Frozen in the Past: When it Comes to Analogy Fears, It's Time For Us to "Let it Go"

Briana Bettin
Michigan Technological University
Houghton, MI, USA
bcbettin@mtu.edu

Linda Ott
Michigan Technological University
Houghton, MI, USA
linda@mtu.edu

ABSTRACT

Within computer science education research, analogy work has been relatively unexplored and in some cases, even discouraged. For a learning and reasoning tool that is so widely used in our discipline — from instructors and peer groups to the interfaces we encourage students to program on —- it is beyond time to address the perception of analogy in our field.

In this position paper we briefly overview relevant cognition and learning literature, summarize applications of analogy across several other STEM disciplines, and compare research and perception of analogy within computing education. Further, we explore some of analogy's potential as a tool which can allow for highly personal, relevant learning that may even assist in development of a sense of belonging or computing identity.

These arguments highlight a fundamental difference in the attitude within our field toward analogy versus that of other STEM disciplines. We aim to understand the differences by exploring the themes behind the concern surrounding the use of analogy in our discipline. In addition, we provide suggestions for how we may address these concerns in order to advance research into the use of analogy in computing education to determine if indeed analogy can enhance student learning.

CCS CONCEPTS

• **Social and professional topics** → **Computer science education**;

KEYWORDS

analogy, metaphor, analogical reasoning, computing education research, cognitive and learning sciences, interest and engagement

ACM Reference Format:
Briana Bettin and Linda Ott. 2021. Frozen in the Past: When it Comes to Analogy Fears, It's Time For Us to "Let it Go" . In *26th ACM Conference on Innovation and Technology in Computer Science Education V. 1 (ITiCSE 2021), June 26-July 1, 2021, Virtual Event, Germany*. ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3430665.3456381

ITiCSE 2021, June 26-July 1, 2021, Virtual Event, Germany

ACM ISBN 978-1-4503-8214-4/21/06…$15.00
https://doi.org/10.1145/3430665.3456381

1 INTRODUCTION

Analogy and analogical reasoning are fundamental communication devices and cognitive reasoning tools for learners and experts alike. Instructors and researchers regularly make use of analogy. Likewise, computing interfaces and languages are a proverbial melting pot of analogies. Despite the prolific use of analogy in education and computing, Computing Education Research (CER) has largely ignored or discouraged the use and exploration of analogy. These "cultural values" of the field have impacted research conducted, methods used, and understanding gained. It is time to develop a better understanding of analogy, of why these themes exist, and to name them in order to move forward as a discipline.

2 MOTIVATIONS

This paper was inspired by experiences of uncertainty and apprehension toward analogy research in CER spaces. In multiple instances when discussing the first author's dissertation work [5], experienced researchers in our field expressed concern about pursuing this topic. When exploring these concerns, the misgivings were not easily explained, but certainly embedded in the discourse.

Further exploration into the literature revealed how pervasive this attitude was, and yet how difficult its rationale was to trace. It was known that instructors use analogies, and many researchers used them frequently or researched concepts that were clearly analogous in nature. Still, the term "analogy" and subsequently, analogical theory were often absent from discussion. Other researchers in computing education have also recently come to recognize this phenomena and the attitudes in our discipline. [32].

3 BACKGROUND

To better situate the conversation surrounding analogy in Computing Education Research, this literature review draws from three distinct areas: (1) theories of mind and cognition, which are fundamental to the ways we process information and learn; (2) analogy research conducted in other STEM Discipline Based Education Research (DBER) areas, as computing fits within the STEM umbrella, making these fields some of our nearest neighbors for best practices; and (3) research within the field of CER that specifically targets and discusses analogy, to parallel the literature from the broader STEM examples.

3.1 A Brief Overview of Cognitive Theories

There is a wealth of valuable and compelling knowledge from within the field of cognitive and learning sciences in relation to analogy. This section contains an abridged background for understanding analogy as a cognitive process and human reasoning tool. For a more extensive literature review of this area, the first author's dissertation explores these themes in more detail [5],

3.1.1 Mental Models. Our **mental models** continue to be tested and modified or validated as we interact with the world [49]. Defined briefly, mental models are our perceptions of how the world and systems within it operate, as well as how we should interact and react. Our mental models are based on prior knowledge, familiar representations, and contextual cues, among other things. The information we engage with and interactions we have cause our mental models to morph and adapt.

Models change through critiques and challenges[12]. Mental models do not begin as perfect — they are imperfect and become more accurate over time, through critique and application. Further, even when challenged, changes to a mental model do not happen instantly. In describing the key properties of mental models, Norman includes incompleteness among these properties [49]. He also asserts that if a model is not applied with frequency, details may be forgotten. When novices are learning a new concept, they may require several critiques and applications of their model before it becomes appropriately defined and recallable.

"Mental model" is also not a term describing a single model of the world or sometimes even a specific phenomena. Rather, our mental models contain several models of systems and subsystems. These models create a vast ecosystem that encompasses our sum knowledge. Models may be interconnected: a programmer may possess a model of a concept such as loops across many different mental models for various programming languages, which is a part of a larger still model encompassing their understanding of "programming". Concepts may even map to other model systems entirely, such as recognizing the humor in jokes mapping the programming language "Java" and the colloquialism for coffee, "java". It is suggested that expert knowledge is comprised of interrelated networks [29], which is represented well in the interconnection between our mental models.

Another potentially obvious, but important characteristic of mental models is that we may possess more than one mental model of any given concept. We often carry multiple models of a single concept, which may encompass entirely disjoint subconcepts [33].

3.1.2 Fluid and Crystal Intelligence, Working Memory, Cognitive Load. Among theories of intelligence, CHC theory [37] poses that fluid and crystallized intelligence are distinct "bodies" of our intellectual capacities that do not cross. **Crystallized knowledge** is formed from experiences, culture, and education, while **fluid intelligence** is the ability to abstractly reason and adapt [11]. While original assumptions suggested that fluid intelligence was fixed [37], modern neuroscience has suggested training is possible [48, 50, 61].

Advances in fluid intelligence are relevant to **working memory**, as this recognized cognitive system is strongly correlated with fluid intelligence measures [50, 61]. Working memory is a limited but necessary resource in everyday problem solving. The effects of **cognitive load** impact working memory, and can decrease performance. This has been indicated in several studies of programming tasks [1, 46, 59]. The limitations of working memory can result in learners attempting to process too much information, leading to decreased performance as the number of concepts becomes too difficult to reason about.

Models of working memory have shown that task-related information (nodes) are more frequently accessed, and that frequency of access lowers future activation thresholds [43]. Put simply: the more we apply some piece of information, the more likely we are able to easily re-apply it in the future. This fits not only to content knowledge for a specific problem — it is also how we quickly connect topics of personal interest. We find connections to our favorite song precisely because these pathways are so well worn. Such information is easy to access through repetitively accessing it, and thus reduces cognitive load when considering it.

3.1.3 Connecting Knowledge: Pattern Recognition and Analogy. **Pattern recognition** is the process of matching information from a stimuli to known information. We engage in this process when new information is encountered, allowing us to connect this knowledge to our existing models and understanding. Applying existing knowledge to new information requires **abstraction**, an act of "decreasing the specificity (and thereby increasing the scope) of a concept." [24]. This requires attention to be focused on *specific aspects* of the whole, and thus removed from other aspects [18]. **Overhypothesis** abstracts beyond the problem scope specified [24], and is one of the driving forces in our ability to predict outcomes even when we are reasoning about a novel problem space. The processes of pattern recognition and abstraction can be engaged across domain boundaries (cross-domain), so long as specificity is relaxed in order to increase scope.

Thus, **analogical reasoning** helps us develop broader concept categories. Through reasoning about two concepts in order to identify relations between them, we promote the formation of an abstraction that includes both elements. These generalized models we develop constitute **schema**: a general structure of knowledge representing and encapsulating some information [61]. Analogical reasoning may not always be activated in novel contexts [25, 28], but if a pattern is discerned, these processes occur [33]. Even in cases where one may not recognize a pattern and engage in analogical reasoning of their own accord, the invitation or prompting to consider or compare can evoke the process effectively [24, 28].

Analogical reasoning is fully defined as **any type of thinking that relies upon an analogy** [3]. Comparing two very similar and two disjoint concepts uses the same cognitive reasoning faculties. Cross-domain analogy often comes to mind, but reasoned analogs do not need to be from distinct domains. Further, analogies are not exclusively linguistic forms mimicking "X is like Y" [30] — this linguistic format simply represents and invites the abstraction process.

When considering cross-domain analogies, differing **source domains** can allow distinct traits of the phenomena or problem to be highlighted [23]. A source domain represents previously understood knowledge and concepts, which is used to better understand the novel **target domain**. For exploring possible domains, authentic exploration should not place limits hindering the potential value in a source domain [47], and concerns raised by the use of a single source domain can be overcome through additional source domains applied to the same target [56], such as Linder's analogous description of light as both a particle and wave phenomena [40].

In the same way that multiple analogies may allow for exploration of a domain, analogies can be used as bridges to "build" toward an idea. This fits the concept of **progressive alignment**: abstractions with some sort of "concrete" general structure are easier to further abstract [24]. When a schema is appropriately generalized, this promotes schema transfer [25] as the schema applies to more cases, which promotes continued abstraction and reapplication.

A **morphism** is an abstraction formed from a structure-preserving map [30]. Maintaining structure while mapping information between two analogs is key for analogical reasoning to succeed. This premise underlies the theory of **structure mapping** [22], which posits that analogical reasoning processes prefer relationships, not surface aspects. Mapping aspects of X and Y is done through a process called **structural alignment** [24]. This creates an abstracted common structure, which must hold true for both X and Y. **Systematicity**, a process of determining the likely structural relations, occurs during the mapping process as we reason analogically [27].

The concept of "surface" versus "structural" features is considered in relation to goal attainment [28]. What is a surface feature (such as "color") in one scenario may be structural in another. Context is imperative: what we cognitively find relevant relates to the goal and problem space.

Structure mapping and systematicity are key in the adaptation of our mental models. As we reason analogically, we abstract and create more generalized schema, which may be applicable in a greater number of scenarios, allowing us to advance in understanding. As we repeat this process with various stimuli and critiques, our mental models grow and change. We move from the immediate scenarios toward deeper implementations, promoting further connection and broadening our spectrum of associated knowledge [42].

3.2 Examples of STEM Analogy Research

The field of physics contains a wealth of explorations and advancements in educational analogy research. Several studies have explored the use and value of providing multiple source domains in explaining distinct physics phenomena [23, 40, 47]. Additional work has explored progressively bridging analogies to incrementally align student understanding [7, 8]. In fact, many of the papers in the seminal book "Mental Models" [26] are from the field of physics and relate to analogy development/analogical reasoning about phenomena of the field.

Chemistry [55], geology [4], and general sciences [17, 29, 41, 60, 63] have also conducted studies into the use of educational analogy as a classroom activity. The findings among these researchers largely indicate that discussion and exploration of analogy, especially with peers, can increase understanding and capability of learners. Such activities should be done with appropriate guidance in analogy generation and discussion, helping learners avoid overextension or misapplication [17, 60, 63].

3.3 Analogy in Computing Education Research

Chee [6] developed a study in which a single "office" analogy was utilized across all instructional material in two experimental conditions. His findings indicated a weak analogy did not do "more damage" to students than no analogy at all. He also found that the strong analogy condition aided in student's comprehension and composition of programs. Forišek and Steinová [21] developed analogies for algorithms and data structures, and found that the inclusion of a valid metaphor did aid student success when compared to using no metaphor.

Alizadeh et al. [13] incorporated analogy into data structures tutoring sessions, finding that 90% of spoken words were by tutors. Interestingly, when students contributed to these sessions which used analogy, there was a positive effect on learning outcomes.

Harsley et al. [35] developed a computer science tutoring system in which they incorporated three approaches for assisting students working on programming problems: analogy, analogy and a worked out example, or simply a worked out example. This work found that analogy was least beneficial *when learners had prior knowledge of the problem space* [35]. Since the solution of the programming problem was the success metric, a code example would likely provide syntax clues able to be easily replicated in the side-by-side environment used in the study. The study's success metric does not readily indicate effects on learning, understanding, and retention, especially as long-term impact was not explored.

Sanford et al. [52] explored instructional metaphors used by computer science educators. Their work aimed to better understand the metaphors used and their effectiveness. Often, these metaphors classify concepts as physical objects: "return statements are like Harry Potter portkeys", "variables are like boxes", "pointers are like zombies" [52]. They note the need for identification of the limits of an analogy, which these physical object/entity based metaphors exemplify. To relate back to structure mapping: understanding of the *relevant relationships* of the object/entity to the concept is necessary for mapping.

While not specified as analogy, James' [38] work most certainly is. Incorporating Black music and DJing into assignment design for a data structures course, he posed relevant problems from data structures as topical analogs and found enhanced engagement and enjoyment from learners. The course connected target domain knowledge to experiential source domains from a learner's life and prior knowledge, showcasing analogies which increased engagement through interest to the learner.

4 EMBEDDED ATTITUDES, EMERGENT THEMES

While there has been some CER research that makes distinct use of analogy, far more work exists that recommends against the use or exploration of analogy. Given the success of analogy in educational contexts across STEM fields, and successes that we have seen in CER, we explore the reason these misgivings may exist.

4.1 Embedded Attitudes

The negative perception of analogies in computing education research is largely a cultural problem. Attitudes toward analogy are often not rooted in citations or known evidence. Attempting to uncover the source of these beliefs requires an exploration of the evidence in CER's history.

4.1.1 Historic Roots. Edsger Dijkstra was notably antagonistic toward the use of analogy, describing computing as "too novel to be represented by analogy or anthropomorphization" [14]. While computing may be novel, and often includes newly developed subject matter [9], this *furthers* the case for analogy. If computer science as a phenomena cannot be accurately captured by a single analogy, then much like Linder's light phenomena [40], we may need a multitude of analogies in order to understand and represent it. Teaching it as "radical novelty" in the way Dijkstra suggests assumes students can be "blank slates" on the topic. We know students are not blank slates based on our understanding of cognitive processes. Students will generate context and correlations even if we do not provide any. Analogical reasoning is fundamental to our cognition. Dijkstra's dismissiveness of analogy lacks cognitive grounding, and many of his grievances are entirely disparate from analogy or anthropomorphism. However, Dijkstra's prominence within the field of computer science gives credence to his opinion, and this opinion has been a strong undercurrent in the discourse surrounding analogy.

4.1.2 Missing or Misrepresented Context. In their work investigating metaphors used by computer science teachers, Sanford et al. [52] noted that all metaphors break down. This is a true statement, but does not inherently indicate ineffectiveness of analogy as an aid to understanding. Further, the citation associated with this observation is not, in fact, about *all* analogy: it is describing a specific conversation surrounding a non-computer science analogy in which one participant notes that "the analogy breaks down" [44]. Sanford et al. also do not specify this as a weakness [52]. Breakdown can indeed be a strength if understood and used for development and critiques. However, the Sanford paper has been used as an argument by some *against* the use of analogy because of this line!

Halasz and Moran [34] describe the difficulty that can come from using analogy to explain computing concepts. They indicate how features from a physical world analog may be irrelevant to the computing target. Their provocative title evokes Dijkstra, *Analogy Considered Harmful*[1]. Their work cites Gentner and describes structure mapping, but misconstrues the concepts of surface and structural alignment in making their case. They indicate, for example, the "lack of drawers" in a computer file system when a filing cabinet has drawers as fostering an inappropriate analogy. This however, confuses "surface" with "structural" features, which is the very concept Gentner's work explores [22]. The "physical appearance" of the file cabinet is not important when considering the structural relation of *what the filing cabinet does*. It is the structural relationships that are the basis of structure mapping.

Qian and Lehman's [51] discussion of learner misconceptions largely indites analogy as the primary culprit. They back their claim with a citation to Taber, who refers to analogy, alongside models and anthropomorphism, as representations that learners may take too literally [58]. In contrast to Taber's discussion of models which includes visualizations, Qian and Lehman indicate visualizations are *effective* in teaching. Halasz and Moran, just discussed above, are also cited [34] by Qian and Lehman to support their case against analogy. Pinpointing analogy as problematic while advocating for strategies "closely related to" analogy is not exclusive to this work — it is pervasive throughout CER.

Perhaps most surprising considering the widespread use of analogies in computing education despite the myths surrounding their use, the Computing Education Handbook [19] has no index listing for "analogy" or "metaphor". The closest listing being "analogical encoding" which is described in a single paragraph and claims "analogical encoding" is different from "instructional analogy". There is no indication made that analogy is a useful tool in computing education [45]. One can find index entries for many concepts that *use* analogical reasoning — we will show that some of these are well explored in CER later in this paper. Confining analogical processing to a single paragraph and omitting analogy completely may be considered indicative of the perception of analogy in the field.

4.1.3 One Shot, One Opportunity. Relevant to the cognitive overview provided at the onset of this paper, misperceptions about mental models by researchers can also affect perceptions of analogies. Krishnamurthi and Fisler [39] cite Slotta and Chi [53], as well as Gupta et al. [31] in suggesting that updating a "flawed" mental model is more difficult to correct than simply developing a new one. However, Slotta and Chi describe mental models surrounding a specific instance of a physical phenomena that would have been developed long before learning the course material, and in fact argue that in other cases beyond that one, adaptation was feasible and encouraged [53]. Gupta et al. [31] in fact push farther for adaptation being possible, responding that even the case specified by Slotta and Chi should be considered dynamic and adaptable.

The sentiment that flawed mental models are cognitively hard to revise has been shared broadly within CER. This perception suggests that learning is "one shot", and that if a learner's initial model is incomplete or incorrect, that initial model must be discarded and an entirely new model must be forged. Understandably, such a belief would lead to concerns about pedagogical approaches that may lead to "imperfect" models. Hence, the concerns about using analogy. But this "one shot" theory is in direct contrast to the description of our learning processes that cognitive scientists present: that learning is a process of abstraction, re-representation, and generalization.

4.1.4 A Haunted House. Arawjo [2] explored the contextual history of how we came to "write" code. This notion of history and its influence on our present is relevant to the perception in the CER community of analogy. Our citations are situated to the problem context we are working to solve, but our work may be leveraged or cited outside of that context. Prominent figures within our field influence our perceptions and biases, and encourage us to adopt their viewpoints [42]. Cautions about the use of analogy over time become abstracted, skewed, and generalized into myths which are communicated among practitioners, continuing the culture.

4.2 Emergent Themes

Through a review of the literature in CER, we have identified several major themes that emerge as arguments against the use of analogies in computing education. The identification of these themes allows us to consider the development of research studies that may support or refute those concerns, or identify ways to address such issues.

4.2.1 Limitations on Analogy. Analogies fit within a certain contextual frame, and address the similarities of a target domain to a source domain in some instance or form — x behaves like y when z occurs. The concern that a learner will use an analogy beyond its contextual limit has been one of the largest undercurrents in the

[1]While not cited here, this title parallels Dijkstra's widely discussed work within the computer science field at large, *Goto Considered Harmful*

cases made against the use of analogy. Candidate inference is one of the strengths of analogical reasoning — “if x behaves like y, then the result of doing ‘a’ with x should be similar to the result of ‘b’ with y”. This process is used in all forms of analogical reasoning, not just cross-domain. However, the concern of over-extension has been specifically targeted at cross-domain applications, as it can lead to difficulty or confusion when the inference is misguided.

4.2.2 Relevance to Learner. Learners from different sociocultural contexts are likely to have different lived experiences, and even within the same sociocultural sphere, each learner has had different experiences throughout their life. One concern with using analogy is the inability for a learner to have the relevant context to understand the analogy, as it does not come from their lived experiences or understanding of the world.

4.2.3 Didactic Implications. As analogy is typically considered a verbal act (despite analogy existing across mediums), the didactic nature of an instructor analogy provided during lecture may cause difficulties for learners. A learner may not have enough context to compare the elements at all, or may overextend if they cannot effectively contextualize what they have heard. In a lecture environment, a student may not be working on a specific problem, thus they also may not have context for the analogical behavior in an application, and thus may form false assumptions.

4.2.4 Structural Soundness. Limitations of analogy tie into the idea of structural soundness — is the analogy well-formed structurally and conveying the intended concepts through its composition? Analogy creation is often ad-hoc, which precludes devoting time to fully consider its structural soundness. However, the boon of analogies is our tendency towards structural stability through cognitive application of concepts such as systematicity [22].

The question of an analogy being strongly structured — strong enough to relate effectively in context, but also contextualized enough to discourage overextending candidate inference — is ultimately a question of design and analysis of that design.

5 DISCUSSION

In this paper we have shown that cognitive science literature supports analogical theory and consideration of analogy as a viable tool in education. We have also uncovered historical roots that have limited the acceptance of analogy as a viable tool among computing education researchers. Can these opposing perceptions be reconciled?

5.1 CER Research Areas Leveraging Analogy

There are actually many CER areas that *leverage* analogy and make use of analogical reasoning and themes. “Worked out examples” rely on multiple examples that have structural similarity. These examples allow learners to analogically reason about them and to develop a schema in order to solve similar problems. This technique has been shown to be effective [13, 35]. Subgoal labeling [45] identifies component parts of a problem to provide scaffolding and assist learners in developing appropriate schema. Hand traces [10] and visualizations or visual aids [57] are meant to analogously represent the machine’s processes to help learners model and reason about them. All of these methods rely on analogy — and all provide an abundance of new research opportunities if analogy theory is incorporated.

Notional machines are idealized, correct abstractions of a specific process, intended for learners [15, 16, 54]. They are frequently discussed in CER but the term is exclusive to CER. The notional machine mirrors the conceptual model from other STEM disciplines, which are aids used to teach and reason about a system. Such an aid is an analogy for what the conceptual model is teaching about — thus, notional machines are analogies as well.

5.2 Facing the Thematic Fears

Given that several themes beyond cultural ingraining were identified, the logical question is what comes next? How might we navigate moving past the myths and exploring analogy as a viable pedagogical tool withing CER?

5.2.1 Context for Limitations. The analogy “a variable is like a box” has oft been discussed in CER as a failed analogy, due to a multitude of incorrect candidate inferences that can be drawn. Thus, to address analogy limitations, we must ask: *in what situation was this analogy relevant?* What problem are we attempting to solve with this description? Were we attempting to explain how variables can hold information? If so, why describe the entire *concept* of a variable with this analogy? Analogical structures describe relations — a variable is *not* a box, so the comparison must be situational, where the relationships a variable and a box undergo have parallel structures.

The requirement of context and fear of limitations furthers the case for multiple source domains. If a variable behaves like a box in one scenario, and like something else in another, we are less likely to overextend the analogies past their contextualized limitations, as there is not exclusively one source we consider it “to be like”. These multiple sources may interconnect and merge as part of our mental model development, allowing us to form a cohesive understanding of the target as we abstract and re-apply components.

5.2.2 Malleable Domains for Relevance. Analogy must be relevant to the learner, but sociocultural factors can make it difficult if not impossible to create a “one size fits all” analogy. Addressing concerns of relevance requires the ability to adapt source domains to the learner audience, or to provide multiple source domains for a concept. Domain isomorphism would allow for further abstraction as learners explore multiple domains which share structure, progressively aligning a more general schema.

Pedagogical activities and scaffolds which engage active discussion of analogy may promote learners to develop isomorphic domains among peers, and even encourage the finding of relevant domains that evaded the instructor as well.

5.2.3 Dialectic, not Didactic. The literature suggests that analogy use and creation should be guided and discussed, not simply “given”. Analogical reasoning is most effective when instructors act as guides, and learners actively participate with the analogy. As Haglund notes: “it is in the promotion of the critical scrutiny in challenging the analogy, attempting to apply it and recognizing when and why it breaks down that the opportunity for learning really takes place. The focus is in this sense diverted from attempts to find the perfect analogy.” [33]

Students should be encouraged to explore the limitations of classroom presented analogies in order to engage and internalize. Exploring peer and instructor analogies allows learners to become a part of the process. Analogy should not be didactic — it should form a conversation. Instructors can guide inference and exploration through activities and probing questions: "what should happen if we do X?" after an analogy has been provided promotes overhypothesis. Presenting a question promotes discussion and exploration, inviting the comparison and navigation of its implications. Dialectic analogy, alongside questions that challenge structural representations can allow for gap closing and abstraction development.

5.2.4 Well-Formed Analogy for...The Win? A major argument against analogy has been the existence of misapplied or poorly designed analogies. Better informed analogical design through CER can improve the quality of analogy we use, our understanding of their application, and our ability to engage learners in conversations surrounding the implications of the analogy.

Several works promote validation and assessment methods for analogy [20, 28, 29, 36, 62]. These range from assessing the structure of an analogy to its application in the classroom. Designing, analyzing, and critiquing analogy for form is not outside the realm of possibility. However, much of the work identified here is either at a high level without deeper assessment of analogy structure, or targeted to analogy-specific research beyond educational context. The development of guidance and materials that is grounded in these principles may allow educators and researchers to more easily design, identify, critique, and replicate analogies.

6 FUTURE WORK

As this position paper has argued, there is much work that can be done in repairing the perception of analogy within the field of CER. Application and exploration of the ideas in the discussion as research questions and studies is a start.

Analogy is shown to be a powerful tool for learning from both the cognitive and learning perspective and is applied across other STEM disciplines. The study of analogy design and use in our field is important not just for elevating our discussions, but for aiding the students we teach as well. The first author's dissertation [5] begins exploration on designing well-formed analogy in computing — but this is only one small piece. More exploration into the theory of analogy and its application in our field allows for further research avenues, elevated discussions, and as the evidence suggests, potential boons for our students.

7 CONCLUSION

This position paper has argued for the value of analogy as an important cognitive process and learning implement, and has identified sources critical of analogy as a viable pedagogical method within computing education. Contextualization, malleable domains, dialectic approaches, and well-formed structure were presented as methods for combating the arguments against analogy. Fighting the historical roots that have generated myths surrounding the use of analogy can be challenging, but it is possible and necessary for elevating the field of computing education.

REFERENCES

[1] John R. Anderson and Robin Jeffries. 1985. Novice LISP Errors: Undetected Losses of Information from Working Memory. *Hum.-Comput. Interact.* 1, 2 (June 1985), 107–131. https://doi.org/10.1207/s15327051hci0102_2

[2] Ian Arawjo. 2020. To Write Code: The Cultural Fabrication of Programming Notation and Practice. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–15. https://doi.org/10.1145/3313831.3376731

[3] Paul Bartha. 2019. Analogy and Analogical Reasoning. In *The Stanford Encyclopedia of Philosophy* (spring 2019 ed.), Edward N. Zalta (Ed.). Metaphysics Research Lab, Stanford University.

[4] Callan Bentley. 2008. Using Analogies to Assess Student Learning. *Inquiry: The Journal of the Virginia Community Colleges* 13 (2008). https://commons.vccs.edu/inquiry/vol13/iss1/4

[5] Briana Bettin. 2020. The Stained Glass of Knowledge: On Understanding Novice Mental Models of Computing. , 307 pages. https://doi.org/10.37099/mtu.dc.etdr/1086 Open Access Dissertation.

[6] Yam San Chee. 1993. Applying Gentner's Theory of Analogy to the Teaching of Computer Programming. *Int. J. Man-Mach. Stud.* 38, 3 (March 1993), 347–368. https://doi.org/10.1006/imms.1993.1016

[7] John Clement, David E. Brown, and Aletta Zietsman. 1989. Not all preconceptions are misconceptions: finding 'anchoring conceptions' for grounding instruction on students' intuitions. *International Journal of Science Education* 11, 5 (1989), 554–565. https://doi.org/10.1080/0950069890110507 arXiv:https://doi.org/10.1080/0950069890110507

[8] John J. Clement. 1998. Expert novice similarities and instruction using analogies. *International Journal of Science Education* 20, 10 (1998), 1271–1286. https://doi.org/10.1080/0950069980201007

[9] T. R. Colburn and G. M. Shute. 2008. Metaphor in Computer Science. *Journal of Applied Logic* 6, 4 (2008), 526–533. https://doi.org/10.1016/j.jal.2008.09.005

[10] Kathryn Cunningham, Shannon Ke, Mark Guzdial, and Barbara Ericson. 2019. Novice Rationales for Sketching and Tracing, and How They Try to Avoid It. In *Proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE '19)*. ACM, New York, NY, USA, 37–43. https://doi.org/10.1145/3304221.3319788

[11] Ian Davidson and Peter Walker. 2019. Towards Fluid Machine Intelligence: Can We Make a Gifted AI? *Proceedings of the AAAI Conference on Artificial Intelligence* 33 (07 2019), 9760–9764. https://doi.org/10.1609/aaai.v33i01.33019760

[12] Johan de Kleer and John Seely Brown. 1983. Assumptions and Ambiguities in Mechanistic Mental Models. In *Mental Models*. Lawrence Erlbaum Associates, Inc., Chapter 8, 155–190.

[13] Barbara Di Eugenio, Nick Green, Omar AlZoubi, Mehrdad Alizadeh, Rachel Harsley, and Davide Fossati. 2015. Worked-out Examples in a Computer Science Intelligent Tutoring System. In *Proceedings of the 16th Annual Conference on Information Technology Education (SIGITE '15)*. Association for Computing Machinery, New York, NY, USA, 121. https://doi.org/10.1145/2808006.2808011

[14] Edsger W. Dijkstra. 1988. On the cruelty of really teaching computing science. (Dec. 1988). http://www.cs.utexas.edu/users/EWD/ewd10xx/EWD1036.PDF Circulated privately.

[15] Benedict du Boulay. 1989. Some Difficulties of Learning to Program. In *Studying the Novice Programmer*, E. Soloway and J. C. Spohrer (Eds.). Lawrence Erlbaum Associates, 283–299. http://www.sussex.ac.uk/Users/bend/papers/diffsofprogramming.pdf

[16] Benedict du Boulay, Tim O'Shea, and John Monk. 1981. The black box inside the glass box: presenting computing concepts to novices. *International Journal of Man-Machine Studies* 14, 3 (1981), 237 – 249. https://doi.org/10.1016/S0020-7373(81)80056-9

[17] Reinders Duit, Wolff-Michael Roth, Michael Komorek, and Jens Wilbers. 2001. Fostering conceptual change by analogies - Between Scylla and Charybdis. *Learning and Instruction* 11 (08 2001), 283–303. https://doi.org/10.1016/S0959-4752(00)00034-7

[18] John Duncan, Daphne Chylinski, Daniel Mitchell, and Apoorva Bhandari. 2017. Complexity and compositionality in fluid intelligence. *Proceedings of the National Academy of Sciences* 114 (05 2017), 201621147. https://doi.org/10.1073/pnas.1621147114

[19] Sally Fincher and Anthony Robins. 2019. *The Cambridge Handbook of Computing Education Research*. Cambridge University Press. 11–39 pages. https://doi.org/10.1017/9781108654555.002

[20] Kenneth D. Forbus, Dedre Gentner, and Melissa M. Wu. 1997. Towards a Computational Model of Evaluating and Using Analogical Inferences. In *Cognitive Science - COGSCI*.

[21] Michal Forišek and Monika Steinová. 2012. Metaphors and Analogies for Teaching Algorithms. In *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education (SIGCSE '12)*. ACM, New York, NY, USA, 15–20. https://doi.org/10.1145/2157136.2157147

[22] Dedre Gentner. 1983. Structure-mapping: A theoretical framework for analogy. *Cognitive Science* 7, 2 (1983), 155 – 170. https://doi.org/10.1016/S0364-0213(83)

80009-3
[23] Dedre Gentner and Donald R. Gentner. 1983. Flowing waters or teeming crowds: Mental models of electricity. In *Mental Models*, Dedre Gentner and Albert L Stevens (Eds.). Erlbaum, 99–129.
[24] Dedre Gentner and Christian Hoyos. 2017. Analogy and Abstraction. *Topics in Cognitive Science* 9, 3 (2017), 672–693. https://doi.org/10.1111/tops.12278 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/tops.12278
[25] Dedre Gentner, Jeffrey Loewenstein, Leigh Thompson, and Kenneth D. Forbus. 2009. Reviving Inert Knowledge: Analogical Abstraction Supports Relational Retrieval of Past Events. *Cognitive Science* 33, 8 (2009), 1343–1382. https://doi.org/10.1111/j.1551-6709.2009.01070.x arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1551-6709.2009.01070.x
[26] Dedre Gentner and Albert L Stevens. 1983. *Mental Models*. Erlbaum.
[27] Dedre Gentner and Cecile Toupin. 1986. Systematicity and surface similarity in the development of analogy. *Cognitive Science* 10, 3 (1986), 277 – 300. https://doi.org/10.1016/S0364-0213(86)80019-2
[28] Mary L. Gick and Keith J. Holyoak. 1980. Analogical problem solving. *Cognitive Psychology* 12, 3 (1980), 306 – 355. https://doi.org/10.1016/0010-0285(80)90013-4
[29] Shawn M. Glynn, GA. National Reading Research Center, Athens, and MD. National Reading Research Center, College Park. 1994. *Teaching Science with Analogies [microform] : A Strategy for Teachers and Textbook Authors. Reading Research Report No. 15 / Shawn M. Glynn.* Distributed by ERIC Clearinghouse [Washington, D.C.]. 34 p. pages. https://eric.ed.gov/?id=ED373306
[30] Marcello Guarini, Amy T. Butchart, Paul L. Simard Smith, and Andrei Moldovan. 2009. Resources for Research on Analogy: A Multi-disciplinary Guide. *Informal Logic* 29 (2009), 84–197.
[31] Ayush Gupta, David Hammer, and Edward Redish. 2008. The Case for Dynamic Models of Learners' Ontologies in Physics. *Journal of the Learning Sciences* 19 (02 2008). https://doi.org/10.1080/10508406.2010.491751
[32] Mark Guzdial. 2020. Dijkstra Was Wrong About 'Radical Novelty': Metaphors in CS Education. https://cacm.acm.org/blogs/blog-cacm/248985-dijkstra-was-wrong-about-radical-novelty-metaphors-in-cs-education/fulltext
[33] Jesper Haglund. 2013. Collaborative and self-generated analogies in science education. *Studies in Science Education* 49 (03 2013), 1–34. https://doi.org/10.1080/03057267.2013.801119
[34] Frank Halasz and Thomas P. Moran. 1982. Analogy Considered Harmful. In *Proceedings of the 1982 Conference on Human Factors in Computing Systems (CHI '82)*. ACM, New York, NY, USA, 383–386. https://doi.org/10.1145/800049.801816
[35] Rachel Harsley, Nick Green, Mehrdad Alizadeh, Sabita Acharya, Davide Fossati, Barbara Di Eugenio, and Omar AlZoubi. 2016. Incorporating Analogies and Worked Out Examples As Pedagogical Strategies in a Computer Science Tutoring System. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education (SIGCSE '16)*. ACM, New York, NY, USA, 675–680. https://doi.org/10.1145/2839509.2844637
[36] Keith J. Holyoak and Lindsey E. Richland. 2014. Using Analogies as a Basis for Teaching Cognitive Readiness. In *Teaching and Measuring Cognitive Readiness*. Springer Science and Business Media, Chapter 12. https://doi.org/10.1007/978-1-4614-7579-8_12
[37] John L. Horn and Raymond B. Cattell. 1966. Refinement and test of the theory of fluid and crystallized general intelligences. *Journal of Educational Psychology* 57, 5 (1966), 253–270. https://doi.org/10.1037/h0023816
[38] David James. 2020. The Use of DJing Tasks as a Pedagogical Bridge to Learning Data Structures. In *Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE '20)*. Association for Computing Machinery, New York, NY, USA, 193–197. https://doi.org/10.1145/3341525.3387427
[39] Shriram Krishnamurthi and Kathi Fisler. 2019. Programming Paradigms and Beyond. In *The Cambridge Handbook of Computing Education Research*. https://cs.brown.edu/~sk/Publications/Papers/Published/kf-prog-paradigms-and-beyond
[40] Cedric J. Linder. 1993. A challenge to conceptual change. *Science Education* 77, 3 (1993), 293–300. https://doi.org/10.1002/sce.3730770304 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/sce.3730770304
[41] Regina Lipkens and Steven Hayes. 2009. Producing and Recognizing Analogical Relations. *Journal of the experimental analysis of behavior* 91 (02 2009), 105–26. https://doi.org/10.1901/jeab.2009.91-105
[42] Jeffrey Loewenstein. 2017. Structure Mapping and Vocabularies for Thinking. *Topics in Cognitive Science* 9, 3 (2017), 842–858. https://doi.org/10.1111/tops.12280 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/tops.12280
[43] Marsha Lovett, Lynne Reder, and Christian Lebiere. 1999. Modeling working memory in a unified architecture: An ACT-R perspective. *Models of Working Memory: Mechanisms of Active Maintenance and Executive Control* (01 1999).
[44] Sarah J. McCarthey and Elizabeth Birr Moje. 2002. Identity matters. *Reading Research Quarterly* 37, 2 (2002), 228–238. https://doi.org/10.1598/RRQ.37.2.6 arXiv:https://ila.onlinelibrary.wiley.com/doi/pdf/10.1598/RRQ.37.2.6
[45] Briana Morrison, Lauren Margulieux, and Adrienne Decker. 2019. Using Subgoal Labeling in Teaching CS1. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education (SIGCSE '19)*. Association for Computing Machinery, New York, NY, USA, 1237. https://doi.org/10.1145/3287324.3287540
[46] Orna Muller, David Ginat, and Bruria Haberman. 2007. Pattern-Oriented Instruction and Its Influence on Problem Decomposition and Solution Construction. In *Proceedings of the 12th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education (ITiCSE '07)*. Association for Computing Machinery, New York, NY, USA, 151–155. https://doi.org/10.1145/1268784.1268830
[47] Nancy Nersessian. 2008. *Creating Scientific Concepts*. The MIT Press. https://doi.org/10.7551/mitpress/7967.001.0001
[48] Mathilde Neugnot-Cerioli, Charlotte Gagner, and Miriam H. Beauchamp. 2017. Training of fluid and crystallized intelligence: A game-based approach in adolescents presenting with below average IQ. *Cogent Psychology* 4, 1 (2017), 1284360. https://doi.org/10.1080/23311908.2017.1284360 arXiv:https://www.tandfonline.com/doi/pdf/10.1080/23311908.2017.1284360
[49] Donald A. Norman. 1983. Some Observations on Mental Models. In *Mental Models*, Dedre Gentner and Albert Stevens (Eds.). Lawrence Erlbaum Associates, Inc., Chapter 1, 7–14.
[50] Franziska Preusse, Elke van der Meer, Gopikrishna Deshpande, Frank Krueger, and Isabell Wartenburger. 2011. Fluid Intelligence Allows Flexible Recruitment of the Parieto-Frontal Network in Analogical Reasoning. *Frontiers in human neuroscience* 5 (03 2011), 22. https://doi.org/10.3389/fnhum.2011.00022
[51] Yizhou Qian and James Lehman. 2017. Students' Misconceptions and Other Difficulties in Introductory Programming: A Literature Review. *ACM Trans. Comput. Educ.* 18, 1, Article 1 (Oct. 2017), 24 pages. https://doi.org/10.1145/3077618
[52] Joseph P. Sanford, Aaron Tietz, Saad Farooq, Samuel Guyer, and R. Benjamin Shapiro. 2014. Metaphors We Teach by. In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education (SIGCSE '14)*. ACM, New York, NY, USA, 585–590. https://doi.org/10.1145/2538862.2538945
[53] James D. Slotta and Michelene T. H. Chi. 2006. Helping Students Understand Challenging Topics in Science Through Ontology Training. *Cognition and Instruction* 24 (2006), 261 – 289.
[54] Juha Sorva. 2013. Notional Machines and Introductory Programming Education. *Trans. Comput. Educ.* 13, 2, Article 8 (July 2013), 31 pages. https://doi.org/10.1145/2483710.2483713
[55] Lesley Spier-Dance, Jolie Mayer-Smith, Nigel Dance, and Samia Khan. 2005. The role of student-generated analogies in promoting conceptual understanding for undergraduate chemistry students. *Research in Science and Technological Education* 23 (02 2005). https://doi.org/10.1080/02635140500266401
[56] Rand Spiro, Paul J. Feltovich, Richard Coulson, and Daniel Anderson. 1987. Multiple analogies for complex concepts : antidotes for analogy-induced misconception in advanced knowledge acquisition. In *Similarity and analogical reasoning*. Cambridge University Press.
[57] Sangho Suh. 2019. Using Concreteness Fading to Model & Design Learning Process. In *Proceedings of the 2019 ACM Conference on International Computing Education Research (ICER '19)*. ACM, New York, NY, USA, 353–354. https://doi.org/10.1145/3291279.3339445
[58] Keith Taber. 2014. *Alternative Conceptions/Frameworks/Misconceptions*. Springer, 1–5. https://doi.org/10.1007/978-94-007-6165-0_88-2
[59] Vesa Vainio and Jorma Sajaniemi. 2007. Factors in Novice Programmers' Poor Tracing Skills. In *Proceedings of the 12th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education (ITiCSE '07)*. Association for Computing Machinery, New York, NY, USA, 236–240. https://doi.org/10.1145/1268784.1268853
[60] Randy Yerrick, Elizabeth Doster, Jeff Nugent, Helen Parke, and Frank Crawley. 2003. Social interaction and the use of analogy: An analysis of preservice teachers' talk during physics inquiry lessons. *Journal of Research in Science Teaching* 40 (05 2003), 443 – 463. https://doi.org/10.1002/tea.10084
[61] Kun Yuan, Jeffrey Steedle, Richard Shavelson, Alicia Alonzo, and Marily Oppezzo. 2006. Working memory, fluid intelligence, and science learning. *Educational Research Review* 1, 2 (2006), 83 – 98. https://doi.org/10.1016/j.edurev.2006.08.005
[62] Hassan Hussein Zeitoun. 1984. Teaching Scientific Analogies: a proposed model. *Research in Science & Technological Education* 2, 2 (1984), 107–125. https://doi.org/10.1080/0263514840020203 arXiv:https://doi.org/10.1080/0263514840020203
[63] Kevin B. Zook. 1991. Effects of Analogical Processes on Learning and Misrepresentation. *Educational Psychology Review* 3, 1 (1991), 41–72. http://www.jstor.org/stable/23359215