Work in Progress: Preliminary Assessment of a Problem-Solving Think-Aloud Project using the PROCESS Framework

Dr. Justine Chasmar, Minnesota State University, Mankato

Dr. Chasmar is an adjunct professor for Iron Range Engineering through Minnesota State University, Mankato. She teaches self-directed learning classes, supporting student engineers as they transition to full time co-op. Dr. Chasmar earned a Ph.D. in Engineering and Science Education and M.S. and B.S. in Mathematical Sciences from Clemson University. She is a STEM educator, practitioner, and scholar.

Before IRE, Dr. Chasmar spent over a decade serving in and directing learning centers. Most recently, Dr. Chasmar served as an Assistant Professor of mathematics and the founding Director of the Quantitative Reasoning Center at Goucher College, supporting student numeracy and STEM programs across campus. Prior to that, she coordinated large student support programs in learning centers at Clemson University and the Georgia Institute of Technology.

Dr. Chasmar's research focuses on student motivation, self-directed learning, numeracy education, and professional identity development. Through her background in learning centers, she has applied this research to undergraduate students and peer learning programs.

Dr. Sarah Jane Grigg, Embry–Riddle Aeronautical University, Daytona Beach

Dr. Sarah Grigg joined Embry-Riddle in 2019 as an Assistant Professor in the Engineering Fundamentals Department where she teaches EGR 120 Engineering Graphics with CATIA.

She earned her bachelor's, master's, and doctoral degrees in Industrial Engineering, a master's degree in Business Administration, and a graduate certificate in Engineering and Science Education from Clemson University. She joined the Clemson faculty in 2012 and was a senior lecturer in the General Engineering Program teaching engineering courses in problem solving, programming, computer aided-design, and innovation and entrepreneurship before joining ERAU.

As a human factors design engineer, she specializes in process improvement, error mitigation, and sustainable and accessible design. Current research interests in engineering education includes the development of PROCESS framework and assessment for problem solving, iLEARNED modular scaffolding approach to organizing course content in a flipped classroom, and the incorporation of entrepreneurial minded learning into engineering courses.

Dr. Darcie Christensen, Minnesota State University, Mankato

Dr. Darcie Christensen is a probationary Assistant Professor in the Department of Integrated Engineering at Minnesota State University Mankato. She teaches for Bell Engineering, which is a subset of Iron Range Engineering on the Mesabi Range College Campus. Dr. Christensen received her Ph.D. in Engineering Education from Utah State University in the Summer of 2021. The title of her Dissertation is "A Mixed-Method Approach to Explore Student Needs for Peer Mentoring in a College of Engineering." Darcie holds a Master of Engineering degree in Environmental Engineering (2019) and Bachelor of Science degree in Biological Engineering (2017), both from Utah State University. She is passionate about student success and support, both inside and outside of the classroom.

WIP: Preliminary Assessment of a Problem-Solving Think-Aloud Project Using the PROCESS Framework

Introduction

In this Work‑In‑Progress empirical research paper, we present preliminary findings from a qualitative analysis of a think‑aloud problem‑solving project using the PROCESS framework—**P**roblem definition, **R**epresenting the problem, **O**rganizing information, **C**alculations, **E**valuating the solution, **S**olution communication, and **S**elf‑assessment. The project, introduced in a prior 2025 ASEE article [1], involves students recording themselves solving an introductory fluid mechanics problem using the think-aloud method within a self-directed learning course. The project aims to introduce beneficial metacognitive problem‑solving strategies and evaluate how well PROCESS captures students' behaviors [2], [3]; this work in particular updated the original codebook for the current context. Initial analysis shows frequent metacognitive actions including strategy selection, problem framing, confidence calibration, and real‑time error monitoring, prompting minor refinements to PROCESS codes. These results support the utility of think‑aloud activities for documenting metacognitive processes and using PROCESS as a framework guiding future large‑scale studies.

Background and Motivation

Problem-solving is a core competency for engineering graduates, reflected in ABET Outcome 7, which requires the ability to acquire and apply new knowledge using appropriate learning strategies [4]. Achieving this outcome depends on metacognitive skills, as students must monitor and regulate their thinking. Metacognition, or "thinking about thinking," is central to self-directed learning and effective problem-solving in complex contexts [5]–[8]. Unlike traditional problem-solving, which emphasizes correct answers, metacognitive problem-solving involves deliberate monitoring and control of strategies, assumptions, and problem framing, including planning, monitoring, and evaluation of intermediate and final results [9], [10].

Prior studies suggest that interventions supporting metacognition can influence how students approach engineering problems [11], and skills can be cultivated through reflective writing and structured instructional support [12]. However, few studies both train students in metacognitive problem-solving and measure those processes during authentic problem-solving. To address this gap, the present study examines how students engage in problem-solving through a structured think-aloud. Artifacts from first year engineering were analyzed using the PROCESS framework, which captures cognitive and metacognitive dimensions for well-structured problems [13], [14]: task execution, strategy use, errors, and answer states [15],[16]. Students first develop proficiency with well-structured problems before transitioning to ill-structured, authentic design problems [13], [14]. This work extends PROCESS to non-traditional engineering students and upper-level engineering coursework.

The larger study is guided by the research question: *What metacognitive problem-solving skills do engineering students demonstrate and develop through participation in a think-aloud project?* The present paper focuses on the sub-question: *To what extent does the PROCESS coding scheme capture observed behaviors, and what refinements are needed to fully represent students' processes?*

Methodology

Participants and Context. Participants were first-semester students enrolled in a non-traditional, co-op–based engineering program and taking an introductory fluids course. The program emphasizes project-based learning, reflective practice, and development of professional skills alongside technical content [17]. All students had prior exposure to structured problem-solving and reflection activities as part of the curriculum. Three participants' artifacts were selected because they illustrated in-depth think-alouds with more evidence of metacognition that could provide greater opportunity to validate current codes and elucidate additional codes. Participants were given pseudonyms in this paper to protect identity.

These artifacts were coded by two raters using consensus coding [18] to evaluate task documentation and metacognitive elements. Coder 1 was the instructor for the think-aloud assignment and Coder 2 was the developer of the coding scheme with extensive experience using the codebook on first year engineering problems. This assignment was given to non-traditional students in an upper-division engineering course with an emphasis on establishing generalizability to other engineering courses and updating it as needed for the new context (Fluid Mechanics), not focused on student outcomes. Coder 1 transcribed the think-alouds and then Coder 1 and 2 separately watched the video and coded the transcription of the assignment. They reviewed each other's codes to agree on the most appropriate code when disagreement occurred.

Instructional Intervention. As part of a problem-solving project, students participated in workshops on metacognition and problem-solving, then created a think-aloud video while solving a fluid-mechanics gauges problem (full assignment and instructions in Appendix A). Students were instructed to verbalize their thinking while working and to submit the recording. Full instructional design details are reported elsewhere [1].

Data Analysis

This study analyzed three think-aloud videos selected for in-depth qualitative examination. Coding was grounded in the validated PROCESS framework [15] to align observed behaviors with conceptual definitions and ensure analytical rigor [19]. Prior work with PROCESS shows higher-quality solutions exhibit stronger evidence of representation, planning, monitoring, and self-management, whereas lower-quality solutions show missing elements, sequencing breakdowns, and limited evaluation [15], [16]. Additional audio codes, from prior PROCESS work [20], include task documentation and metacognitive codes that are not explicit in written work but appear in think-alouds.

We employed a qualitative approach combining *a priori* PROCESS codes with iterative refinement: codes were refined only when data revealed patterns that did not fit existing definitions, reducing the risk of forcing behaviors into inappropriate categories. Analytic memos documented decisions and emerging insights, creating a transparent record. Units of analysis were meaningful segments of student talk or action, focusing on problem-solving steps, strategy use, errors, and self-assessment behaviors. Metacognitive strategies are included in **E**valuating the solution, **S**olution communication, and **S**elf-assessment codes. The goal was to evaluate framework fit, clarity, and usability rather than quantify frequency. The resulting coding scheme is included in Appendix B.

Results and Discussion

Across the three participants, we observed frequent self-regulation during problem-solving, especially when students noticed and corrected errors, chose and adapted strategies, clarified problem setup, and expressed confidence or uncertainty along the way. These self-regulation behaviors were captured using the PROCESS framework along with audio and metacognitive codes; see Appendix B for details on these codes. In the sections that follow, we describe these behaviors and explicitly note how they led to targeted refinements of the PROCESS coding framework.

Error Management. Participants identified and fixed errors as they worked, showing real-time monitoring of their calculations and notation. For example, Sam corrected a numerical slip, "Hm, no… P_1 is not 9… This is supposed to be 11." which was coded as Identify error (correctly) with an Incorrect calculation error. Taylor demonstrated quick error detection and repair (e.g., correcting P_A to P_B), which was captured by splitting contiguous utterances into an initial error followed by correct identification. Alex made an early error, corrected it after consulting a solution video, and then fixed the remaining error promptly in the final recording. Together, these examples show ongoing self-checks, diagnosis, and immediate adjustments, hallmark features of metacognitive problem-solving. The fact that participants verbalized strategies and confidence aligns with prior work showing that non-reactive think-alouds can reveal meaningful aspects of how students think during problem-solving [10]–[12], [21]. Differences in how students verbalized monitoring (brief doubt versus affirmative checks) motivated **splitting $\mathbf{S_2}$** Evaluating performance into $\mathbf{S_2}$ Expressing doubt and $\mathbf{S_2}$ Expressing confidence, enabling a more precise capture of real time error monitoring in PROCESS.

Strategy Use. Participants made their strategy choices explicit, often referring to notes, previous work, or worked examples to move past sticking points. Alex used a solution video to resolve an early error and described redoing the problem as helpful extra practice. Taylor referred to notes and self-checks to confirm assumptions and calculations, and Sam used strategy-focused self-talk while narrowing down the source of a mistake. Prior research has shown that think-alouds can be useful for surfacing these kinds of strategic decisions, which often remain hidden in written solutions [3], [22], [23]. From an instructional standpoint, simple prompts such as "What will you try next, and why?" may further encourage students to plan and reflect on their strategy choices. Observed use of external resources (notes, worked examples, solution videos) and explicit planning directly motivated **adding** the $\mathbf{S_2}$ Information seeking code to PROCESS, enabling clearer capture of when and why students seek strategy support.

Problem Representation (Framing and Definitions). Another common behavior was students clarifying what the problem meant before calculating, including identifying variables, defining terms, and deciding what counted as the system versus the surroundings. For Taylor, several statements originally coded as self-assessment were more accurately coded as **R**epresenting the problem, such as: "P_B is… this is the surrounding, this is the system… It is a vacuum… so this one is gauge." Carefully framing the problem is widely recognized as an important part of engineering problem-solving, and making these moves visible and codable helps explain how students avoid, or create, later errors [24], [25]. Brief pre-calculation prompts that ask students to explain symbols or define the system may help support this step. Frequent utterances that clarified symbols, units, and system boundaries motivated **adding** the **R** Identify definitions code in PROCESS, enabling more precise capture of problem setup moves.

Self-Assessment. Students expressed confidence or uncertainty while working, offering short evaluations of their progress. There were often brief self-evaluative comments: Sam frequently voiced uncertainty (e.g., "Hm, no. This is not right. Let's see.") while identifying errors, whereas Alex reflected more positively on learning from repetition ("This is good practice for me to set up this third problem"). While this in-the-moment self-assessment was common, explicit end-of-problem checks were less consistent. For instance, Sam arrived at the correct answer but did not clearly evaluate the final result. This pattern reflects earlier findings that students are more likely to monitor their work during problem-solving than to pause and evaluate their final answer unless prompted [26]. Brief, end-of-problem questions, such as asking students to justify why their answer makes sense, may help make this step more routine. Variation in how students expressed confidence or uncertainty during these self-evaluations motivated **splitting** the original $\mathbf{S_2}$ Evaluating performance code into $\mathbf{S_2}$ Expressing doubt and Expressing confidence for more precise capture of in-the-moment monitoring.

Finally, comments such as Alex's "Oops, eraser" and "sorry that's a little un-neat" suggest that using recording and writing software introduced some additional cognitive load as was evidenced in a prior study [27]. Although examining tool-related effects was outside the scope of the current study, this observation points to an opportunity for future work on how digital interfaces shape students' problem-solving experience [3].

Conclusions

Overall, the PROCESS framework was a strong fit, with minor refinements, for capturing the observable metacognitive behaviors elicited from the think-aloud intervention, including error monitoring, strategy selection, confidence calibration, and early problem framing. The refinements made during analysis improved clarity without expanding the framework beyond its original intent. We also clarified that self-assessment codes were not counted as task documentation, preventing inflated task totals preserving metacognitive detail. Instructionally, we recommend structured prompts to encourage end-of-problem evaluation and early representational framing, as well as formal reliability reporting in future analyses.

Limitations and Future Work

This study is limited by its small sample size and single-institution context. The analysis focused on three recordings from one introductory engineering course, limiting generalizability but supporting the goal of evaluating framework fit. Future work should apply the revised scheme across larger datasets, multiple courses, and varied problem types. Studies should also examine how instructional prompts influence strategy use and evaluation behaviors, report formal inter-coder reliability measures [28], and document coder training and reconciliation to further strengthen claims of reliability and validity.

Acknowledgements

ChatGPT [29] and Microsoft 365 Copilot [30] were used for language and formatting assistance; all research design, analysis, and interpretation were conducted by the authors.

References

[1] J. Chasmar, S. J. Grigg, and D. Christensen, “WIP: Scaffolding the metacognitive problem-solving process in an undergraduate engineering peer-review project,” in *Proc. ASEE Annu. Conf. Exposition*, Montreal, QC, Canada, Jun. 2025, doi: 10.18260/1-2--57443.

[2] K. A. Ericsson and H. A. Simon, “How to study thinking in everyday life: Contrasting think-aloud protocols with descriptions and explanations of thinking,” *Mind, Culture, and Activity*, vol. 5, no. 3, pp. 178–186, 1998.

[3] R. S. Nickerson, “The teaching of thinking and problem solving,” in *Thinking and Problem Solving*, R. J. Sternberg, Ed. New York, NY, USA: Academic Press, 1994, pp. 409–449.

[4] ABET, *Criteria for Accrediting Engineering Programs, 2024–2025*. Baltimore, MD, USA: ABET, 2024. [Online]. Available: https://www.abet.org/accreditation/accreditation-criteria/criteria-for-accrediting-engineering-programs-2024-2025/

[5] S. D. Sheppard, K. Macatangay, A. Colby, and W. M. Sullivan, *Educating Engineers: Designing for the Future of the Field*. San Francisco, CA, USA: Jossey-Bass, 2009.

[6] J. G. Donald, *Learning to Think: Disciplinary Perspectives*. San Francisco, CA, USA: Jossey-Bass, 2002.

[7] National Academy of Engineering, *The Engineer of 2020: Visions of Engineering in the New Century*. Washington, DC, USA: National Academies Press, 2004.

[8] ASEE, *The Engineering Mindset Report: A Vision for Change in Undergraduate Engineering and Engineering Technology Education*. Washington, DC, USA: American Society for Engineering Education, 2024. [Online]. Available: https://mindset.asee.org/

[9] J. H. Flavell, “Metacognition and cognitive monitoring: A new area of cognitive–developmental inquiry,” *Amer. Psychol.*, vol. 34, no. 10, pp. 906–911, Oct. 1979, doi: 10.1037/0003-066X.34.10.906.

[10] B. J. Zimmerman, “Becoming a self-regulated learner: An overview,” *Theory Pract.*, vol. 41, no. 2, pp. 64–70, Spring 2002, doi: 10.1207/s15430421tip4102_2.

[11] J. Alvarez, M. Fowler, J. R. Amos, and Y. Gertner, “WIP: Students’ metacognition and how it relates to their performance in conceptual problem-solving introductory engineering courses,” in *Proc. ASEE Annu. Conf. Exposition*, Montreal, QC, Canada, Jun. 2025, doi: 10.18260/1-2--57450.

[12] P. Wojahn, G. Degardin, M. Dawood, M. J. Guynn, and R. Boren, “Increasing metacognitive awareness through reflective writing: Optimizing learning in engineering,” in *Proc. ASEE Virtual Annu. Conf.*, Jun. 2020, doi: 10.18260/1-2--34818.

[13] D. H. Jonassen, “Instructional design models for well-structured and ill-structured problem-solving learning outcomes,” *Educational Technology Research and Development*, vol. 45, no. 1, pp. 65–94, 1997.

[14] D. H. Jonassen, “What is problem solving?” in *Learning to Solve Problems: An Instructional Design Guide*, vol. 6. San Francisco, CA, USA: John Wiley & Sons, 2004, ch. 1.

[15] S. J. Grigg and L. C. Benson, “A coding scheme for analysing problem-solving processes of first-year engineering students,” *Eur. J. Eng. Educ.*, vol. 39, no. 6, pp. 617–635, Nov. 2014, doi: 10.1080/03043797.2014.895709.

[16] S. J. Grigg and L. C. Benson, “Promoting problem-solving proficiency in first-year engineering: PROCESS assessment,” in *Proc. ASEE Annu. Conf. Exposition*, Seattle, WA, USA, Jun. 2015, Paper ID 1278.

[17] R. Ulseth, B. Johnson, and C. Kennedy, “Iron Range Engineering,” *Adv. Eng. Educ.*, vol. 9, no. 3, pp. 1–34, 2021.

[18] M. B. Miles, A. M. Huberman, and J. Saldaña, *Qualitative Data Analysis: A Methods Sourcebook*, 4th ed. Thousand Oaks, CA, USA: SAGE, 2019.

[19] K. Walther, N. W. Sochacka, and N. N. Kellam, “Quality in interpretive engineering education research: Reflections on an example study,” *J. Eng. Educ.*, vol. 102, no. 4, pp. 626–659, Oct. 2013, doi: 10.1002/jee.20029.

[20] L. Benson, M. Cook, C. McGough, and S. Grigg, “Work in progress: Audio reflections provide evidence of metacognition during students' problem solving attempts,” in *2012 Frontiers in Education Conference Proceedings*, Oct. 2012, pp. 1–2.

[21] T. A. Litzinger et al., “A cognitive study of problem solving in statics,” *J. Eng. Educ.*, vol. 99, no. 4, pp. 337–353, Oct. 2010.

[22] G. Schraw and D. Moshman, “Metacognitive theories,” *Educ. Psychol. Rev.*, vol. 7, no. 4, pp. 351–371, 1995.

[23] K. A. Ericsson and H. A. Simon, *Protocol Analysis: Verbal Reports as Data*, Rev. ed. Cambridge, MA, USA: MIT Press, 1993.

[24] C. J. Atman et al., “Engineering design processes: A comparison of students and expert practitioners,” *J. Eng. Educ.*, vol. 96, no. 4, pp. 359–379, Oct. 2007.

[25] D. H. Jonassen, *Engineers as Problem Solvers*. Cambridge, UK: Cambridge Univ. Press, 2014.

[26] B. J. Zimmerman and M. M. Pons, “Development of a structured interview for assessing student use of self-regulated learning strategies,” *American Educational Research Journal*, vol. 23, no. 4, pp. 614–628, 1986.

[27] S. J. Grigg, S. K. Garrett, and L. C. Benson, “Using the NASA-TLX to assess first year engineering problem difficulty,” in *Proc. Ind. Syst. Eng. Res. Conf.* (*ISERC*), Orlando, FL, USA, May 19–23, 2012, Abstract ID 992.

[28] K. Krippendorff, *Content Analysis: An Introduction to Its Methodology*, 4th ed. Thousand Oaks, CA, USA: Sage, 2018.

[29] OpenAI, “ChatGPT (GPT-5),” OpenAI, San Francisco, CA, USA. [Online]. Available: https://openai.com/chatgpt. Accessed: Jan. 13, 2026.

[30] Microsoft, *Microsoft 365 Copilot*. [Online]. Available: https://copilot.microsoft.com/. Accessed: Jan. 11, 2026.

APPENDIX A The Think-Aloud Assignment Including Example Fluid Mechanics Homework Problem for the Problem-Solving Project from [1].

Problem-Solving Post-Assignment Video Recording

Metacognitive reflection and recording a "think-aloud" while solving a problem has been shown to improve the problem-solving process and reduce errors.

For this assignment, please record yourself working through the assignment and doing metacognitive reflections. Record on your ipad with audio so that your audio reflections are matched with your work on the problem. You can record directly on your ipad using a screen recording (but make sure the audio is on) or record on Zoom. Make sure that as you work, you vocalize everything you're doing, including accessing the homework solution video, textbook, Googled equations, etc. Turn in both your recording and your written solution in the Google Drive folder with your name by [date].

Tips for reflecting on your work:
Explain what you are thinking, feeling, recalling, remembering, or doing as you are working the problem. Your reflection might begin with:
" I am thinking...", " I remember...", " I recall...", "I wonder..."

Explain why you are doing what you are doing (not just what you are doing, and your motivation or expectations for writing something down. Your reflection might begin with:
"I want to...", "In an earlier problem I______, so here I ...", "I know that by...", "I expect..."

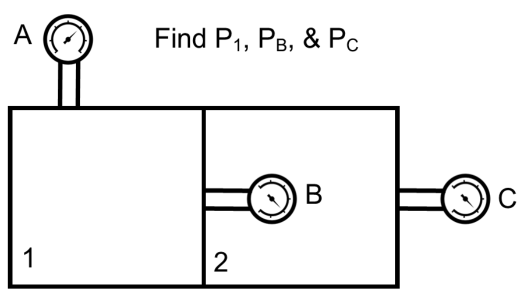
If you encountered anything surprising in your work (or even while replaying your work), tell us about it. Such a reflection might start with:
"I am surprised to find that…" or "I didn't realize until now that…"

For Fluids: See assignment from Fluids (Gauges). Contact your professor with any questions.

Fluid Mechanics Homework—Gauges

A, B, & C are gauges. 1 & 2 are systems. If
P_{atm} = 15 psi_a, P_A = 4 psi_v, & P_2 = 20 psi_a

Find P_1, P_B, & P_C



APPENDIX B PROCESS Codes for Think-Alouds.

Each problem-solving element is given one Stage, Code, Subcode and an Audio code to summarize the result with updated codes highlighted in blue.

PROCESS Stages	PROCESS Codes	Subcodes
P	Identify desired variables and units	Identify unknown value
	Identify problem constraints	Restate problem
	Communicate assumptions	Communicate assumptions
R	Sketch a representation of relationships	Draw a picture / diagram
	Sketch a representation of relationships	Make a table
	Sketch a representation of relationships	Labeling / Renaming
	Identify relevant theoretical equations	Identify relevant theoretical equations
	Identify relevant theoretical equations	Identify definitions
	Match given values to variables	Relate variables
O	Identify known values	Identify known value
	Identify known values	Identify other given information
	Identify reasonable limits on variables	Identify reasonable limits on variables
	Identify conversion factors	Identify conversion factors
C	Solve for unknown variable algebraically	Use ratios
	Solve for unknown variable algebraically	Manipulate equation
	Solve for unknown variable algebraically	Document math
	Solve for unknown variable algebraically	Solve intermediate value
	Convert to SI units	Derive units
	Convert to SI units	Derive units
	Convert to SI units	Derive units
	Input values into equation	Plug values in equation
	Convert to desired units	Convert to desired units
E	Confirm desired values	Identify final answer
	Confirm desired values	Identify final answer
	Check reasonableness / precision	Check reasonableness / precision
	Check units match desired	Check units match desired
S_1	Compare answer to expected value	Check accuracy
	Ensure reasonableness based on limitations	Ensure reasonableness based on limitations
	Justify reasoning	Justify reasoning
S_2	Evaluating performance	Express doubt
	Evaluating performance	Express confidence
	Describe your strategy	Segmentation
	Describe your strategy	Information seeking
	Describe your strategy	Erase work
	Describe your strategy	Abandon process / Start over

Audio Codes
Task documentation
Error committed
Strategy identification
Acknowledge importance of task
Acknowledge confusion / frustration
Acknowledge confidence
Identify error (Correctly)
Identify error (Incorrectly)
Identify error (False Alarm)