Collaborative, Computational STEM Learning Environment

Gautam Biswas, Akos Ledeczi, Nicole Hutchins, Miklos Maroti, Naveed Mohammed, Brian Broll; **Vanderbilt University;** Kevin McElhaney, Satabdi Basu, Christopher Harris, Carol Tate, **SRI International;** Luke Conlin, **Salem State University**; Dan Schwartz, Kristen Blair, Doris Chin, Rachel Wolf, **Stanford University**; Shuchi Grover, **Independent Researcher;** Shannon Campe, **ETR**

STEM+C PI Meeting

Project Overview

Employs **novel computational paradigm** that combines

- Visual programming
- Domain specific modeling languages (DSMLs)
- Learning by Modeling and Simulation

for **synergistic STEM + CT Learning** in K-12 classrooms

C2STEM Highlights

- ✓ **Challenge-based, evidence-centered design** of STEM curricula to meet NGSS & state science standards
- ✓ **Low threshold, wide walls, high ceiling**: accomplished using domain-specific block structured languages to support learning
- ✓ **Coupled multi-level representations** to support learning: conceptual modeling & inquiry components offer new forms of exploring & decomposing STEM domain
- ✓ **Synergistic Learning**: emphasis on integrating CT with existing science curricula – complements CS4All programs
- ✓ **Simultaneous assessments for STEM & CT**: Utilize ECD & PFL assessments for studying learning performance and behaviors
- ✓ **Collaborative model building** to support interaction & problem-solving skills
- ✓ **Involve teachers** in curriculum development and support for classroom activities

C2STEM Instructional Design

High school Physics Honors Kinematics + Mechanics

Learning by Modeling – learn Physics by building simulation models of physical processes (e.g., movement of objects)

- Step by step modeling approach (introduce students to concept of simulation step; relations expressed in Δt increments)
- Creates synergistic learning opportunities

Evidence-Centered Design (ECD) applied to developing curricular modules and embedded assessments

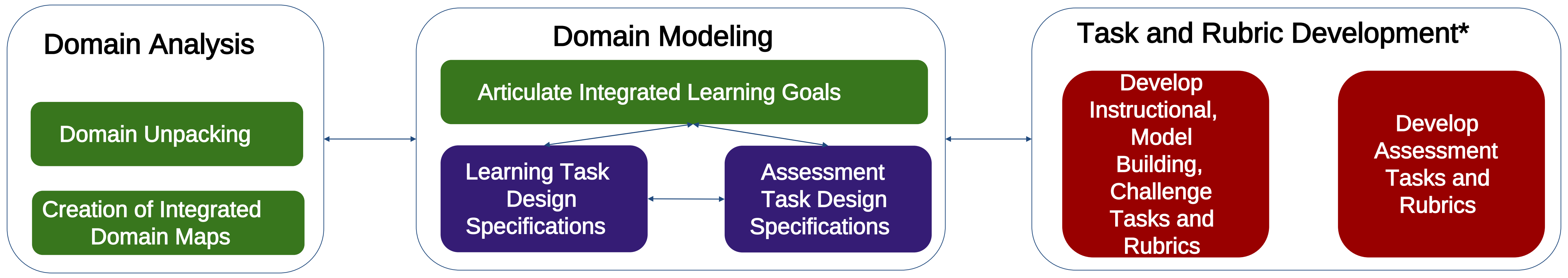
- Align with NGSS standards and classroom curricula
- Create effective synergistic learning opportunities through embedded assessments

Preparation for Future Learning (PFL) assessments to study transfer of learning

Classroom Activities in C2STEM

- ***Instructional Tasks*** - highly scaffolded with the goal of focusing student attention on the learning and application of primary Physics & CT concepts, often one at a time
- ***Model Building*** – students apply their learned Physics + CT concepts to build computational models of relevant Physics phenomena.
- ***Formative Assessments*** – Assess student learning with multiple choice, short answer questions, & small computational model building exercises.
- ***Challenge Problems*** – comprehensive; test students' abilities to put together all concepts & practices to build a computational model that solves a difficult problem.

Principled-Design Process



Domain Analysis

- Domain Unpacking
- Creation of Integrated Domain Maps

Domain Modeling

- Articulate Integrated Learning Goals
- Learning Task Design Specifications
- Assessment Task Design Specifications

Task and Rubric Development*

- Develop Instructional, Model Building, Challenge Tasks and Rubrics
- Develop Assessment Tasks and Rubrics

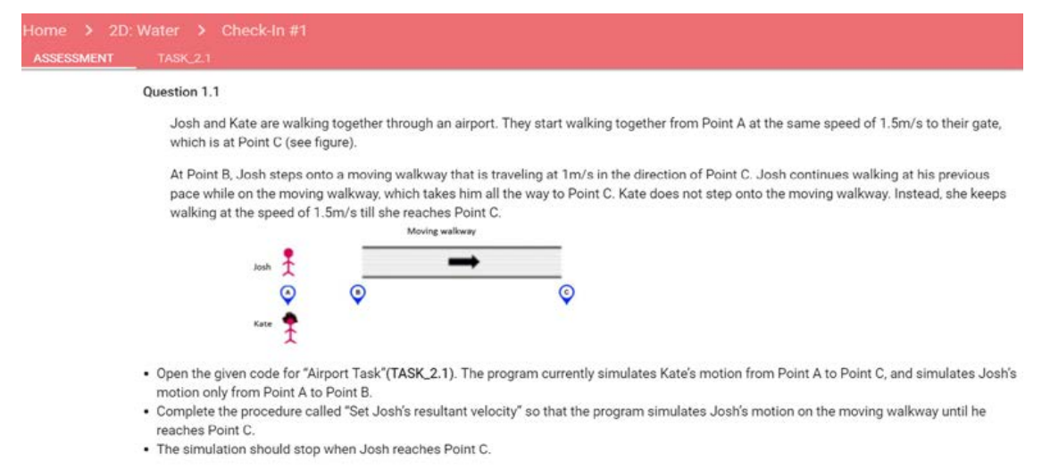
Synergistic Learning Tools

Computational Modeling

- Key to synergistic learning of STEM + CT
- Uses coupled representations
  - Conceptual Modeling used to frame computational modeling problem
  - Computational model constructed from domain-specific blocks
- Model Execution structure
  - Initialization block + Run block
  - Step-by step model (simulation step, Δt)
- Visualize model execution
  - Animation, inspecting variable values, generating plots

Domain-Specific STEM Programming Blocks provide the grounding for Synergistic Learning

Embedded Assessments



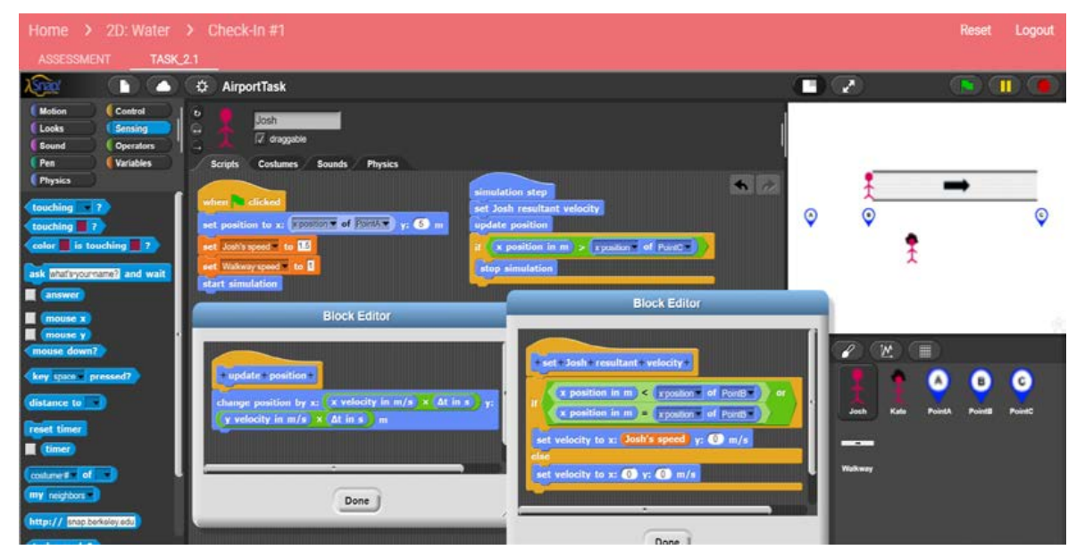
Home > 2D: Water > Check-In #1
ASSESSMENT TASK_2.1

Question 1.1

Josh and Kate are walking together through an airport. They start walking together from Point A at the same speed of 1.5m/s to their gate, which is at Point C (see figure).

At Point B, Josh steps onto a moving walkway that is traveling at 1m/s in the direction of Point C. Josh continues walking at his previous pace while on the moving walkway, which takes him all the way to Point C. Kate does not step onto the moving walkway. Instead, she keeps walking at the speed of 1.5m/s till she reaches Point C.

- Open the given code for "Airport Task"(TASK_2.1). The program currently simulates Kate's motion from Point A to Point C, and simulates Josh's motion only from Point A to Point B.
- Complete the procedure called "Set Josh's resultant velocity" so that the program simulates Josh's motion on the moving walkway until he reaches Point C.
- The simulation should stop when Josh reaches Point C.



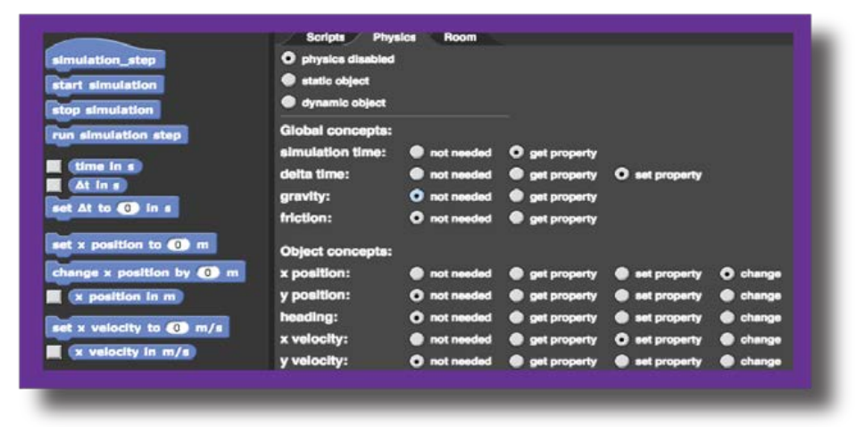
Example task and rubric addressing the integrated learning goal: Develop a computational model that simulates 1-D, constant velocity motion using addition of velocity vectors that occur only under particular conditions.

Integrated assessments tasks using task design specifications

- Addresses
  - Physics domain concepts
  - Program development and debugging
- Rubric
  - distinguishes how well students express physics relationships from their use of programming concepts in the computational model

Criterion #	Rubric for scoring the Airport task	Points
	Expressing physics relations in a computational model (physics component): 2 point rubric	
1	Program expresses correct relations among velocity, position and time, and correct units for each	1 point
2	Program reflects that walking on the moving walkway causes resultant speed to be additive in the x direction (walking speed +walkway speed) and constant (no acceleration)	1 point
	Using programming concepts to model physics phenomena (CT component): 4 point rubric	
3	Program makes the distinction between actions that need to happen once during initialization and actions that need to be repeated in the simulation step	1 point
4	Program correctly determines which action always happens and which happens under certain conditions	1 point
5	Program updates the variable corresponding to Josh's velocity on the walkway a. under the correct conditions (Use conditionals with appropriate expressions to update Josh's velocity under correct conditions between Point B and Point C only), and b. in the correct fashion (the x velocity is set to a new constant value instead of changing at every simulation step)	1 point
6	All code in the program is reachable and can be executed	1 point

Conceptual Modeling supports Computational Modeling: enables Planning, Problem Formulation, and Problem Decomposition!



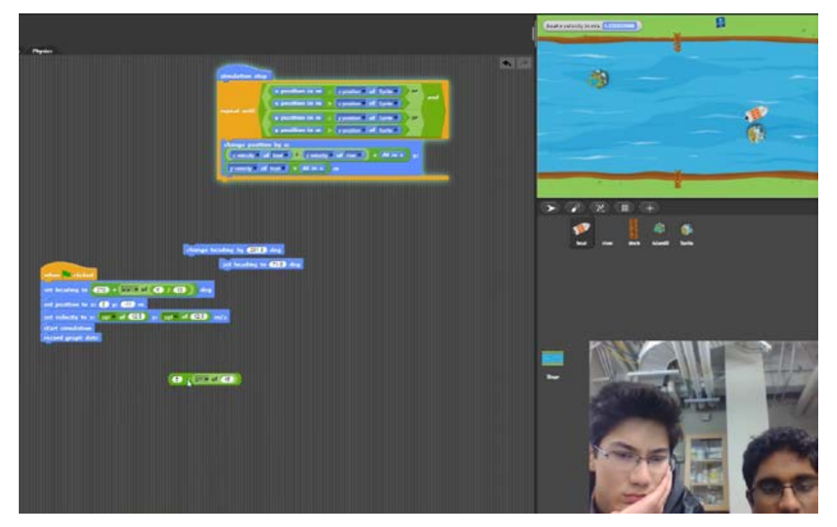
Enhancing Inquiry

From racing sloths to conducting experiments on the effects of gravity, we have added unique inquiry tools for engaging and motivating STEM learning. Prior to building their own simulations, students can run tests, use scientific tools (e.g., our graphing component), and compare results with expert model code to inspire powerful ideas!

Collaborative Problem Solving

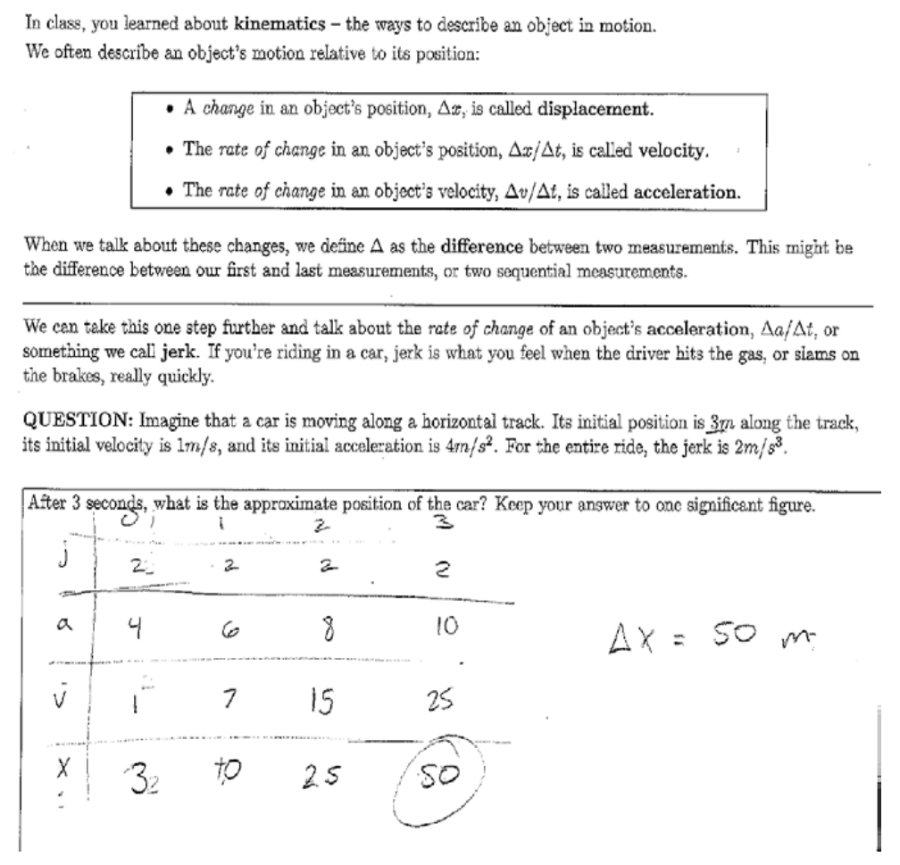
Model Building and Problem solving in groups of two or three

- Students share workspace
  - Create shared representations (models) and shared understanding
  - Socially shared-regulation supports task understanding, planning, strategy use, monitoring, overcoming difficulties, and motivation



PFL Assessments

- **Preparation for Future Learning (PFL) assessments** provide opportunities to learn *during* the assessment
- PFL measures focus on students' ability and propensity to apply computational constructs and CT practices while learning new STEM topics within and outside of kinematics.
- Exemplar targets the kinematics concept of jerk



In class, you learned about **kinematics** – the ways to describe an object in motion. We often describe an object's motion relative to its position:

- A *change* in an object's position, Δx, is called **displacement**.
- The *rate of change* in an object's position, $\Delta x/\Delta t$, is called **velocity**.
- The *rate of change* in an object's velocity, $\Delta v/\Delta t$, is called **acceleration**.

When we talk about these changes, we define Δ as the **difference** between two measurements. This might be the difference between our first and last measurements, or two sequential measurements.

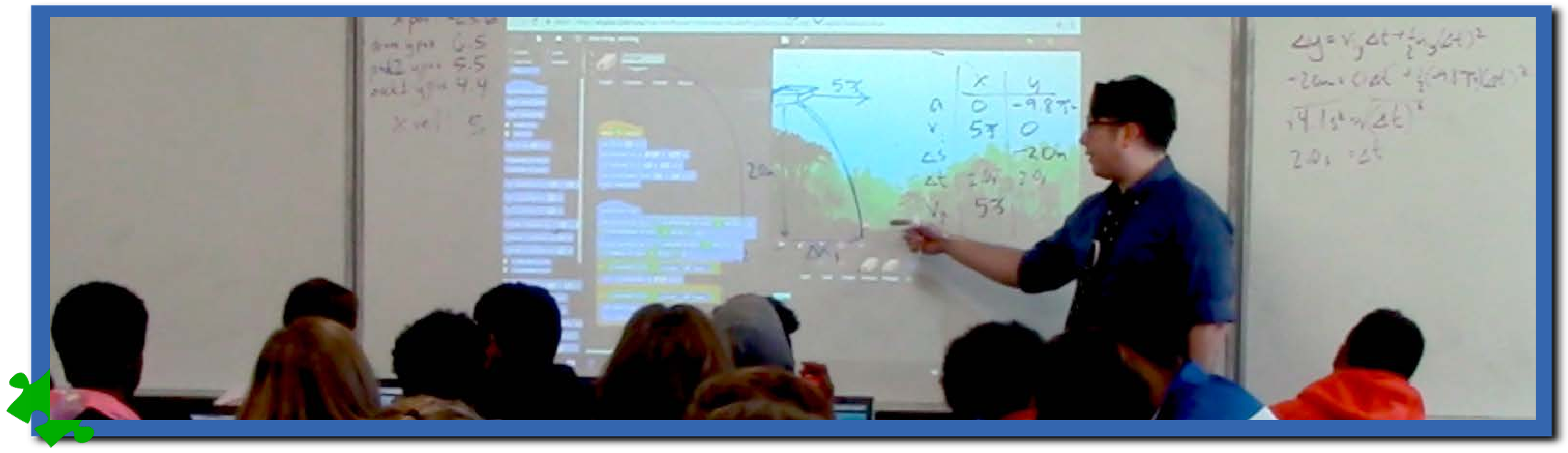
We can take this one step further and talk about the *rate of change* of an object's acceleration, $\Delta a/\Delta t$, or something we call **jerk**. If you're riding in a car, jerk is what you feel when the driver hits the gas, or slams on the brakes, really quickly.

QUESTION: Imagine that a car is moving along a horizontal track. Its initial position is $3m$ along the track, its initial velocity is $1m/s$, and its initial acceleration is $4m/s^2$. For the entire ride, the jerk is $2m/s^3$.

After 3 seconds, what is the approximate position of the car? Keep your answer to one significant figure.

	0	1	2	3
j	2	2	2	2
a	4	6	8	10
v	1	7	15	25
x	3	10	25	50

$\Delta x = 50$ m

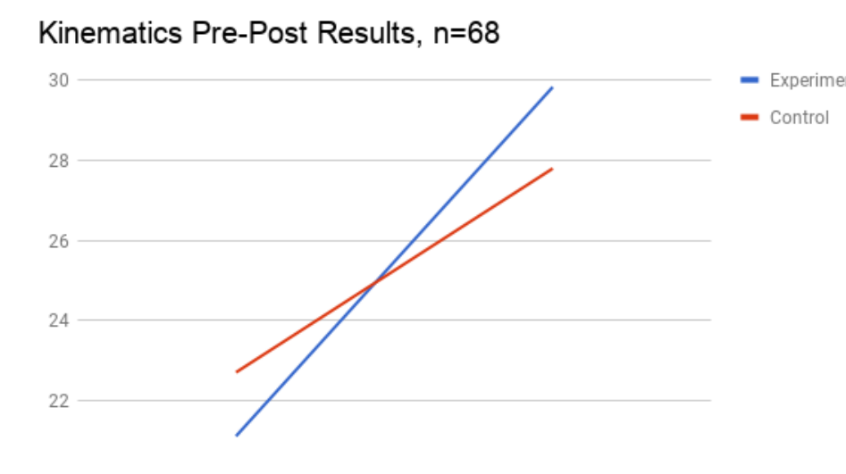


Classroom Studies

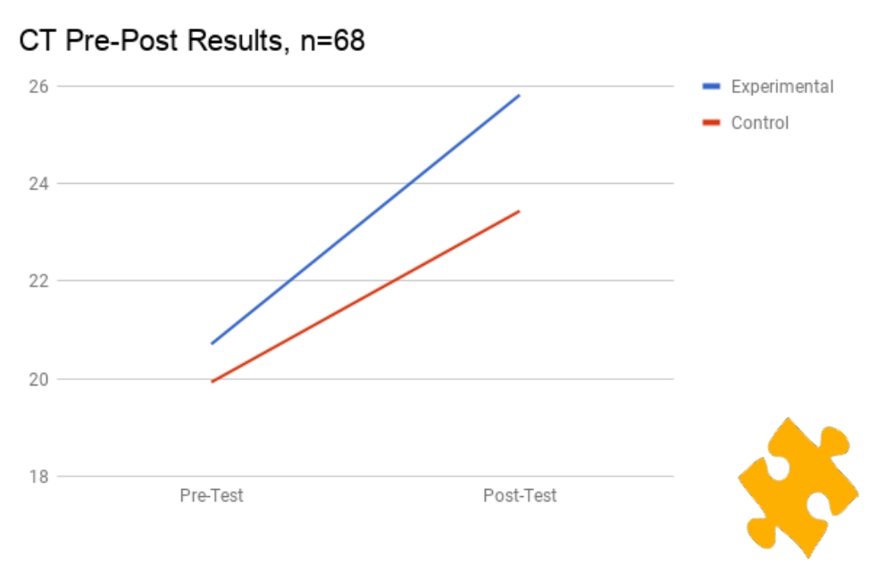
Preliminary results from studies in high school Physics classrooms in Nashville, Tennessee.

Multiple Analyses: Pre-Post, Embedded Assessments, Video Analyses, PFL Assessments

Pre-Post Kinematics + CT Results, Classroom Study



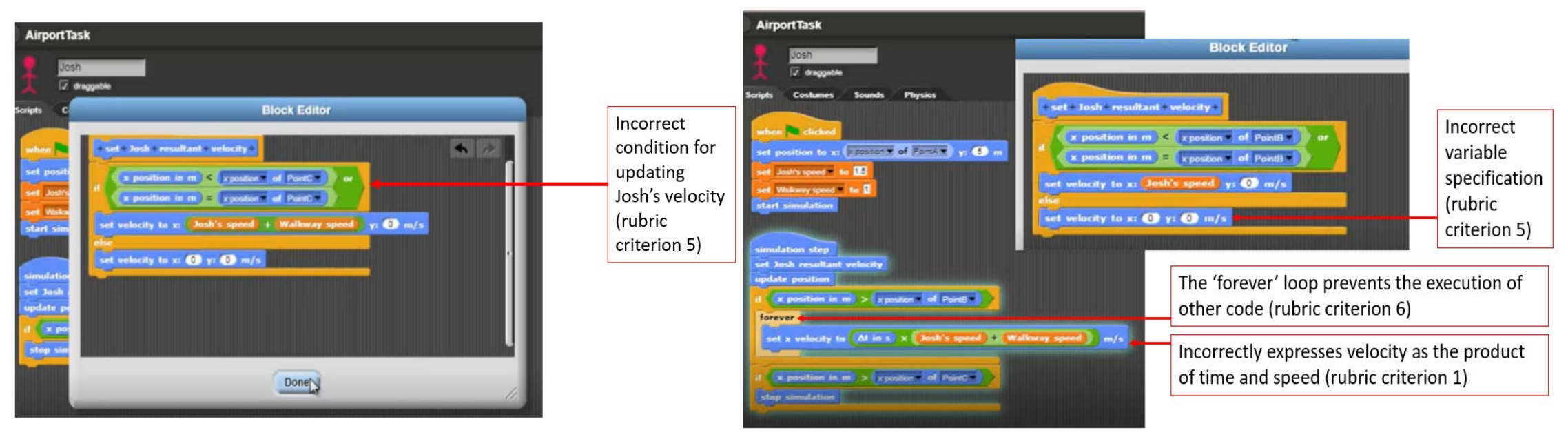
Learning Gains p-value: 0.073



Learning Gains p-value: 0.045

Embedded Assessment Cases, Pilot Study

Example solutions illustrating challenges students have with integrating physics and CT in a computational modeling task.



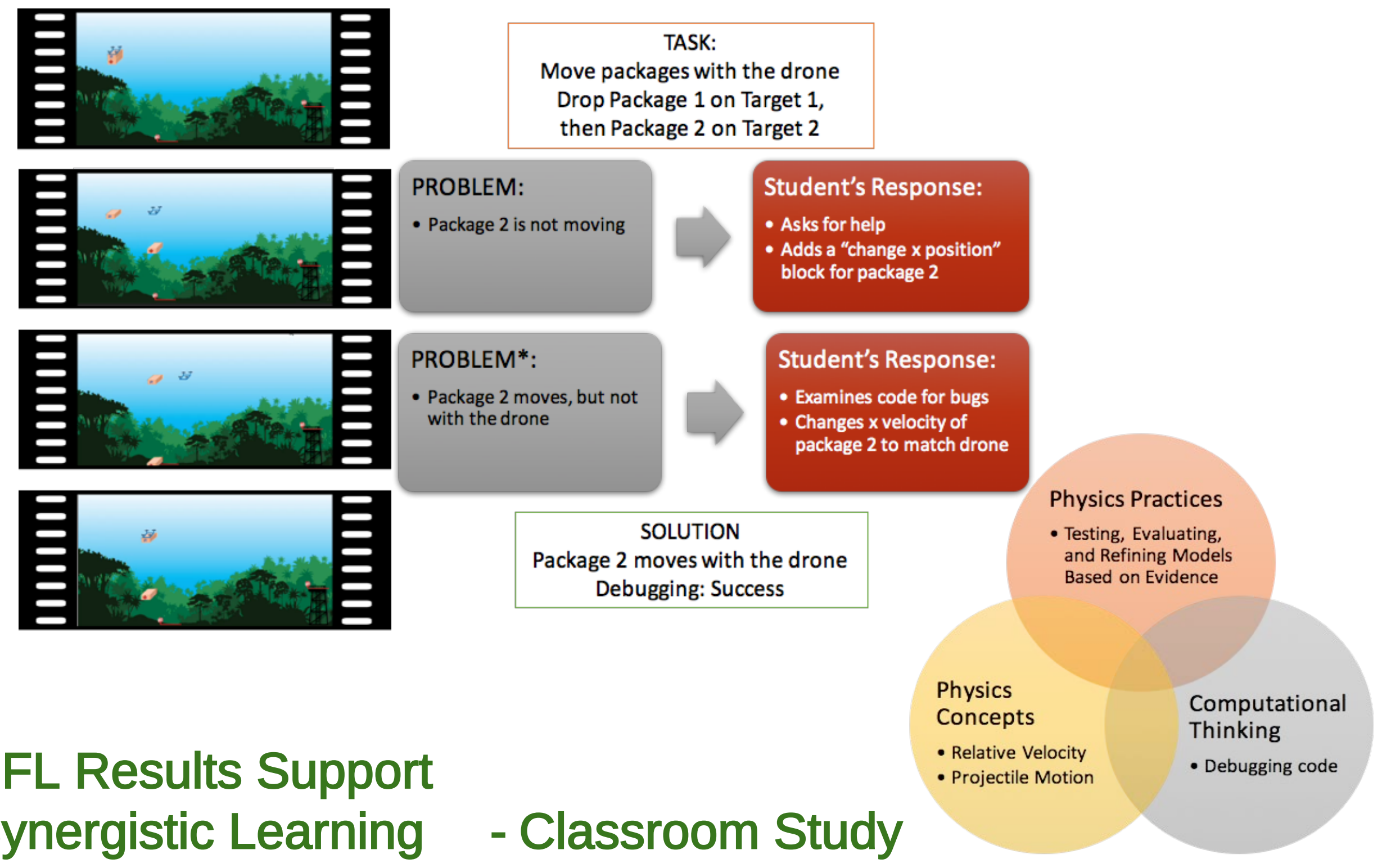
Incorrect condition for updating Josh's velocity (rubric criterion 5)

Incorrect variable specification (rubric criterion 5)

The 'forever' loop prevents the execution of other code (rubric criterion 6)

Incorrectly expresses velocity as the product of time and speed (rubric criterion 1)

Video Analysis of a Students' Debugging Process, Classroom Study



TASK:
Move packages with the drone
Drop Package 1 on Target 1,
then Package 2 on Target 2

PROBLEM:
- Package 2 is not moving

Student's Response:
- Asks for help
- Adds a "change x position" block for package 2

PROBLEM*:
- Package 2 moves, but not with the drone

Student's Response:
- Examines code for bugs
- Changes x velocity of package 2 to match drone

SOLUTION
Package 2 moves with the drone
Debugging: Success

Physics Practices
- Testing, Evaluating, and Refining Models Based on Evidence

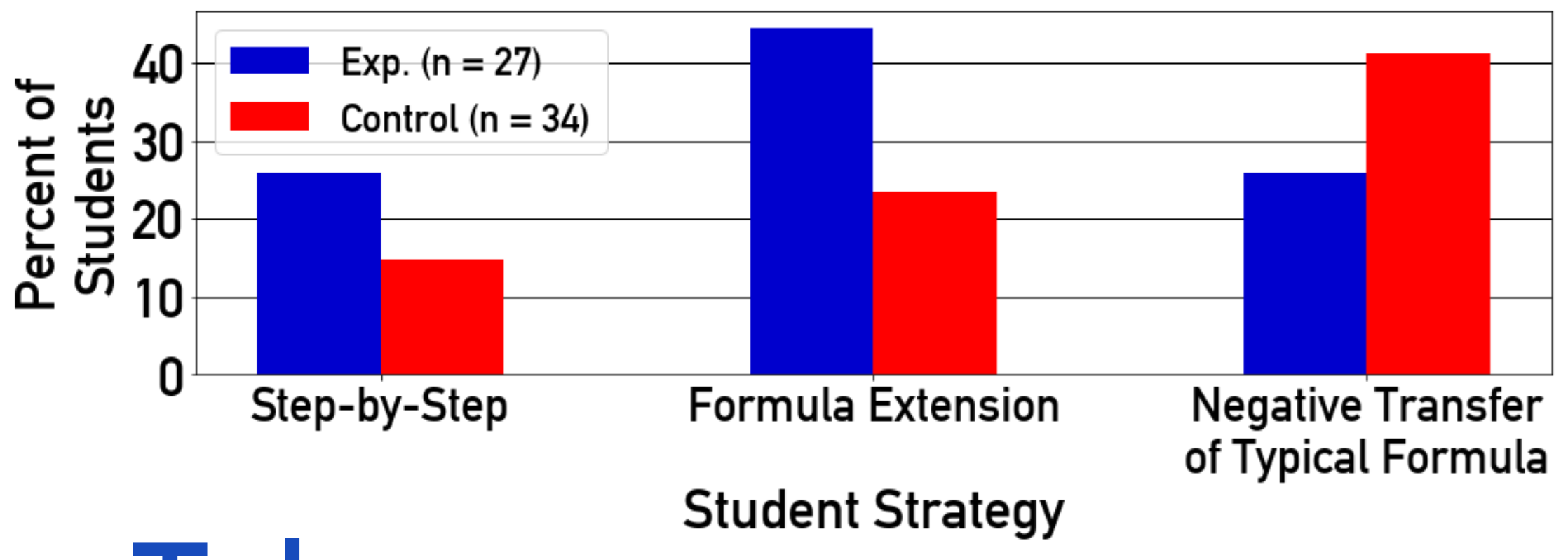
Physics Concepts
- Relative Velocity
- Projectile Motion

Computational Thinking
- Debugging code

PFL Results Support Synergistic Learning - Classroom Study

Responses were initially coded for problem-solving approaches:

- "Step-by-step" - analyzing the problem in iterative, discrete time intervals, e.g., finding acceleration, velocity, and position at $t = 1, 2, 3$ s
- Extension of known kinematics formulae - attempting to modify an existing formula to account for jerk
- Negative direct transfer of kinematics formulae - application of existing formulae without accounting for jerk



Takeaways

- A principled evidence-centered design process supported the alignment of curriculum and formative assessments
- System integrated well with classroom instruction
- Initial analyses provide evidence for the synergistic learning of Physics + CT concepts and practices
- Students were able to overcome difficulties in understanding relations between velocity and acceleration, relative velocities, gravity as acceleration, resultant forces, & frictional forces
- Debugging, an important CT practice – supported model building

This material is based in part upon work supported by the National Science Foundation under NSF Award DRL-1640199. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.