

Scientific practices in minimally working programs

Dan P. Oleynik and Paul W. Irving

Department of Physics and Astronomy, Michigan State University, 567 Wilson Rd, East Lansing, MI, 48824

Computational problem solving practices are beginning to be the center of many introductory physics courses. Specifically, at MSU, students regularly work on computational problems situated in physics that involve minimally working programs. Currently, very little research has been done on minimally working programs in regards to problem difficulty. From problem-based literature, the paper will use the traits of Complexity and Structuredness from Jonassen et al. Complexity is used to create a framework of problem difficulty applicable to minimally working programs while Structuredness is used to study possible epistemological conflicts between curriculum intention and design.

I. INTRODUCTION

There is a current movement towards integrating computation into the physics curriculum at both the high school and undergraduate level [1]. This movement is motivated by the need to prepare the next generation of STEM majors for a future in which computation has become ubiquitous with doing STEM professionally [2]. In introductory physics courses, students typically have little to no previous experience coding. Minimally working programs (MWP) have been introduced as one approach to integrating computation into introductory physics courses in a way that does not require previous coding knowledge [3–5]. MWP are semi-structured computational activities that are partially completed, requiring students to add to or edit given code rather than writing a program from scratch. Despite their use at multiple institutions, MWP are currently understudied in PER. In this paper, we lay the foundations of understanding how students interact with MWPs within a problem-based learning (PBL) course at Michigan State University (MSU) called Projects and Practices in Physics (P^3). By blending PBL design research with video data collected of students working on MWPs in P^3 , we model an approach to gauging difficulty of MWPs. Using this approach, we hope to answer whether it is possible to understand the complexity and structuredness of a MWP through the lens of PBL literature on problem design.

A. PBL & MWP Described

Problem-based learning is a teaching style developed at McMaster University in the late 1960s that emphasizes open-ended problems as the driving characteristic for student learning [6]. Dolmans describes problem-based learning as having three characteristics [7]: (1) problems are used as a stimulus for learning (as opposed to using the teacher as a stimulus for learning), (2) tutors are facilitators of the problems to help provide guidance, and (3) group work is a stimulus for interaction. When designing an activity in PBL, the activity should follow two principles [8]. Firstly, PBL problems should be open-ended and allow for multiple different ways of solving the same problem. Secondly, PBL problems should simulate or be comparable to real problems that professionals may encounter in a practical environment. This allows students to relate to the problem because there's a practical real-world relationship within the problem. Students won't relate to their professional experience, but will use their own real-world interpretation and visualization. PBL problems must also be provocative and challenging to create the motivation for learning for the students.

In a typical MWP, students are given a chunk of code as part of the problem statement [9]. The code will successfully run, but intentionally model a physics-based scenario unsuccessfully, often conflicting with student expectations (e.g. two balls will pass through each other rather than colliding and bouncing off). This allows students to immediately un-

derstand that something with the code is incorrect, and that it is their task to modify the code so that it successfully models the physics scenario.

Currently, MWPs are designed through an iterative approach in class, relying on the expertise of instructors to incorporate computational thinking skills, problem solving skills and conceptual understanding into the activity in a scaffolded way. There is little research in the design of MWPs, and no articulated framework exists to guide the design of a MWP [10]. This has resulted in a buildup of unanswered questions. For example how much of the physics scenario should be modelled correctly? Should certain variables, such as initial conditions, within the MWP be left accurate to simulate correct physics, or should they be left to the students to correct?

B. The Hovercraft Problem in P^3

P^3 is a flipped, PBL introductory mechanics classroom at MSU for science and engineering majors. In-class work is entirely group-based, with students working in groups of 4 to solve complex problems as a team [3]. In seven of those class periods, students are given problems in the form of MWPs. Since P^3 is a first-semester, introductory mechanics course, many of the enrolled students have little to no computational experience. Because code is provided for students as part of the activity, MWPs significantly reduce the focus on writing code and instead re-frame computation as a tool to do physics. From this perspective, it is explicitly messaged, several times over the semester, that students are not expected to learn a programming language as well as a physics curriculum at the same time. Instead, the focus is to gain experience with computational thinking practices that are important to doing physics.

In the given problem to the students, the starting point for the MWP is a hovercraft moving with a constant velocity to the edge of the cliff, and then coming to a stop. Students are asked to modify the given code so that it will model the first hovercraft falling after it has passed the cliff edge, and then they are also asked to create and model the second hovercraft accelerating to the cliff edge and falling. The focus of our analysis will be with students working on the above problem.

II. COMBINING MWP WITH PBL

In addition to MWP providing a starting point for coding in an introductory context, MWP were chosen for P^3 because they could be easily incorporated into the PBL design principles. The alternative approach to teaching programming in introductory physics has historically been in the form of cookbook style programs developed from scratch [11]. This approach is inconsistent with the constructivist perspective that informs the design of a PBL classroom. The MWPs in P^3 are designed to have an ideal solution path (the most efficient),

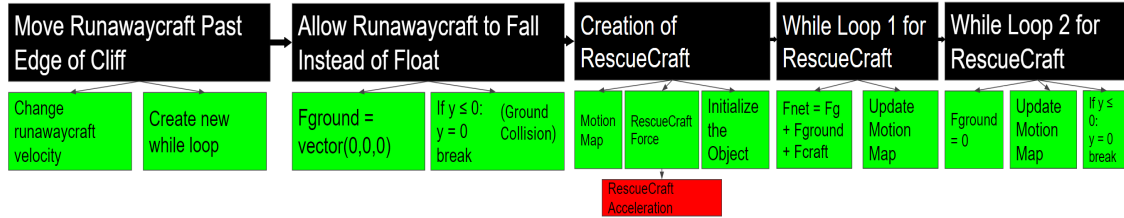


FIG. 1. Task analysis showing minimum intricacy. Top row represents Top Tier categorizations.

but often there are alternative approaches that will result in a successful model of the physics. The MWP's are also designed to be complex, but the designers of P³ have no specific gauge for complexity or difficulty.

While little research exists on the design and study of MWP's, PBL has a rich and extended history of research detailing different methods of problem design and study. However, the focus of this paper uses Jonassen and Hung's paper on problem design and difficulty in PBL [12], and translates this analysis to a MWP context. This framework was chosen as it focused on the shared qualities of MWP's and PBL problems. It was also chosen as aspects of the framework could be applied to analyze in-class video data. The framework outlines examining problems from two separate perspectives, each with their own properties: Complexity and Structuredness. Structuredness is a description of the number of possible solutions to a problem, with an ill-structured problem having many and a well structured problem having one defined solution. The bulk of the analysis presented here is focused on complexity, however we also discuss structuredness as an insight into how open-ended a problem is through examination of legitimate competing alternative solutions to a problem.

Jonassen and Hung outline different properties of complexity [12]. They view complexity as influenced by breadth of knowledge required, the attainment level of domain knowledge, the intricacy of the problem-solving procedures, and the relational complexity. For the purposes of this study, we focus on the aspects of complexity that can be observed in students in-class work: intricacy of the problem-solving procedures and the relational complexity. Breath of knowledge required and attainment level could be unpacked from the student perspective but this would have to be completed on an individual basis. Jonassen and Hung describe the intricacy as the solution path length [12]. It includes the number of steps to be executed in a solution path. From the work of Jonassen & Hung, they include the solution path length to be the number of steps [12]. Jonassen and Hung also define relational complexity as "the number of relations that need to be processed in parallel during a problem solving process [12]." In the following parts, we will redefine both the idea of a step, and relational complexity, in the context of this research.

III. RESEARCH METHOD

For the purposes of this study, a step is defined as a task that was either verbalized in the video by an error message, a student, or the instructor working with the group. While Jonassen & Hung use step, we will be explicitly using the term "task" in the context of the research. Relational complexity is defined as the number of relations between tasks that are processed at the same time. The more complex the relations, and the more tasks that are processed at the same time, the more complex the problem is as a whole. In our analysis, we tiered the tasks that occurred based on their relationship to each other vertically. Within our data, it may be possible that relational complexity can also work horizontally, but it was found to be too complex to show when groups are working on the same problem at the same time. We acknowledge that this is likely a limiting factor of our research. If a task is needed to be solved in order to make progress on a related task then the task that was needed to be solved was tiered below the task that needed its solution. We measured relational complexity at the top tier task level and counted the amount of lower tier tasks that needed to be solved in order to come to a resolution to the top tier task.

Initially we created a task analysis from an expert perspective of the ideal solution path length. At its core, a task analysis is a methodology that outlines the steps one takes to solve a particular problem, emphasizing the actions or processes taken to solve a specific goal [13]. The task analysis allows observation into the specific problem solving path that each group took to solving the problem, and compare these paths. Students can and will encounter many different types of tasks as they attempt to problem-solve, depending on how they perceive the problem. These perceptions can be the primary influence, and can even influence decisions that they make throughout the process. However, every problem has required tasks that students must take to solve the problem. Finding the essential tasks to the hovercraft problem outlines the quickest path that each group has to take, even if their overall solution paths may be different, creating an initial framework for the hovercraft problem. The ideal path identified through task analysis is displayed in Fig.1. Using this task analysis, each problem was broken down into different categories of tasks outlined in Table I. The tasks were classified based on a tiered system described previously but

illustrated in the following description.

TABLE I. Task Analysis Categories

Categorization	Definition
Top Tier Tasks	The 5 main tasks that have to be resolved to solve part 3 of the hovercraft problem
Primary Tasks	Path students take to solve the <i>Top Tier</i> tasks.
Secondary Tasks	Integral to be solved before resolving primary tasks
Tertiary Tasks	Integral to solve before resolving the secondary tasks in the task analysis
Quarternary Tasks	Integral to solve before resolve the tertiary tasks

The students first view the craft come up to the edge of the cliff and then stop. This is due to the existence of only one while loop: *while runawaycraft.pos.x<0*. With this, the first primary task is to make the craft move past the edge of the cliff (where the cliff edge is defined as $x=0$) with the correct velocity. The secondary tasks require students to change the initial velocity to the velocity found in Part 1, and create a new while loop for $x>0$. The next primary task is to add the correct physics within the while loop so that the runawaycraft falls instead of floats after x is greater than 0. The secondary tasks require that the students change the normal force vector to the 0 vector and create some form of collision detection when $y = 0$. After finishing the first runawaycraft, the following primary task is to model the rescuecraft. To complete this, the students must create the motion map, object, and the force for the accelerating crafts. Within the secondary task, there exists a tertiary task of inputting the rescuecraft acceleration from Part 1 of the problem. Finally, the group would repeat what was done for the runwaycraft with the rescuecraft, creating both while loops and the necessary physics equations required to model the motion accurately. The second step in our research process was to observe students solving the MWP from in-class recordings. We analyzed four separate groups solving the hovercraft MWP. Within the recording, a new task was created within the task analysis if either an actor within the video (e.g. students, instructor, computer error) verbalized that a task occurred or if the students verbalized a falsehood about the problem. The initial version of Table I was iterated on after the student based task analysis was complete to include tertiary and quarternary tiers.

IV. RESULTS

For each group analyzed, a visualization of their solution path was produced (shown in Fig. reffig2) and a table indicating the number of tasks in their solution was created (shown in Table II). Each point in Fig. 2 is represented in a sequential manner, where each succeeding point is a task initialized

afterwards by the group. To demonstrate the relational complexity, we also indicate the number of tasks that fell under each top tier task for each group (shown in Table III).

Table II quantifies the intricacy of the problem solving path and creates a comparison between the number of tasks taken by each group after a 2 hour class period. The ending Top Tier task was determined to be the problem that the group was working on once the class period ended.

TABLE II. Intricacy of Problem Solving Path

Group	Number of Steps	Ending Top Tier Problem
Ideal	14	Completion
Alpha	15	Fall instead of Float
Beta	16	While Loop 2 for Rescuecraft
Gamma	20	Creation of Rescuecraft
Delta	17	Creation of Rescuecraft

The results highlight that groups encounter a greater number of tasks than the experts without even getting through all of the top tier tasks. This analysis also highlights that no group successfully completed the code within the time limit. It is clear that even to get to the creation of the rescuecraft object involves at the very least 3 more tasks than the ideal path length. In isolation, these results do not yet allow us to unpack the relationship between tasks and level of complexity. The variation in amount of tasks attempted mixed with the variation in how much of the problem was solved hinders any claims related to average task count to a solution. However, this approach to examining intricacy with more data could provide insight into the relationship between ideal and actual tasks and identifying the appropriate number of tasks for a problem to be solved within the 2 hour time constraint of problem sessions.

In contrast, the relational complexity analysis does reveal some insights into this particular problem. It indicates that the brunt of the complexity for this problem can be found in getting the craft past the edge of the cliff. This is interesting as the amount of physics concepts involved in developing a solution to this problem is minimal and instead the focus is on adjusting the parameters of the while loop. The results also highlight that there is a different legitimate path direction that can be taken, which is to model the rescuecraft first. The problem designers thought the ideal path would be to model the craft that did not need to include a force in the x -direction on top of the cliff but Beta group went with the rescuecraft first. From the perspective of either craft, these results would indicate from a curriculum design perspective that having to model two crafts is not a suitable problem given the time constraints.

A. Structuredness and Tangents

The analysis process undertaken demonstrated that there were unclassified tasks that students identified that would not

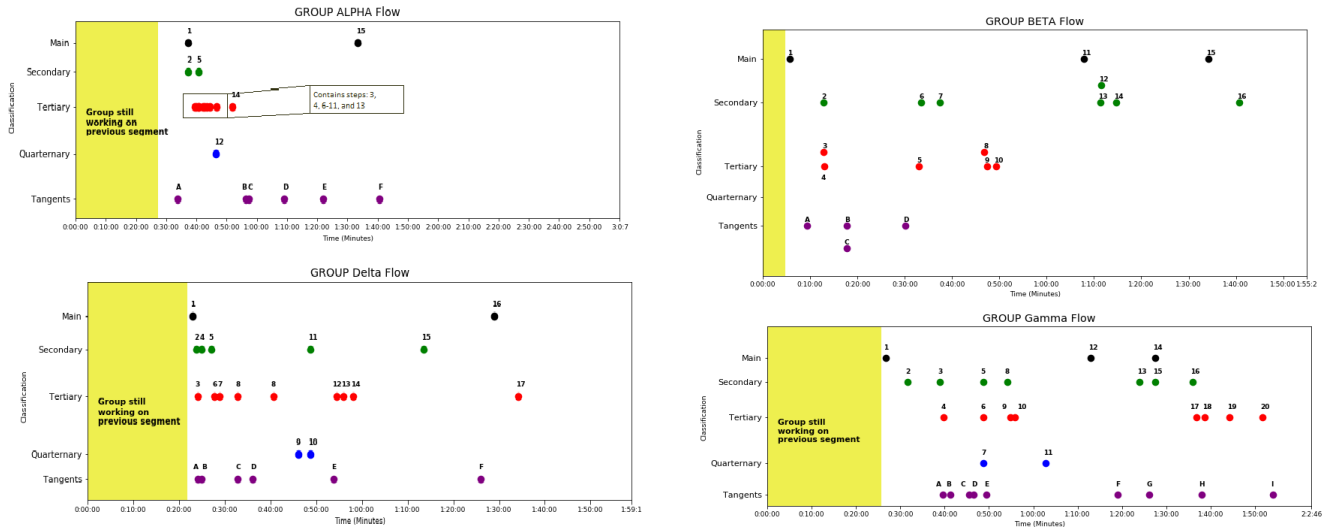


FIG. 2. Group Task Analysis Example

TABLE III. Relational Complexity

Group	Craft Past Edge	Fall Instead of Float	Creation of Rescuecraft	Rescue Loop 1	Rescue Loop 2
Alpha	14	1	NA	NA	NA
Beta	NA	NA	10	4	2
Gamma	11	2	7	NA	NA
Delta	15	2	NA	NA	NA

end up in their solution path. These were initially classified as “tangents.” Legitimacy of competing interpretations is the best lens to study these tangents because tangents exist outside of the ideal problem solving path as an alternate solution, such as using time in a while loop instead of position. Legitimacy of competing interpretations also helps align our framework with PBL literature by acknowledging the multitude of responses students may give while completing a MWP. This interpretation of using time was used by 3 out of the 4 groups and is certainly a legitimate method of solving this problem, even if it may not be the ideal method given that position is already set up for the group to solve. The video data revealed that instructors pushed students away from using time within the while loop. This reveals that although the problem is open-ended and in keeping with PBL tradition, the teaching practice within the classroom is conflicting with the learning approach encouraged by the curriculum design.

V. DISCUSSION & CONCLUSION

Analyzing groups solving problems from the PBL perspective of complexity and in particular intricacy of problem-solving procedures and relational complexity provided some initial insights. It helped to identify the areas of the problem that students are finding the most complex. Furthermore, it

highlighted a disconnect between the PBL and MWP teaching philosophy in addition to the disconnect of teaching practice within the P³ environment. As mentioned previously, the disconnect occurred when combining the instructor expectations of successful teaching experiences with the teaching philosophy of PBL literature. The difference in number of tasks between the ideal solution and the groups’ solutions and the difference in tasks between groups who got to the same point in the problem highlights a limitation to this approach. The learning environment and enacting of these problems is complex and reducing this complexity down to number of tasks reduces the insight the analysis can provide. Because the students are solving the problems as groups as opposed to as an individual, there could be extra tasks through having to negotiate four individuals’ understanding of the problem. However, we would argue that a continued analysis that included examining more groups who solved the same problem and examining other MWPs used in this class could provide a deeper insight into MWP design especially when combined with the relational complexity analysis.

-
- [1] N. R. Council *et al.*, *A framework for K-12 science education: Practices, crosscutting concepts, and core ideas* (National Academies Press, 2012).
- [2] S. Olson and D.G. Riordan, Engage to excel: Producing one million additional college graduates with degrees in science, technology, engineering, and mathematics. report to the president., Executive Office of the President (2012).
- [3] P. W. Irving, M. J. Obsniuk, and M. D. Caballero, P3: A practice focused learning environment, *European Journal of Physics* **38**, 055701 (2017).
- [4] S. Weatherford and R. Chabay, Student predictions of functional but incomplete example programs in introductory calculus-based physics, in *AIP Conference Proceedings*, Vol. 1513 (AIP, 2013) pp. 42–45.
- [5] M. A. Kohlmyer, Student performance in computer modeling and problem solving in a modern introductory physics course, Unpublished doctoral dissertation, Carnegie Mellon University (2005).
- [6] H. S. Barrows, *Problem-based learning applied to medical education* (Southern Illinois University School of Medicine, 2000).
- [7] D. H. J. M. Dolmans, W. De Grave, I. H. A. P. Wolfhagen, and C. P. M. Van Der Vleuten, Problem based learning: future challenges for educational practice and research, *Medical Education* **39**, [10.1111/j.1365-2929.2005.02205.x](https://doi.org/10.1111/j.1365-2929.2005.02205.x) (2005).
- [8] L. Wilkerson and W. Gijssels, Bringing problem-based learning to higher education: Theory and practice, *New directions in teaching and learning* **68** (1996).
- [9] S. Weatherford, *Student use of physics to make sense of incomplete but functional VPython programs in a lab setting* (2011).
- [10] B. R. Lunk, *A Framework for Understanding Physics Students' Computational Modeling Practices* (North Carolina State University, 2012).
- [11] R. Chabay and B. Sherwood, Computational physics in the introductory calculus-based course, *American Journal of Physics* **76**, 307 (2008).
- [12] D.H. Jonassen and W. Hung, All problems are not equal: Implications for problem based learning, *Essential Readings in Problem Based Learning* **5**, 7 (2015).
- [13] B. Kirwan and L.K. Ainsworth, *A guide to task analysis: the task analysis working group* (CRC press, 1992).