existing frameworks, Shute et al. (2017) developed a framework focused on the foundational understanding of CT in K-12 subjects. They consider CT to be composed of the six CT facets: *decomposition, abstraction, algorithms, debugging, iteration,* and *generalisation*. They also identify three subcategories of abstraction, which are *data collection and analysis, pattern recognition* and *modelling,* and subcategories of the algorithm facet, which are *algorithm design, parallelism, efficiency,* and *automation.*

Kallia et al. (2021) focus on the intersection of CT and mathematical thinking (MT). They consider both to be problem solving processes that emphasise *contextualisation*, which is the translation from real world situations to a mathematical or computational model and vice versa. They identified four distinct cognitive activities in this process: (1) translating from a context into a model, (2) working within the model, (3) translating the result from the model back into the context and (4) verifying that the solution solves the problem. This is depicted in Figure 1.

Furthermore, Kallia et al. (2021) view combining MT and CT as using mathematics as a context for CT, from which they obtained their process model in Figure 2. This characterisation of CT in the context of mathematics is structurally similar to the mathematical modelling cycle under the influence of digital tools (Greefrath, 2011).

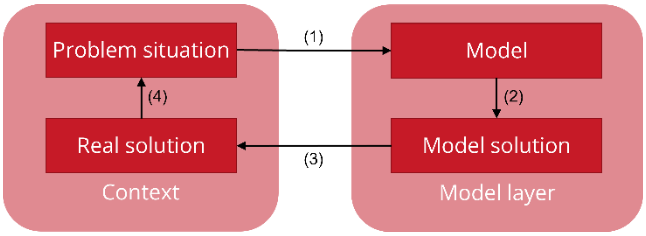

Figure 1. Contextualisation activities. The model layer can be either of mathematical or computational nature (adapted from Kallia et al., 2021)

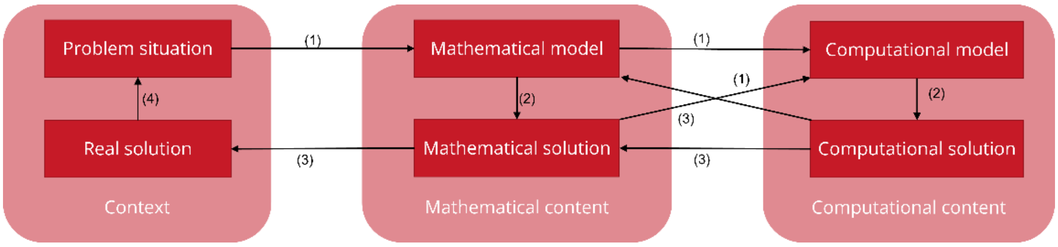

Figure 2. CT in the context of mathematical thinking (adapted from Kallia et al., 2021)

Regarding the implementation of CT in mathematics education, Kallia et al. (2021) conducted a literature-informed Delphi study to identify the aspects of CT that can be addressed in mathematics courses: *data analysis, data representation, abstraction, decomposition, algorithmic thinking, pattern recognition, automation, modelling, generalisation,* and *evaluation.*

Methodology

Mathematical modelling and CT describe problem solving processes, that, considering the cycles of Greefrath (2011) and the model by Kallia et al. (2021) in Figure 2, follow similar steps. As modelling is also present in multiple CT frameworks such as those of Weintrop et al. (2016) and Shute et al. (2017), the relationship of CT and mathematical modelling should be investigated more closely, especially when considering to enhance mathematics education with CT. Therefore, our research addresses the following two research questions, with the first being the primary focus of this paper:

1. What aspects of CT are present in mathematical modelling projects with students?
2. How can CT be fostered by mathematical modelling and how can CT support the mathematical modelling process?

To answer our research questions, we observed the mathematical modelling activities during a modelling project with fourteen students from grade 9 to 11 at a secondary school in Germany. This project was part of a project week at that school before summer holidays and the modelling project was offered by a teacher (T) supported by two researchers (R1 and R2) and a trainee teacher (TT) as supervisors. The project week spanned four days of working in the respective projects and one day presenting the projects at a school festival, following a similar structure as modelling weeks. Rather than observing a modelling week explicitly designed to foster CT, the observed project week was proposed by the teacher T without having CT in mind and the researchers were asked to support the project.

Seven modelling problems were offered, of which five were proposed by the students and two by the researchers. The students were free to use the time until the presentation day as they regarded meaningful for their chosen problem, leaving them with overall 18 hours (including breaks) to work on the problems over the four days. The researchers and teachers supervised the modelling activities and only assisted when problems emerged or the groups had questions, and helped as much as needed but as little as possible, following the "principle of minimal help" (Aebli, 2019, p. 300).

The groups of students moved freely around the school. Due to logistical reasons and privacy issues of non-participating students, audio or video recording wasn't feasible and data was collected via participant observation.

The modelling activities in the project were observed by R1 and field notes were taken of scenes that were deemed to be relevant for the research questions. After each day, an observation protocol was created based on the field notes. The transcripts presented in the result sections are therefore close representations of the dialogues as noted by R1 in the field notes and should be treated as such.

The observation protocols were coded by three researchers for emerging or underlying CT aspects in a deductive qualitative content analysis approach based on Mayring (2014). The coding scheme for the qualitative content analysis on the observation protocols was based on the CT aspects and cognitive activities given by Kallia et al. (2021), as the framework considered CT aspects explicitly in mathematics courses and linked CT to mathematical modelling. After a first overview of the material, the aspects of *debugging* and *data collection* were added, which can also be found as aspects of CT in Shute et al. (2017). It could be argued that part of analysing data is to collect data beforehand, but we viewed this as its own process, similar to Weintrop et al. (2016).

Table 1 shows the coded CT aspects and our working definitions, which are mostly based on the definitions given in Shute et al. (2017), as Kallia et al. (2021) did not provide a thorough definition.

Considering that modelling is itself a complex process and our research is situated in the context of mathematical modelling, the aspect of modelling in Table 1 was discussed and a similar constraint as in Villa-Ochoa et al. (2022) was applied. The aspect of modelling was reduced to just the formulation of a model, be it a mathematical model or a computational model like a simulation.

Table 1. CT aspects observed in the modelling project

CT aspect	Definition
Data collection (DC)	Collecting or generating data
Data representation (DR)	Changing the representation of data
Data analysis (DA)	Analysing data
Pattern Recognition (PR)	Identification of and search for underlying structures in data, systems or the problem itself
Abstraction (AB)	Identification of the most important information of a system or problem
Decomposition (DE)	Decompose a problem into subproblems and solving the subproblems to inform a solution of the original problem
Algorithms (AL)	Usage and design of algorithms, as well as reflecting on the usage of them
Automation (AU)	Automating a solution
Generalisation (GE)	Generalising from special cases to the general problem and generalising solutions to be transferable to other problems and contexts
Modelling (MO)	Formulation of a Model of a system, problem or process
Evaluation (EV)	Evaluating solutions and strategies
Debugging (DB)	(Systematic) identification of errors (or error sources) and strategies to fix them

Besides the given CT aspects, we also looked for instances of the CT definition given by Cuny et al. (2010, as cited in Wing, 2010). This means scenarios in which the formulation of problems or solutions with an external agent (not necessarily a machine) in mind was present in some way. Furthermore, we also looked into which of the processes depicted in Figure 1 and Figure 2 were observable during a week of modelling.

Student groups and modelling problems

Seven modelling problems were offered, of which five were proposed by the students and two were presented by the researchers.

In the end, group 1 (five male students from grade 9) chose the problem of predicting the winner of the 2024 UEFA European Championship. Group 2 (five female students, two from grade 10 and three from grade 9) decided to analyse the game "Dobble", sometimes also called "Spot it", by the game publisher Asmodee. There are 55 cards in this game, each with 8 symbols on them, where two different cards are matched by exactly one symbol. The goal is to spot the matching symbol on two cards. As such, not all cards are matched with the exact same symbol. The group focused on the mathematics behind designing those cards. This was the only group that changed their modelling problem halfway through in consultation with the teacher. Their second problem regarded the modelling of a certain type of throw in the game “Boules”[2], where you try to throw a ball as close as possible to a target ball while keeping the balls of your opponents away from the target. Group 3 (four male students, one from grade 11, one from grade 10, two from grade 9) searched for an optimal strategy to play the boardgame “Shut-The-Box”. The version presented to the students has ten tiles numbered from 1 to 10. The task as a player is to throw two six-sided dice and add the numbers of both. After this, the player has to eliminate tiles in such a way (a combination of tiles or a single one), that the sum of the eliminated tiles matches the sum of the dice. If a player has no combination of tiles left to match the dice, the game is finished. When playing with multiple people, different evaluation systems can be considered to determine the winner, for example trying to stay as long as possible in the game or having the least number of tiles left in the end.

Results

In the following section, we will report on the modelling process of the three groups during the week. We will highlight the different CT aspects in the modelling process of each group as well as the problems the students had, that relate to those CT aspects.

Predicting the winner of the European Championship (EC)

The group predicting the winner of the UEFA EC 2024 started by researching the schedule of the matches. Two of the students looked at the outcome of past games, while the rest tried to compare data like the number of own goals or ball possession.

However, the group showed problems in selecting a first small set of the most relevant data. This was especially observable when the group started to research the grass quality of the playing field in the then upcoming game of Spain vs. Germany. When asked about the search for this data by researcher R1, a student answered: “We are simply collecting everything and then we see if we can recognize something to predict the game.”

This reflects the CT aspects of *data collection, data analysis*, *abstraction* and *pattern recognition*. The group is collecting data on the played matches with the goal of analysing them for any underlying patterns. Regarding the aspect of abstraction, they seem to fail in reducing the amount of data to a subset of the most relevant information.

At some point, the group also searched online for a “program to create EC predictions”, which led them to a tutorial for creating predictions in Excel, which they quickly rejected. This could also be interpreted as a slight manifestation of the aspect of *automation* as the group tried to find a (already existing) way to automatically solve the given problem.

The problem of reducing the complexity of the given data could also be the reason why the group abandoned their online search for data during the first day and decided to do a survey at their school. The participants of the survey provided demographic information, their prediction of the outcome Spain vs. Germany and their guess of the tournament champion. The participants were also asked to rate their expertise on a four-point scale. The scale here can also be interpreted as a first *modelling* decision. After prompts by the supervisors, they also did a second survey on the matches of Turkey against Austria and Romania against Netherlands as those matches were in the evening of the first day of the project week.

The data from the surveys was collected in Excel. The group worked in the CT aspect of *data representation* when trying to create a diagram of the data following an online tutorial. Researcher R1 asked what they wanted to plot and how, so the group started to discuss the type of diagram needed for each column in the data set. One student prompted to plot everything as a line chart, while another argued for a bar chart with the guesses for the winner of the upcoming match.

When the resulting chart didn't match their expectation, they followed the tutorial multiple times but couldn't identify the error. They then turned to sample data, which was just a table with two columns and some numbers, to experiment with the chart types. This was one of the instances where we saw the aspect of *debugging* being highlighted as the group showed some form of strategy to identify the error by examining the tutorial again and using another data set to test.

As the group wasn't able to use the data from the surveys to their satisfaction, they went back to the data search from the beginning after some discussion with the researchers and teachers. They started identifying a small set of properties they wanted to use for their predictions.

They defined a function $y(x, a, b) = 15 \cdot x \cdot a \cdot b$ to determine the winner of a match or rather a scoring of each team, where x is the performance of the team, a is the market value of the squad in millions divided by ten and b is the number of fans in thousands. Furthermore, they drafted a first calculation of the performance with $x = m + t + p$ with m being the number of wins in the tournament so far, t being the difference in goals in the tournament, and p being the power-ranking at the start of the tournament. Figure 3 shows the values with this model for some of the teams in the tournament.

As this is a first mathematical model for predictions by reducing the amount of data to a small, relevant subset, this clearly can be categorized under the CT aspects of *modelling* and *abstraction*.

We also observed a wish for *automation* again, when one of the students tried to use ChatGPT[1] as a calculator by explaining their evaluation scheme and prompting it to calculate the scorings of the EC pairings. As he didn't specify the year, ChatGPT responded with the wrong matches and the student used multiple prompts to steer the AI to a useful response. This could also be interpreted as *debugging* again.

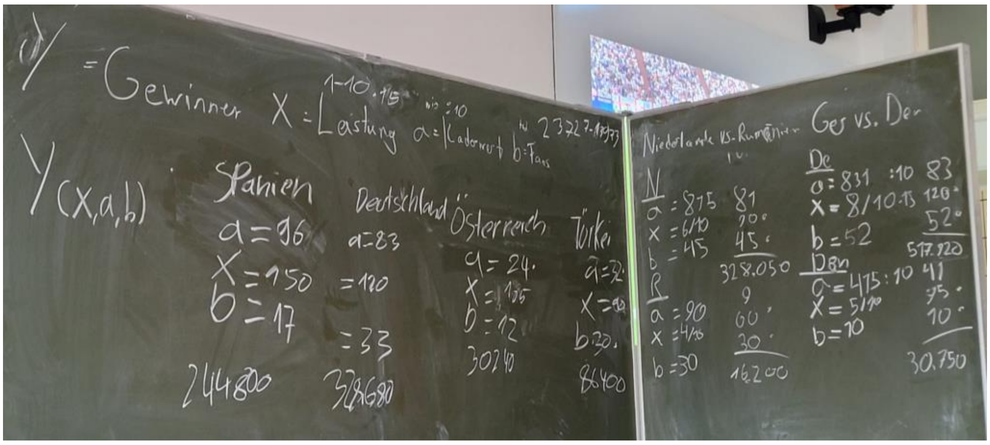

Figure 3. Rating of the different teams according to the model of the students. From left to right: Spain, Germany, Austria, Turkey, Netherlands (N), Romania (R), Germany again (De) and Denmark (Den)

The same student (S) then discussed the model with the supervisors and came up with a weighting for the calculation of the performance, which is also discussed by researcher R2 with the rest of the group later on:

S: We calculate the performance as the number of games won in the tournament divided by the total number of games. We weight that by 2, add the goal difference and divide that by 2. And the power ranking times 5.

R2: What values did you choose that don't come from the data?

S: Times 2, divided by 2 and times 5.

R2: And I would now recommend using the computer if we want to test this with all the games. Then you can quickly change 2, 2 and 5 and everything adjusts. Have you ever worked with Excel?

S: No, we don't know Excel.

This transcript shows, that the students didn't come up with the idea of using the computer themselves as they also weren't confident in working with it besides trying to create some charts.

After that external input, the group started to build an Excel sheet with support of researcher R2 to evaluate their weightings for the performance. They collected the necessary data from various sources they found online and also changed the modelling of the function $y(x, a, b)$ to be a weighted sum of the inputs instead of a product. The researcher assisted in the technical difficulties with Excel, like cell dependencies, while giving small inputs like "What data do we need for your formula?".

After successfully implementing the Excel sheet, they used the results of the matches from all rounds up to that day to adapt their weights. They changed the weighting in the performance formula as well as in the overarching formula to fit most of the matches, but not all of them. One of the falsely predicted games was Turkey versus Austria, but after discussion with the group and the teacher, they declared that the winner of that game seemed to be surprising and so a falsely predicted match would reflect this.

With this, the students went from a mathematical model to a digital representation of the model to alter the weights and compare them to real data. Although this was started by a prompt from researcher R2, this can be seen as an instance, where CT following the definition by Cuny et al. (2010, as cited in Wing, 2010) would benefit the group as they had to transfer their mathematical model to a computer model. This also addresses the aspect of *automation* again. With their way of fitting the weights as well as the discussion about the correctness of the prediction and implications for the real situation the CT aspect of *evaluation* is also imminent.

Dobble

The students started off by playing the game to explore the context. To solve the problem of understanding the design of the cards in the game Dobble, two of the students (A and B) explained their plan to researcher R1:

R1:	What are you doing right now?
A:	We're collecting data, that is, how often which picture occurs and then hope to, um, recognize patterns.
B:	And then we want to simplify how much you need for 10 cards.
A:	Yes, so collecting data, recognizing patterns and then simplifying.

The planning of the students already contained the CT aspects of *pattern recognition*, *data collection,* and *data analysis*. But the idea to simplify the problem and look at a special case was debated during the coding process, as this case of *specialisation* wasn't really present in the CT frameworks and it didn't really fit the working definition of abstraction. We recognised it as a preparing step for *generalisation* as the analysis of the special cases will eventually lead to generalising the structure.

The group counted the occurrences of symbols on the 55 playing cards and *represented* that *data* in tables, discovering a total of 57 different symbols. To analyse how the cards were designed and to see if 57 was the lowest possible number of symbols, they continued their plan and looked at the special cases of two, three and four symbols per card. Figure 4 shows the results for 3 symbols per card in two different representations used by the students.

For the construction itself, they didn't specify constraints, which would also allow one symbol to be matched across all cards. This wouldn't make for a good game, so we assumed that they followed the basic idea of the game and tried to evenly distribute the matched symbols, that is, Symbol A is matched on the same number of cards as Symbol B.

The students first created a list with cards for three symbols per card and then rearranged them into a matrix scheme as seen in Figure 4, which aligns the cards that match on the first symbol in columns except for the first card ABC.

	ABC	
ADE	BDF	CDG
AFG	BEG	CEF

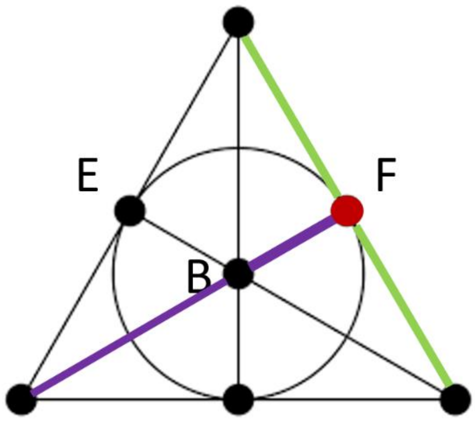

Figure 4. On the left are the seven cards the students found for the case of three symbols per card using the letters A to G as symbols. On the right is a graph representation that the students used when analysing the structure

They started off with ABC and then added cards that shared the symbol A but weren't allowed to share any other symbols, introducing the missing letters D to G. They didn't specify their constraints and couldn't explain, why they didn't continue this scheme with A past the letters F and G. The rest of the columns have the same idea – to match with the first card in only the symbol B or C respectively. For the first row, they kept D as the second letter and chose the remaining one in such a way that it "would fit".

They created a similar matrix for two symbols per card and then *generalized* the idea to y symbols per card. As they recognized that they have y columns but only $y - 1$ rows, this led to the general matrix scheme in Figure 5 and the formula $x = y \cdot (y - 1) + 1$ with x being the total number of symbols needed as well as the total number of cards creatable.

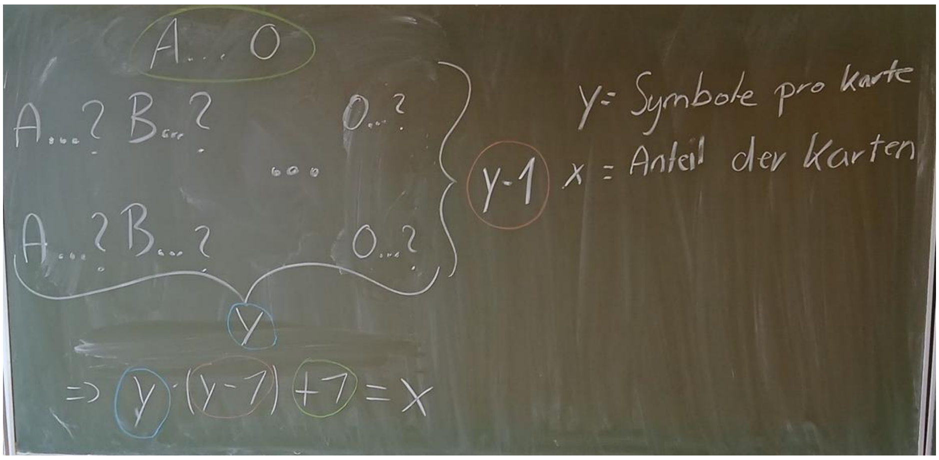

Figure 5. The general matrix structure that the students assumed the cards have for y being the number of symbols per card and x being the total number of cards

They validated their scheme with $y = 4$ and then with $y = 8$ for the original Dobble cards. They recognized that their formula would result in a total number of 57 cards instead of the given 55 cards. To find the two seemingly missing cards, they went through their table of symbol occurrences on the 55 cards and counted each symbol to find the under-represented symbols, revealing once again their implicit constrain of an even distribution of symbols across cards. They then constructed the two missing cards shown in Figure 6 and verified their correctness once again with their table. They recognized an error in the two cards as they didn't have a common symbol and used the table again to determine the missing symbol on the left card.

Figure 6. The two cards "missing" from Dobble with a small error. On the left side there is a snowman symbol missing.

After this, researcher R1 asked them how they created the cards in the matrix or how one could create the cards for other values of y. As hinted, the group explained their creation of cards formally for the first steps (choosing the matrix form and filling the columns with matching first symbols), but then explained that they chose the rest of the symbols "so that they fit".

To answer the question of the researcher regarding a recipe for creating cards, they created the cards for $y = 4$ and tried to recognize patterns in their way of finding fitting symbols. However, this led to identifying some sort of diagonal patterns, which they weren't able to interpret anymore. As they lost the motivation for the game, they consulted with the teacher and continued with modelling Boules.

This process showed multiple different CT aspects. The group used *generalisation* when creating their formula as a mathematical *model* from the analysed special cases. They tried formulating an *algorithm* to create cards with a fixed number of symbols by *recognising patterns* in their way of creating those cards. They failed, however, to provide a clear instruction for others to create those cards, addressing the main idea of CT again.

Boules

After Dobble, the group focused on modelling a throw in a game of Boules in the last one and a half days of the project week.

From video tutorials, they found a throw that lands a few centimetres before an opponent's ball to push it away and roll a bit further. From that video, they determined the ball to start at a height of 1 m and set the target (the opponent's ball) to be at a distance of 12 m.

The teacher T gave them the hint to use a parabola as a fitting model, and as they wanted the ball to land 20 to 30 cm before the target, they modelled the throw as a parabola passing through the points (0,1) and (11.7,0). They then discussed that they were somehow missing the second root of the function to determine the whole function equation. This means they recognized the underdetermined nature of their problem.

Instead of focusing on their original model, one of the students' fathers gave her a physical model for the horizontal throw to use in the modelling at the end of the third day of the week. Her father told her that a non-horizontal throw would need knowledge of sine and cosine, which she didn't have, and so a horizontal throw should suffice.

They used the equation $s_y = \frac{g}{2} \cdot (s_x/v_x)^2$ with $g = 9.81\,\mathrm{m/s}$, s_y as the height, s_x as the distance and v_x as the speed in the x-direction. They solved for v_x and then determined the time t needed for the throw. With the points (0,1) and (11.7,0) from before they determined that the player would have to throw with an initial speed of 23.91 m/s.

Researcher R1 asked the students to convert the result into kilometres per hour, resulting in around 86 km/h. This surprised the students and the supervisors and prompted a discussion about the correctness of the results. They then calculated the speed for a distance of 9 m, resulting in over 60 km/h. Regarding the seemingly high velocities one of the students (A) discussed this with researcher R1.

A:	But the curve is realistic!
R1:	Yes, but for the distance you have to throw with a lot of momentum. That's why you throw an arc.
A:	Makes sense.
R1:	You can test your own throw and measure the speed at which you threw it. Then you'll know how realistic or unrealistic 80 km/h is.

After this, the students went to measure their own throw. They tested if they could throw the regarded distance while trying to maintain a horizontal throw. As they didn't record the throws, they couldn't guarantee their throws to be truly horizontal, but concluded that the results might be achievable.

Besides the *modelling* aspect, the discussion within the group and with the researcher showed the CT aspect of *evaluation* and partly *debugging*. They transferred the results from the mathematical modelling into the original context and evaluated the meaningfulness of the results. As the results seemed to be faulty, they tried to identify possible errors after being prompted to do so by researcher R1.

Shut-The-Box

The group that searched an optimal strategy for Shut-The-Box started off by playing the game to explore the context. After some rounds of playing, they drafted four initial strategies and each student followed one of them.

1. Student A may only flip combinations (so if a 9 is rolled, A may only flip 4 and 5 or 1,2,6, for example, but not the 9 as a single number).
2. Student B may only flip over the sum (a single tile)
3. Student C tries to keep the lowest tiles (1,2,3) as long as possible
4. Student D must alternately choose a combination or flip over the sum (alternately more than one tile or exactly one tile)

The student who can't follow his strategy anymore – even if there would be another legal move – looses. They played ten games looking which strategy allows them to play the most rounds and another ten games to see which strategy allowed them to have the lowest sum of remaining tiles. Based on this data, student B "survived" the longest, while student D had the lowest mean score regarding remaining tiles.

They chose the strategy of student D as a first candidate for an optimal strategy and played a game with that strategy against the teacher. After losing that game, they thought about abandoning the strategy again. The teacher then reminded them, that a single game might not be good evidence for their strategy not working properly.

This first part of their modelling process highlights a lot of different CT aspects. The formulation of strategies can be seen as an *algorithm*, that the students followed while playing and as such also addresses the basic idea of CT. The *data collection* during playing led them through *analysis* for picking a strategy as a candidate to be *evaluated* in the real context, that is, playing a game against an unknown or random opponent.

A next strategy that the group tried was based on an evaluation of the importance of some tiles. They looked at what tiles can be used in the most combinations or rather what tiles they deemed useful. It couldn't be observed if that heuristic was already based on concrete calculations of probabilities or rather experience from playing as this strategy was fostered after the first project day. In either case, it can be argued that this was an instance of *pattern recognition*. Their strategy was to preferably flip the tiles 10, 9 and 8, then the tiles 1, 6 and 7 and keep the tiles 2, 3, 4, 5 as long as possible, although they didn't specify how to act if combinations contained tiles of different categories.

This new strategy was than evaluated against the trainee teacher. One of the students played multiple games against the teacher and conferred with another student after each roll what to flip. That second student had notes on his sheet that the first one looked at to determine his next move. This was another prominent example of the definition of CT by

Cuny et al. (2010 as cited in Wing, 2010), as one of the students acted as the external agent that only executes the strategy that the other student formulated for him.

At some point the trainee teacher prompted the group in the direction of tree diagrams: “You have to draw the paths. See what you can do after rolling the dice.” This led the group to work on a tree diagram of the game to determine the optimal choices for each roll.

The group proceeded to go through every possible dice roll in the first round and looked at the possible combinations of tiles (including flipping a single tile). They evaluated those paths by the probability to lose with the next dice roll. For example, if one would role a 4 in the first round, the possible combinations would be 1 and 3 or 4. Flipping the 1 and 3 would mean that the game would be over if the second dice roll would be a 3 as this couldn’t be represented anymore, making the probability to lose be equal to 2/36.

For each dice roll, they chose the minimal path regarding that probability or allowed multiple ones if they were equally good. The results for their strategy can be seen in Table 2. They continued to calculate the second level of their implicit probability tree diagram by ignoring suboptimal paths on the first level and calculated the same loss probability for the remaining paths in dependence of the rolls on the first level. This resulted in the matrix shown in Figure 7.

Table 2. Optimal choices (e.g., tiles to flip) for the first round in Shut-The-Box

Dice Roll	2	3	4	5	6	7	8	9	10	11	12
Optimal Choice	2	3	4	5	6	7	8	9	10	1+10	5+7

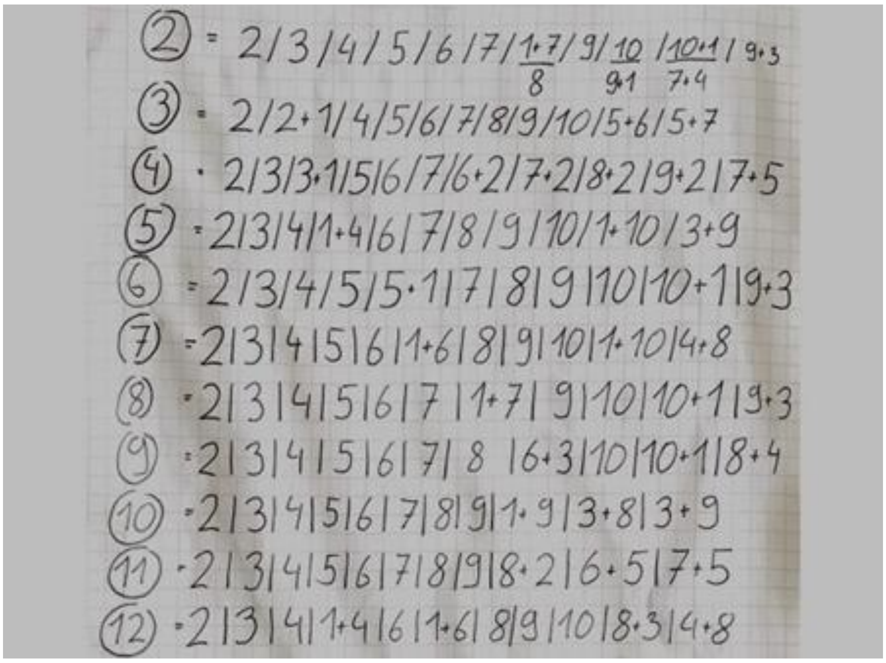

Figure 7. Optimal choices for the second round of Shut-The-Box. To the left is the dice roll in the first round. The positions after the equal sign are meant to represent the roll in the second round, e.g., first position means a 2 was rolled in the second round, last position means a 12.

During the work for the strategy of the second round, researcher R1 together with the trainee teacher TT asked students A and B what their plan for the third and following rounds of Shut-The-Box was.

R1:	And then you'll do the third branch?
TT:	No, no, that would be too much... wouldn't it?
A:	Well, we want a complete strategy.
B:	We can continue with an old strategy afterwards.
TT:	Yes, and that should already be improved by the first two rounds now.

The input of student B was then taken and they combined the probability tree for the first two rounds with the heuristic strategy of flipping 6 through 10 and 1 first if possible after the second round. They evaluated this strategy first against researcher R1 before deciding they would need a truly random opponent. The trainee teacher TT then started to write down all possible tile combinations for all dice rolls, so the students could simulate a random opponent with a random number generator. From 100 played games their strategy only lost in 3 games and 17 ended in a draw.

The process for the final strategy was again very rich in CT aspects. Only looking at the first level and trying to optimize it, greatly depicts *decomposition* as a CT aspect, because they identified the decisions in each round to be subproblems of the whole strategy and used those subproblems to inform the overall strategy. Pruning certain paths in this process that aren't optimal for having a higher loss probability does not only save time in computing the whole probability, but also depicts the aspect of *abstraction* again in reducing the complexity of the (calculation) problem. The need for and the simulation of a random opponent were not only part of the *evaluation* process but can – in conjunction with the evaluation over a hundred games, the termination criterion when the game has to end, and the more or less precise formulation of a strategy – be interpreted as a form of a computational *model*.

Discussion and conclusions

This study showed, that the CT aspects listed in Table 1 can be observed during modelling projects to a greater or lesser extent. Furthermore, we identified opportunities, where fostering CT could benefit the mathematical modelling.

Regarding our first research question, we identified different CT aspects for each project as shown in Table 3. As the examples given by Ang (2021) suggested, the observed CT aspects seemed to depend on the type of modelling project For example, the aspect of decomposition was only observed in the Shut-The-Box project, while generalisation was only observed with the Dobble group.

Table 3. Identified CT aspects in the modelling projects. For abbreviations see Table 1

	DC	DR	DA	PR	AB	DE	AL	AU	GE	MO	EV	DB
Predicting the EC Winner	X	X	X	X	X			(X)		X	X	X
Dobble	X	X	X	X			X		X		X	
Boule										X	X	X
Shut-The-Box	X		X	X	X	X	X				X	

The prediction of the EC 2024 champion demanded mostly constructing a prediction model based on available data on the matter and can therefore be compared to the "from data to model" example by Ang (2021). Consequently, the students showed mostly aspects of working with data and tried to use pattern recognition to derive a model.

Shut-The-Box as a project, on the other hand, was more focused on understanding the game itself and analysing each round to derive an optimal strategy, similar to the "from process to model" example by Ang (2021). Although the project allowed for data collection and analysis, it lends itself more into problem decomposition (by dissecting the game's individual rounds) as well as algorithms and the formulation of the solution for an external agent by providing an optimal strategy.

Compared to the results of Villa-Ochoa et al. (2022), the observed modelling projects weren't created with a focus on digital tools, and in fact, the groups working on Dobble and Shut-The-Box worked unplugged most of the time. As mathematical modelling is itself a problem-solving process, we can also compare this to the framework of Kalelioğlu et al. (2016), who understand CT to also be a problem-solving process and associate the CT aspects from Table 1 with different steps in a problem-solving process. Although some prompts by the researchers and teachers led to a stronger focus on certain aspects, we thus conclude that mathematical modelling and CT as problem-solving process are very similar in nature and share a lot of common aspects.

To address our second research question, the observation also revealed opportunities for CT to support mathematical modelling.

Consider the first group that struggled with selecting relevant data for the predictions of the EC winner. This indicates a need for abstraction (in the sense of identifying relevant information) to reduce the complexity of the data. The second group struggled with formulating their scheme of generating cards in a way that the researchers and teachers could reproduce it, which directly addresses the formulation of solutions for external agents. Furthermore, the third group could have benefited from automation for a lot of their work (simulating the strategy a hundred times, calculating the paths in the decision tree). These three examples can be seen as possible ways for CT to positively influence the mathematical modelling.

Instead of common aspects, that CT could support, we can also focus on the differences recognized between the mathematical modelling process and the CT aspects. We recognized some sort of specialisation process in the Dobble group as a preparing step for generalisation, that didn't fit any aspect. The examination of special cases is a classic mathematical problem-solving strategy (Schoenfeld, 1987) and wasn't present in the reviewed CT frameworks. This might be a unique aspect of mathematical problem-solving but provides an opportunity to apply generalisation and pattern recognition and could help foster CT in that matter.

On the other hand we can focus on the CT definition by Cuny et al. (2010, as cited in Wing, 2010). Fostering CT in the sense of formulating solutions for external agents could not only help in regards of modelling with digital tools and constructing a digital model, but also allow for a more product-oriented modelling.

Lastly, we also want to highlight the aspect of debugging as a way for CT to support mathematical modelling. We observed instances where a thorough systematic error identification is beneficial, for example when the modelled throw for boule yielded a surprisingly high velocity and the students tried to identify possible errors. As Weintrop et al. (2016) recognized, the practices of trouble shooting and debugging are important in the STEM fields, where researchers encounter unexpected outcomes frequently. For mathematical modelling, evaluation of a solution in the real context is an essential part and evaluation was present in all projects (see Table 3). The students may be confronted with problems when applying mathematical solutions to the reality, revealing an opportunity to benefit from debugging or troubleshooting strategies.

This study aims to add to the empirical research on the interconnection of mathematical modelling and CT, but only provided observations on CT aspects in one modelling project during a school week. Further research is needed to expand on how CT is present during mathematical modelling, the impact of CT on the modelling process and how modelling projects can be designed to foster CT, comparing, for example, product-orientation and the deployment of digital tools or programming.

Notes

[1] The group used the free version of ChatGPT available under https://chatgpt.com/ in the version of the first week of July 2024.

[2] Although the students never clarified it, it can be assumed that they focused on Pétanque (https://en.wikipedia.org/wiki/Pétanque).

References

Aebli, H. (2019). Zwölf Grundformen des Lehrens: Eine allgemeine Didaktik auf psychologischer Grundlage: Medien und Inhalte didaktischer Kommunikation, der Lernzyklus (15th ed.). Klett-Cotta.

Ang, K. C. (2021). Computational Thinking and Mathematical Modelling. In F. K. S. Leung, G. A. Stillman, G. Kaiser, & K. L. Wong (Eds.), *Mathematical Modelling Education in East and West* (pp. 19–34). Springer. https://doi.org/10.1007/978-3-030-66996-6_2

Blum, W., & Leiß, D. (2007). 5.1—How do Students and Teachers Deal with Modelling Problems? In C. Haines, P. Galbraith, W. Blum, & S. Khan (Eds.), *Mathematical Modelling* (pp. 222–231). Woodhead Publishing. https://doi.org/10.1533/9780857099419.5.221

Bock, W., & Bracke, M. (2015). Applied School Mathematics—Made in Kaiserslautern. In H. Neunzert & D. Prätzel-Wolters (Eds.), *Currents in Industrial Mathematics: From Concepts to Research to Education* (pp. 403–437). Springer. https://doi.org/10.1007/978-3-662-48258-2_11

Bock, W., Bracke, M., & Capraro, P. (2017). Product orientation in modeling tasks. In T. Dooley & G. Gueudet (Eds.), *Proceedings of CERME 10*. Dublin, Ireland. DCU Institute of Education and ERME. https://hal.science/hal-01933444

Borromeo Ferri, R. (2007). Modelling problems from a cognitive perspective. In C. Haines, P. Galbraith, W. Blum, & S. Khan (Eds.), *Mathematical Modelling* (pp. 260–270). Woodhead Publishing. https://doi.org/10.1533/9780857099419.5.260

Dong, Y., Catete, V., Jocius, R., Lytle, N., Barnes, T., Albert, J., Joshi, D., Robinson, R., & Andrews, A. (2019). PRADA: A practical model for integrating computational thinking in K-12 Education. *Proceedings of the 50th ACM Technical Symposium on Computer Science Education* (pp. 906–912). https://doi.org/10.1145/3287324.3287431

Greefrath, G. (2011). Using Technologies: New Possibilities of Teaching and Learning Modelling – Overview. In G. Kaiser, W. Blum, R. Borromeo Ferri, & G. Stillman (Eds.), *Trends in Teaching and Learning of Mathematical Modelling* (Vol. 1, pp. 301–304). Springer. https://doi.org/10.1007/978-94-007-0910-2_30

Greefrath, G., & Vorhölter, K. (2016). Teaching and learning mathematical modelling: approaches and developments from German speaking countries. In G. Greefrath & K. Vorhölter (Eds.), *Teaching and Learning Mathematical Modelling: Approaches and Developments from German Speaking Countries* (pp. 1–42). Springer. https://doi.org/10.1007/978-3-319-45004-9_1

Kalelioğlu, F., Gülbahar, Y., & Kukul, V. (2016). A framework for computational thinking based on a systematic research review. *Baltic Journal of Modern Computing*, *4*(3), 583–596.

Kallia, M., van Borkulo, S. P., Drijvers, P., Barendsen, E., & Tolboom, J. (2021). Characterising computational thinking in mathematics education: A literature-informed Delphi study. *Research in Mathematics Education*, *23*(2), 159–187. https://doi.org/10.1080/14794802.2020.1852104

Mayring, P. (2014). *Qualitative content analysis: Theoretical foundation, basic procedures and software solution*. Klagenfurt. https://nbn-resolving.org/urn:nbn:de:0168-ssoar-395173

Schoenfeld, A. H. (1987). Pólya, problem solving, and education. *Mathematics Magazine*, *60*(5), 283–291. https://doi.org/10.1080/0025570X.1987.11977325

Selby, C., & Woollard, J. (2013). *Computational thinking: The developing definition* [Monograph]. University of Southampton. https://eprints.soton.ac.uk/356481/

Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational Research Review*, *22*, 142–158. https://doi.org/10.1016/j.edurev.2017.09.003

Teck, L. K., Abdullah, A. H., Mokhtar, M., Zainuddin, S. A. H. S., & Rahman, S. N. S. A. (2023). The development of computational thinking and mathematics problem solving skills through mathematics modelling activities. *Sains Humanika*, *15*(2), Article 2. https://doi.org/10.11113/sh.v15n2.1987

Villa-Ochoa, J. A., Carmona-Mesa, J. A., Quiroz-Vallejo, D. A., Castrillon-Yepes, A., & Farsani, D. (2022). Computational thinking in mathematical modeling projects. A case study with future mathematics teachers. In Á. Rocha, C. Ferrás, A. Méndez Porras, & E. Jimenez Delgado (Eds.), *Information Technology and Systems* (Vol. 414, pp. 460–470). Springer. https://doi.org/10.1007/978-3-030-96293-7_38

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. *Journal of Science Education and Technology*, *25*(1), 127–147. https://doi.org/10.1007/s10956-015-9581-5

Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, *49*(3), 33–35. https://doi.org/10.1145/1118178.1118215

Wing, J. M. (2010). *Computational Thinking: What and Why?.* https://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf

Ye, H., Liang, B., Ng, O.-L., & Chai, C. S. (2023). Integration of computational thinking in K-12 mathematics education: A systematic review on CT-based mathematics instruction and student learning. *International Journal of STEM Education*, *10*(1), 3. https://doi.org/10.1186/s40594-023-00396-w