Effective orchestration features of a project-based approach to learning programming for mathematics investigation

Laura Broley[1], Eunice Ablorh[1], Chantal Buteau[1], Joyce Mgombelo[1], and Eric Muller[1]

[1]Brock University in St. Catharines, Department of Mathematics and Statistics, Canada, cbuteau@brocku.ca

In this exploratory study, we examine the teaching of programming for mathematics investigation in an undergraduate project-based learning environment. Using instrumental orchestration as our theoretical framework, we explore the orchestration features that students consider to be the most effective for supporting their learning. A qualitative analysis of 43 students' questionnaire responses led to the identification of such features, regrouped in 5 main themes (help and support, organization of the course, instructor interventions, instructor characteristics, and class atmosphere). Results suggest that students recognize the need for their instrumental geneses to be steered and highlight the importance of individualized interventions and a supportive learning environment.

Keywords: Teachers' and students' practices at university level, digital and other resources in university mathematics education, instrumental orchestration, project-based learning, programming

INTRODUCTION

Research has documented many affordances of programming for student learning of mathematics at the university level. Studies have shown how programming can support students' activity and understanding in specific areas, such as calculus, abstract algebra, combinatorics, statistics, and probability (Buteau et al., in press). It has also been argued that programming activities can engage students in crucial mathematical disciplinary practices (Broley et al., 2017).

In their recent call to the international community of research on university mathematics education, Lockwood and Mørken (2021) argue that much more needs to be investigated concerning the integration of programming, including effective instructional interventions, teaching practices, and didactic models:

> there are already varying models around the world for how computing is being integrated into mathematics classes and programs, and we see opportunities for systematically studying different ways for this integration to occur. […] the RUME community can explore what kinds of programs are effective and why. (p. 6)

This paper addresses the above area of need by exploring effective features of a project-based learning (PBL) model for integrating programming in university mathematics. As an integration approach, PBL engages students in actively constructing knowledge by working through an inquiry process that is structured by authentic and complex tasks (Shpeizer, 2019). PBL has been implemented in various academic institutions and fields, with the literature generally arguing for its positive effects on learning outcomes (Thomas, 2000). In the context of programming-based math education, PBL has deep roots: it has been over 50 years since Seymour Papert pointed to the potential of engaging students in meaningful projects in which they construct (i.e., program) computer environments to learn and do mathematics like mathematicians do. Papert (1980) emphasized that such a "constructionist" approach requires a fundamental change to "traditional" teaching methods: from presenting mathematical ideas to students, to creating didactic conditions that foster students' own pursuit of ideas.

In the current study, we explore students' perspectives on what those conditions might be: in particular, how university math instructors may support students' learning to use programming for math investigations in a project-based approach. Our exploration takes place within a "natural constructionist environment" (Buteau et al., 2015) that has been implemented for 20 years at Brock University (Canada) and as part of a larger 5-year (2017-2022) iterative design research that uses that environment to study how students learn to use programming in authentic pure or applied mathematics projects, if and how that use is sustained over time, and how instructors support that learning.

THEORETICAL FRAMEWORK

In our larger research, we have shown that the instrumental approach (Artigue, 2002) can be useful in studying the teaching and learning of using programming for math investigation projects at the university level. In Gueudet et al. (2020), we used the notion of instrumental genesis to better understand how students transform a programming language (an artefact) into a math investigation tool (an instrument for accomplishing the goals involved in math investigation projects) by developing instrumented action schemes. In this study, we are interested in looking at features of instructional practice that can support this instrumental genesis. Following Buteau et al. (in press), we frame our study using the notion of instrumental orchestration.

Trouche (2004) introduced the notion of instrumental orchestration to highlight the necessity of an external steering of students' individual and collective geneses and to describe the instructional decisions and actions involved. More precisely,

> an *instrumental orchestration* is defined as the teacher's intentional and systematic organisation and use of the various artefacts available in a—in this case computerised—learning environment in a given mathematical task situation, in order to guide students' instrumental genesis (Trouche, 2004). (Drijvers et al., 2010, p. 214)

Building on the work of Trouche (2004), who introduced "didactical configuration" and "exploitation mode" as two key components of an instrumental orchestration, Drijvers et al. (2010) introduced a third component, the "didactical performance," and defined the three components as follows:

> A *didactical configuration* is an arrangement of artefacts in the environment, or, in other words, a configuration of the teaching setting and the artefacts involved in it. …
>
> An *exploitation mode* is the way the teacher decides to exploit a didactical configuration for the benefit of his or her didactical intentions. This includes decisions on the way a task

> is introduced and worked through, on the possible roles of the artefacts to be played, and on the schemes and techniques to be developed and established by the students. …
>
> A *didactical performance* involves the ad hoc decisions taken while teaching on how to actually perform in the chosen didactic configuration and exploitation mode: what question to pose now, how to do justice to (or to set aside) any particular student input, how to deal with an unexpected aspect of the mathematical task or the technological tool, or other emerging goals. (p. 215)

Studies in math education have employed the notion of instrumental orchestration with various technologies (graphing calculators, dynamic geometry software, spreadsheets, …), mainly at the school level. For instance, Drijvers et al. (2010) used the notion to study the use of applets with eighth-grade students and began cataloguing different orchestration types (e.g., *Technical-demo* and *Explain-the-screen*). At the time of writing this paper, we are unaware of other research on the orchestration of programming in investigation projects, i.e., in a PBL approach (Buteau et al., in press).

PBL is an instructional model that organizes learning around projects. According to Thomas (2000), such projects, "as well as the activities, products, and performances that occupy [students'] time, must be orchestrated in the service of an important intellectual purpose" (p. 3). In terms of teaching, PBL redefines the traditional role of the teacher as one of collating sources, facilitating thinking, and inspiring students to impact the world, with class time used to probe students about their sense-making and skills acquisition (Prince & Felder, 2007). This changing role of the teacher is seen as a key challenge in successfully implementing PBL in the classroom (Shpeizer, 2019).

In light of the framework outlined above, we pose the following research question: What are the features of an effective orchestration of programming for mathematics investigation in a project-based approach?

METHODOLOGY

Our study is situated in the context of three semester-long project-based math courses (MICA I-II-III) offered at Brock University. In these courses, math majors and co-majors (including future math teachers) learn to use programming to investigate math concepts, theorems, conjectures, and real-world applications (Buteau et al., 2015).

Our past work has examined the instrumental orchestration of programming in MICA courses primarily from the institution's and instructors' points of view (e.g., Buteau et al., in press). Using institutional documents and interviews with an experienced MICA instructor (who was also involved in the development of the MICA courses), key orchestration features were highlighted and discussed: e.g., the teaching format (part of the didactic configuration), which includes weekly 2-hour lectures (where the instructor introduces the math) and 2-hour labs (where the students work on projects); the assessment structure (part of the exploitation mode), the heart of which (70-80% of students' grades) are 4-5 investigation projects developed by the instructor; and the kind of help (both individual and collective) given to students to support their work in labs (part of the didactic performance). In this study, we complement this past work by examining key features of the orchestration from students' points of view.

Part of years 2-4 of our larger research included students being invited to voluntarily respond to questionnaires administered before (pre) and after (post) each MICA course. There were various sections featured in the questionnaire, including demographics, students' perceptions of the importance of programming, confidence in programming, etc. The question we consider in this study is taken from the post-questionnaire, where students in years 3-4 were asked to indicate and elaborate on (by writing a text) what their instructors or teaching assistants (TAs) did that had the most impact on their assignment work or learning in any of the MICA courses they had taken so far.

In this study, we analyzed the responses of 43 MICA students from years 3-4 (2019-20) of the research (25 from MICA I, 5 from MICA II, and 13 from MICA III). Responses were coded independently by two coders and then codes were consolidated. Codes were then grouped into themes and sub-themes using an emerging theme approach. Finally, we reflected on our results using our theoretical framework. We note that some participants' responses were coded with several codes, possibly within more than one sub-theme or theme. Also, our findings are representative of every participant's voice: even if a sub-theme emerged from only one participant response, we considered it valuable to include it in our results since, in this initial study, we aim to identify possible features of an effective orchestration (from students' points of view). Given the relatively small sample size and voluntary participation, we also note that participants are not necessarily representative of all MICA students.

RESULTS: MOST IMPACTFUL ORCHESTRATION FEATURES

After coding and consolidation, 5 main themes and 16 sub-themes emerged, as synthesized in Tables 1-5. These themes characterize orchestration features that, according to students, had the most impact on their assignment work and learning. Given the context in which we work (as described above), we interpret these as being features that may contribute to an effective orchestration of programming for mathematics investigation in a project-based approach. In the following, we describe the themes and sub-themes, including several illustrative quotes from students.

Help and support

Many students indicated the most impactful thing the instructors or TAs did included providing help and support (see Table 1, with sub-themes and descriptive quotes). Some students spoke in a general sense, simply highlighting that they were given (lots of) help, while some (also) specified the part of the course on which they received the help or support (assignments, programming, and/or mathematics).

When mentioning "help with programming," some students specified further the part of the programming process with which they received help (e.g., "I was able to … get some help *debugging some minor errors*"). Other students emphasized impactful approaches used by instructors or TAs to provide support for programming, including:

- *giving one-on-one help with programming* ("When I would come for help, the TAs would look through my program and then let me know where the error may be and would hint at how to fix it");
- *providing additional coding information to the class, on-demand, just-in-time* ("Some help hours TAs or instructors realized everyone was having the same issue and would provide some extra coding information to incorporate in the assignment that would help everyone understand the project further");
- *displaying and explaining example codes* (e.g., "It was very helpful when example code was displayed on the screen and explained. It helped to see how it could actually be used"); and
- *making example codes available* ("I was able to download codes from lectures and play around with them and change things so that I was able to better understand how the different codes worked").

Sub-Themes	**Descriptive Quotes**
General help	Helping me when I was stuck.
Help on assignments	With the help of the instructors/TAs I was able to understand the assignments better.
Help with programming	… they provided help for me when I was stuck and didn't know what to do next while I was programming for an assignment.
Help with math	it is also helpful when … [they] know how to help you when you have a problem with both the mathematics and the programming.

Table 1: Sub-themes and descriptive quotes for the Help and Support theme

We interpret this theme as an implicit recognition by students that their instrumental genesis needs to be steered. The "help with programming" sub-theme highlights students' views about different ways in which an effective steering may occur, including certain orchestration types: e.g., *Discuss-the-screen* is connected to *displaying and explaining example codes*.

Organization of the course

Some students' responses pointed to features associated with the format of the course; in particular, the different modes by which help was made available to them by the instructors or TAs, and the general organization of course content (see Table 2). Students explicitly mentioned several ways in which help was made available to them outside lectures, including labs, help hours, emails, and (extra) office hours.

In terms of instrumental orchestration, we interpret this theme as mainly describing elements of the didactic configuration and exploitation mode. The various modes of making help available extends the opportunities for students to experience the same kinds of interventions as in a *Work-and-walk-by* (Drijvers, 2012) classroom orchestration. This suggests that students' instrumental geneses require readily available individualized help/interaction with a mentor.

Sub-Themes	Descriptive Quotes
Modes of making help available	I really like the format of *labs*; it gave me a chance to talk to the instructor or TA about my assignment and also to get help if I needed it. Proving *help hours* so I didn't have to struggle on my own. I was able to access help easily during *help hours* and reach my instructor through *email*. He gave very helpful tips even through *email.* They added *extra office hours* for assignments when needed…
Organization of course material	…the [online] course work was all nicely in one spot. It was easy to find and easy to understand what had to be done. The assignments were also nicely broken up into questions and parts of questions. It was just really organized, and I appreciate that.

Table 2: Sub-themes and descriptive quotes for the Organization of the Course theme

Instructor characteristics

Another theme that emerged expresses students' views on certain impactful "ways of being" of instructors or TAs: according to students, they were not only knowledgeable, but also available, kind, and supportive (see Table 3).

Sub-Themes	Descriptive Quotes
Available	Their anytime response to our doubts irrespective of their schedules.
Knowledgeable	It is also helpful when they're knowledgeable and actually know what they're talking about and know how to help you…
Kind	He was so incredibly kind.
Supportive	Programming was brand new to me in [MICA I] and it was extremely intimidating (still often is) so it is nice to have helpful and supportive instructors and TAs.

Table 3: Sub-themes and descriptive quotes for the Instructor Characteristics theme

Some of these instructor characteristics could be interpreted as linked to the exploitation mode. In particular, being available and supportive towards students aligns with the expectation that students will need a lot of individualized support. Instructors may plan these "ways of being" in order to offer this support.

Instructor interventions

Some students also mentioned specific and effective (according to them) ways in which instructors or TAs facilitate their work or learning (See Table 4).

Sub-Themes	Descriptive Quotes
Ways of providing help	My TA would break the projects down for me to a level that I would understand. Which allowed me to be successful in the course.
Feedback on assignments	The TA and professor would let me know where I lost marks which made me improve those things for future assignments.
Intervention for high-achieving students	I found that I was able to complete most assignments rather quickly. As such, the prof. would often give me ideas that would be difficult to implement and allowed me to brainstorm how I would implement these tasks … These difficult tasks allowed me to learn concepts and think outside the box far more than if I was to just complete assignments as they are written.

Table 4: Sub-themes and descriptive quotes for the Instructor Interventions theme

The "ways of providing help" sub-theme was associated to a rich collection of responses, which specified different methods that instructors or TAs used while offering them the help they needed. In addition to *breaking down content to a student's level of understanding* (exemplified in Table 4), students described the following impactful ways of providing help:

- *re-explaining multiple times when needed* ("taking the time to go through it with me multiple times when I didn't understand something");
- *explaining what a student is doing wrong and why* ("they would inform me what I was doing wrong and WHY it was wrong. By doing this, I can grow and learn from the experience");
- *providing a full explanation* (*"*The professor would always fully explain the issue rather then giving a half-hearted cryptic help response. Sometimes teachers try to give a little hint in hopes you'll figure it out yourself. But I wouldn't so getting a lesson about what went wrong is more helpful");
- *guiding towards rather than telling the answer* (e.g., "They never said 'figure it out' but they helped guide us to the correct answer without fully saying 'here it is'"); and finally
- *giving meaningful answers* ("they didn't give vague answers, they truly did help you").

We interpret this theme as describing elements of the didactic performance. The "ways of providing help" sub-theme suggests effective (from students' points of view) individualized interactions that may occur, for example, in the *Work-and-walk-by* orchestration during labs.

Class atmosphere

Finally, some students' responses pointed to the kind of environment created by the instructors or TAs to foster students' learning (see Table 5). Students indicated that the class was a space where they felt safe to ask questions, encouraged to make contributions, and able to work on their own if they wanted.

Sub-Themes	Descriptive Quotes
Safe to ask questions	Everyone in the class knew they could ask [the TA or instructor] any question and receive a helpful and cheerful answer. They created an environment where students weren't afraid to ask questions and that is what was most needed to fully understand the content.
Encouraged to contribute	Instructor encouraged us to attempt to formulate our own theorems before resorting to finding a pre-existing one to study. Instructor took an interest in the fact that I had recently heard of steps towards cracking the Collatz conjecture.
Can work on own if want	I didn't ask for a lot of help on assignments since I enjoy working things out myself and any time that I got stuck, I was able to get unstuck again.

Table 5: Sub-themes and descriptive quotes for the Class Atmosphere theme

We interpret this theme as being part of an instructor's exploitation mode, which includes how they will present tasks, how students will work through the tasks, and the atmosphere that will surround that work. The fact that students need individualized support as they work through tasks appears to provide guiding principles to how the instructor creates the atmosphere (by explicitly inviting students to ask questions, responding to students' questions in a "kind and cheerful" manner, etc.).

DISCUSSION

In this paper, we answer the call from Lockwood and Mørken (2021) for more research about effective instructional models for integrating computing in university math education by exploring features of an effective orchestration of programming for mathematics investigation in a project-based approach. Our study contributes to the literature involving the instrumental orchestration frame by (1) using it to examine PBL, a particular instructional approach that has not yet been examined using the frame (Buteau et al., in press); and (2) identifying most impactful features from students' points of view, which, to our knowledge, has not been done. Our study also contributes to literature on PBL, in which there is a lack of studies specifying the required teacher's role for a successful implementation (Shpeizer, 2019). Our results align with some key elements that have been identified: e.g., the creation of a safe learning environment, the encouragement of students (including to ask questions), and the importance of formative and summative assessment (Pan et al., 2021).

In this study, we analyzed questionnaire responses from 43 students using an emerging theme approach, which led to 16 sub-themes organized by 5 main themes: help and support, organization of the course, instructor interventions, instructor characteristics and class atmosphere. These themes characterize orchestration features that, according to students, had the most impact on their work and learning and, therefore, may be inferred to contribute to an effective orchestration of programming for math investigation in a PBL approach. Interpreting the themes using the instrumental orchestration frame points to features that were not made explicit in the description of its three components (Drijvers et al., 2010): e.g., "class atmosphere" as a feature of the "exploitation mode," or "feedback on assignments" (ad hoc decisions occurring outside the classroom) as a feature of the "didactic performance." Responses from students also highlight a need, specific to the university level, of considering TAs as additional players, who have orchestrations of their own, which are shaped by and situated within an instructor's orchestration. Our interpretation of the identified themes also suggests some elements that may be specific to a PBL approach (in comparison to a "traditional" one): e.g., the "organization of the course" theme suggests that for a PBL instructor, it may not always be helpful to instruct the entire class based on one person's issue (they may expect students to require individualized support). In relation to this, we propose a new orchestration type (Drijvers, 2012): *Work-and-reach-out- when-needed*.

This initial exploratory study sets the ground for future work examining more deeply the impact (or effectiveness) of the different features we have identified. Some students elaborated on their perception of the impact of instructors' or TAs' actions on their learning or completion of projects: e.g., with respect to the creation of a learning environment where it is safe to ask questions, one student said that "that is what was most needed to fully understand the content." Future work could look more systematically at the impacts of different orchestration features on students' learning and project work. This could inform recommendations for practice, especially at a time when computing is becoming more integrated across mathematics education.

ACKNOWLEDGEMENTS

This work is funded by Social Sciences and Humanities Research Council of Canada (#435-2017-0367) and received ethics clearance at Brock University (REB #17-088).

REFERENCES

Artigue, M. (2002). Learning mathematics in a CAS environment: The genesis of a reflection about instrumentation and the dialectics between technical and conceptual work. *International Journal of Computers for Mathematical Learning, 7*, 245–274.

Broley, L., Buteau, C., & Muller, E. (2017). (Legitimate peripheral) computational thinking in mathematics. In T. Dooley, & G. Gueudet (Eds.), *Proceedings of the*

Tenth Congress of the European Society for Research in Mathematics Education (pp. 2515–2523). DCU Institute of Education and ERME.

Buteau, C., Muller, E., & Marshall, N. (2015). When a university mathematics department adopted core mathematics courses of an unintentionally constructionist nature: Really? *Digital Experiences in Mathematics Education, 1*(2/3), 133–155.

Buteau, C., Muller, E., Santacruz Rodriguez, M., Mgombelo, J., Sacristan, A. I. & Gueudet, G. (in press). Instrumental orchestration of using programming for authentic mathematics investigation projects. In A. Clark-Wilson, O. Robutti, & N. Sinclair (Eds.), *The Mathematics Teacher in the Digital Era (2nd ed.)*. Springer.

Drijvers, P. (2012). Teachers transforming resources into orchestrations. In G. Gueudet, B. Pepin, & L. Trouche (Eds.), *From text to 'lived' resources: Mathematics curriculum materials and teacher development* (pp. 265–281). Dordrecht: Springer.

Drijvers, P., Doorman, M., Boon, P., Reed, H., & Gravemeijer, K. (2010). The teacher and the tool: Instrumental orchestrations in the technology-rich mathematics classroom. *Educational Studies in Mathematics, 75*(2), 213–234.

Gueudet, G., Buteau, C., Muller, E., Mgombelo, J., & Sacristán, A. (2020). Programming as an artefact: What do we learn about university students' activity? In T. Hausberger, M. Bosch, & F. Chellougui (Eds.) *Proceedings of INDRUM 2020 Third Conference of the International Network for Didactic Research in University Mathematics* (pp. 443–452). University of Carthage and INDRUM.

Lockwood, E., & Mørken, K. (2021). A call for research that explores relationships between computing and mathematical thinking and activity in RUME. *International Journal of Research in Undergraduate Mathematics Education,* 1–13.

Pan, G., Seow, P. S., Shankararaman, V., & Koh, K. (2021). An exploration into key roles in making project-based learning happen: Insights from a case study of a university. *Journal of International Education in Business*, *14*(1), 109–129.

Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic Books.

Prince, M., & Felder, R. (2007). The many faces of inductive teaching and learning. *Journal of College Science Teaching, 36*(5), 14–20.

Shpeizer, R. (2019). Towards a successful integration of project-based learning in higher education: Challenges, technologies and methods of implementation. *Universal Journal of Educational Research, 7*(8), 1765–1771.

Thomas, J. W. (2000). *A review of research on project-based learning*. Autodesk Foundation. http://www.bobpearlman.org/BestPractices/PBL_Research.pdf.

Trouche, L. (2004). Managing the complexity of human/machine interactions in computerized learning environments: Guiding students' command process through instrumental orchestrations. *International Journal of Computers for Mathematical Learning, 9*(3), 281–307.