Using *Wright Maps and Learning Progressions* to Engage Teachers and Trainers

Richard S. Brown[1], Mark Wilson[2], Karen Draney[2], Yukie Toyama[2], Perman Gochyyev[2]

[1]National Math & Science Initiative/West Coast Analytics

[2]University Of California, Berkeley

INTRODUCTION

Wright maps are very useful graphical representations when working with Rasch-family models. They can provide psychometricians with information about content validity, by allowing us to examine empirical item difficulty and its relation to predicted difficulty, and about validity based on internal structure, by allowing us to look for banding based on score levels across items. However, Wright maps as they are usually produced by standard IRT software, are not as useful for communicating with non-psychometric clients, such as teachers, or the parents of children in their classrooms. This presentation will focus on modifying Wright maps to allow effective communication with teachers and parents.

Our first endeavor in using Wright maps as a communication tool involved providing reports for an observational assessment of infant-toddler, preschool, and Kindergarten development in California. We were able to use a horizontal presentation to imply a strengths-based interpretation, use the consistency of Thurstonian thresholds for developmental levels across items (sometimes known as banding) to provide a developmental level of interpretation for child locations and avoid providing numerical information that might be over-interpreted (Kriener-Althen et al, 2020), and use a dimensional alignment technique (Feuerstahler & Wilson, 2021) to provide teachers with the ability to compare child growth across dimensions.

Based on our experience working with preschool teachers, we have been developing modified Wright maps for high school teachers, in a project that involves developing measures of college readiness in Computational Thinking. We have focused primarily on providing teachers with views of their entire class with respect to construct levels, ordering students from lower- to higher-level performance, and providing measures of uncertainty (68% confidence intervals).

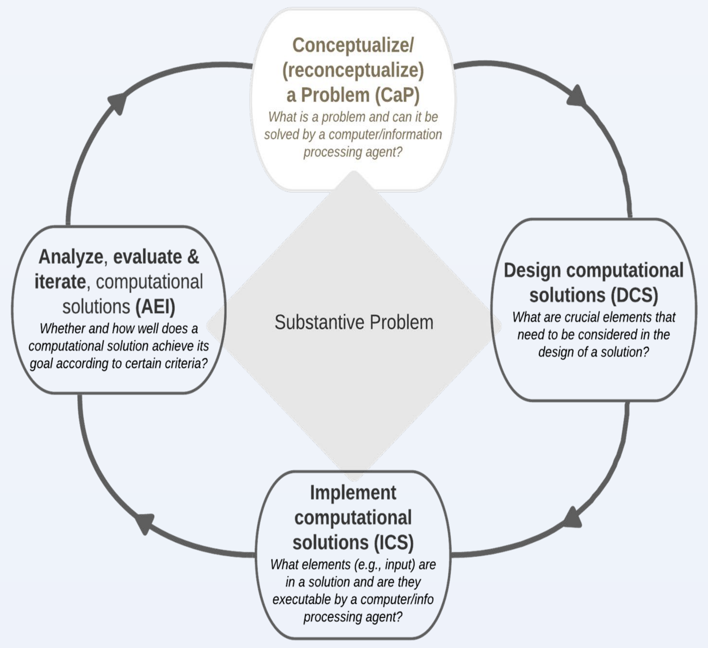
We were recently able to interview teachers and teacher trainers about the use of these maps, along with the descriptions of the learning progressions and sample student responses at each level to find what was more and less useful, and what other features they might like to include. In this presentation, we will discuss the methods used and the feedback we received from the teachers.

OBJECTIVES

The computational thinking framework comprises four dimensions (or constructs):

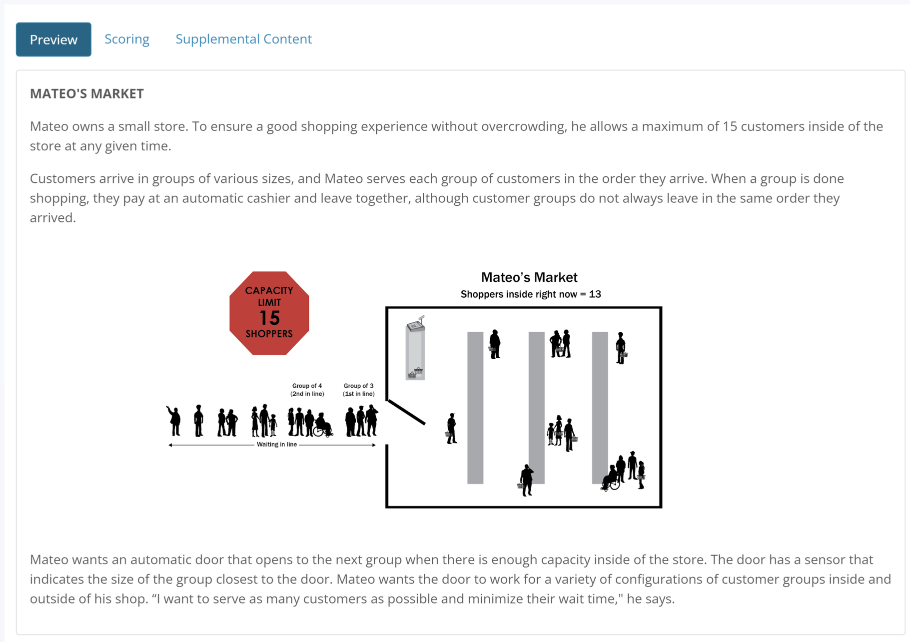
May be viewed as an iterative sequence of steps starting with problem conceptualization and design; however, beginners may start by implementing/modifying existing solutions (Lee et al., 2011).

Each dimension/construct is composed of an underlying continuum in the form of a construct map (Wilson, 2005)



6: Strategic/Step Beyond	• Makes a convincing argument for the best solution to achieve a given purpose through articulating trade-offs among multiple approaches/ competing goals • Fluidly generates multiple (novel/divergent) solutions using multiple evaluation criteria (e.g., accuracy, efficiency, reusability) • Does not prematurely foreclose on known solutions (i.e., avoids rote-application of algorithms), seeking novel/innovative solution(s).
5: Integrated Relational - Complex	• Generates multiple solutions/approaches to a problem using complex operations that require relational understanding among a set of operations • Attends to special situations such as boundary conditions/edge cases • Explains under what condition a particular solution would work • Frames a problem into a familiar task type, by foregrounding certain key aspects while backgrounding less important aspects
4: Integrated Relational - Simple	• Designs a solution/approach to a more complex problem that requires relational understanding of a few subparts that comprise a more complex system/process • Solution includes one or more complex operations/features (e.g., if-then-else, looping, nesting) • Decomposes a complex problem into several related subparts
3: Multi-step Solution	• Designs a solution/approach that requires a sequence of a few subparts through a step-by-step approach, including necessary input(s) and output(s) • Solution may attempt to include at least one complex operation/feature (loop, conditional statement, nesting) • Decomposes a simple problem into a few, discrete subparts
2: One-Step Solution	• Identifies a part of a sequence that can be automated • Identifies a goal, input(s), set of instructions, and/or output(s) in a given context
1: Attempting/Partial	Attempts to design a solution, and uses recognizably appropriate vocabulary, but cannot provide a meaningful response.

MATERIALS AND METHODS



You are a part of a team that must design a computer program for this automatic door.

[MM1a] What pieces of information might your team need to know to design this program?

Write your answer here.

To design an automatic door system for Mateo's store, we need to consider the following requirements:

1. The door should only open when the number of customers inside the store is less than or equal to 15.
2. The door should prioritize the group that has been waiting the longest outside the store.
3. The door should open for a group only if there is enough capacity inside the store for the entire group.

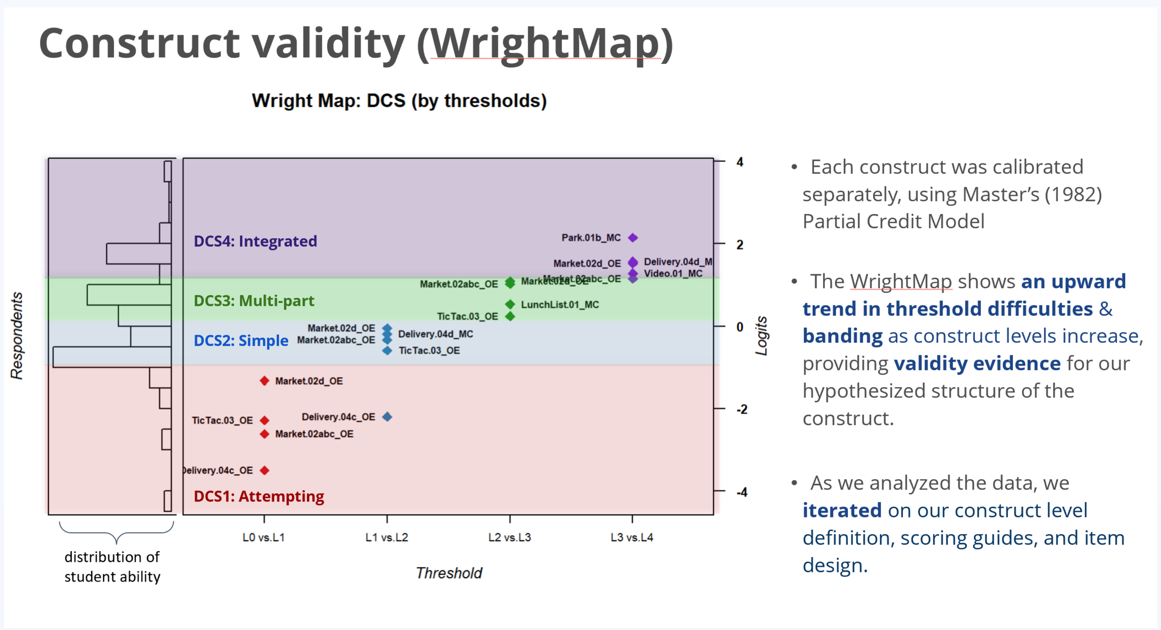
Based on these requirements, we can design an algorithm as follows:

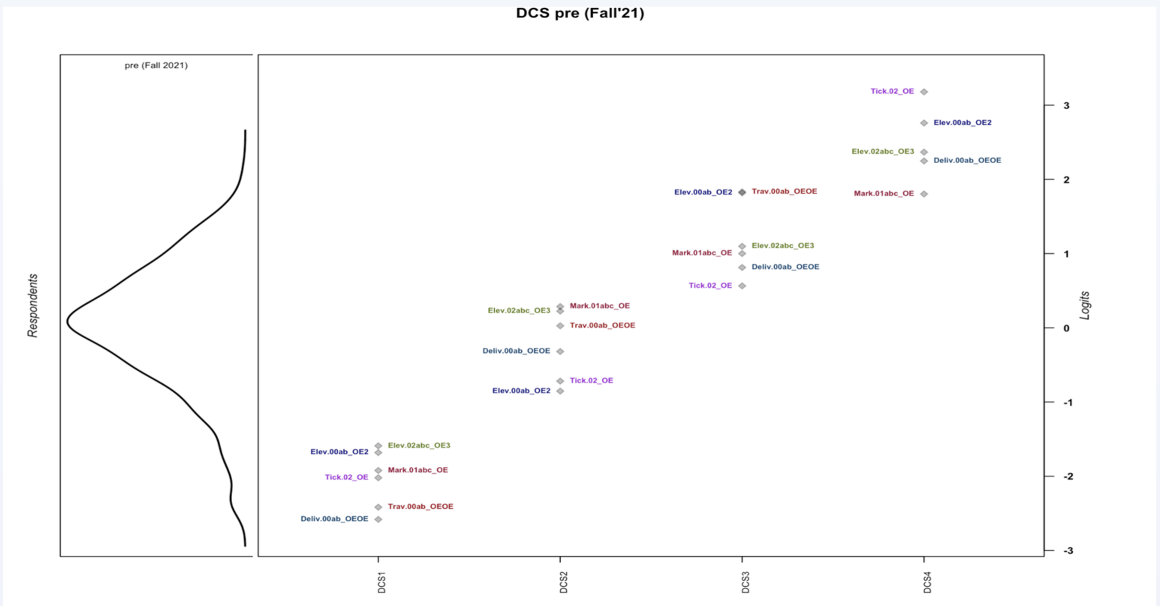
1. Initialize a queue to store the groups waiting outside the store.
2. Initialize a variable to keep track of the current number of customers inside the store.
3. Initialize a variable to keep track of the group closest to the door.
4. Repeat the following steps:
   a. If the queue is not empty and the number of customers inside the store plus the size of the next group in the queue is less than or equal to 15, dequeue the next group and add its size to the current number of customers inside the store.
   b. If the queue is not empty and the number of customers inside the store plus the size of the next group in the queue is greater than 15, set the group closest to the door to the next group in the queue.
   c. If the queue is empty and there are no groups waiting outside the store, set the group closest to the door to None.
   d. If the number of customers inside the store is less than or equal to 15 and there is a group closest to the door, open the door for the entire group closest to the door and remove it from the store.
   e. If the number of customers inside the store is greater than 15 or there is no group closest to the door, do nothing and wait for the next group to arrive.

Scoring Guide

RESULTS

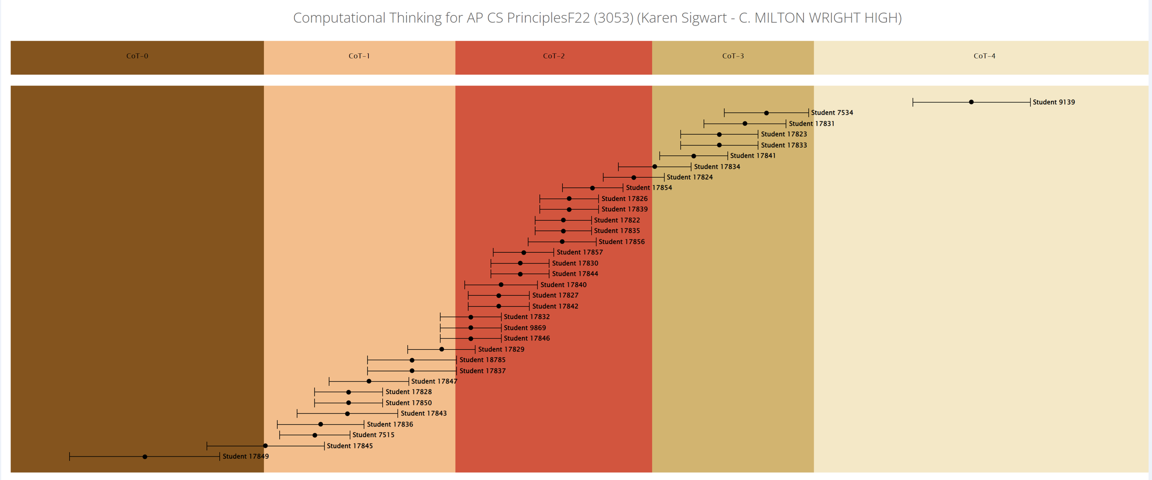
Each construct was calibrated separately, using Master's (1982) Partial Credit Model. The WrightMap shows **an upward trend in threshold difficulties** & **banding** as construct levels increase, providing **validity evidence** for our hypothesized structure of the construct.. As we analyzed the data, we **iterated** on our construct level definition, scoring guides, and item design.

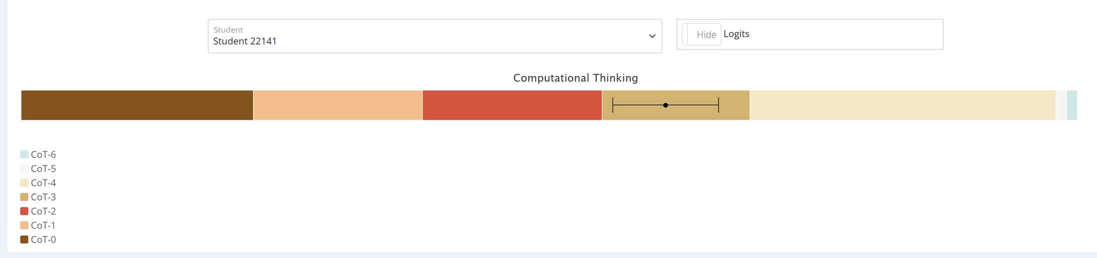


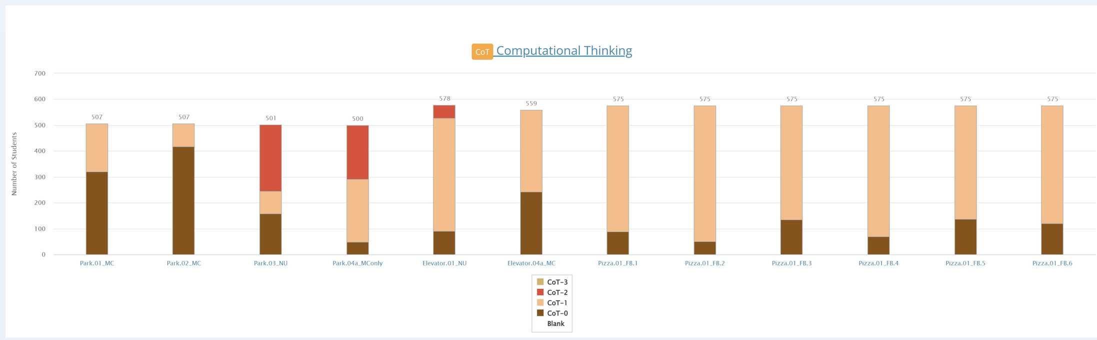


Score Reports

Teachers are provided a series of reports upon assessment completion, include student answers to items and estimated student proficiencies at both group and individual levels.

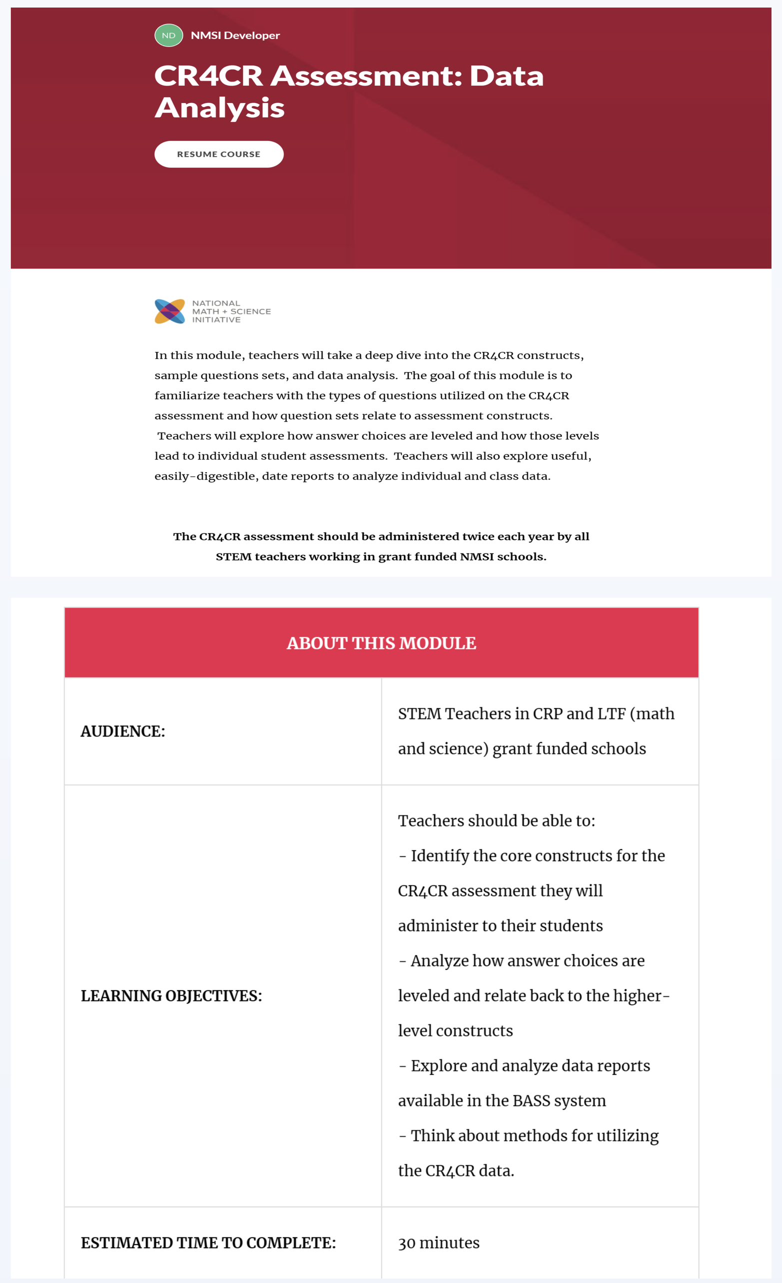






CONCLUSIONS

National Math + Science Initiative (NMSI) Trainers developed a training module for teachers to better utilize and understand the assessment.



Teacher and Trainer Feedback

What is useful:

- Understanding the differences in performance levels in terms of what students can do at each level
- **"It certainly helps me to fully understand the assessment constructs and reports as well as connect to NMSI resources we are designing" – NMSI Trainer**
- The estimation of proficiency by construct in addition to strand
- The immediacy of item specific performance reports
- Class level proficiency reports for grouping of students
- Item Scoring Guides and sample student responses at each level (see examples)

What is needed:

- Real-time proficiency reports
- Linking of frameworks, items, and/or proficiency level abilities to state standards or learning objectives
- Trainer developed strategies to improve student progression from each performance level to the next by item/construct

REFERENCES

Feuerstahler, L. M., & Wilson, M. (2021). Scale alignment in the between-item multidimensional partial credit model. *Applied Psychological Measurement*, *45*(4), 268–282. https://doi.org/10.1177/01466216211013103

Kriener-Althen, K., Newton, E., Draney, K., & Mangione, P. (2020). Measuring readiness for kindergarten using the Desired Results Developmental Profile. *Early Education and Development*, https://doi.org/10.1080/10409289.2020.1743160.

Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., et al. (2011). Computational thinking for youth in practice. Acm Inroads, 2(1), 32-37.

Masters, G. N. (1982). A Rasch model for partial credit scoring. *Psychometrika*, *47*(2), 149–174.

Wilson, M. (2005). *Constructing measures: An item response modeling approach*. Psychology Press, Taylor & Francis Group.

ACKNOWLEDGEMENTS

This project is funded by the National Science Foundation, grant # 2010314. Any opinions, findings, and conclusions or recommendations expressed in these materials are those of the authors and do not necessarily reflect the views of the National Science Foundation.