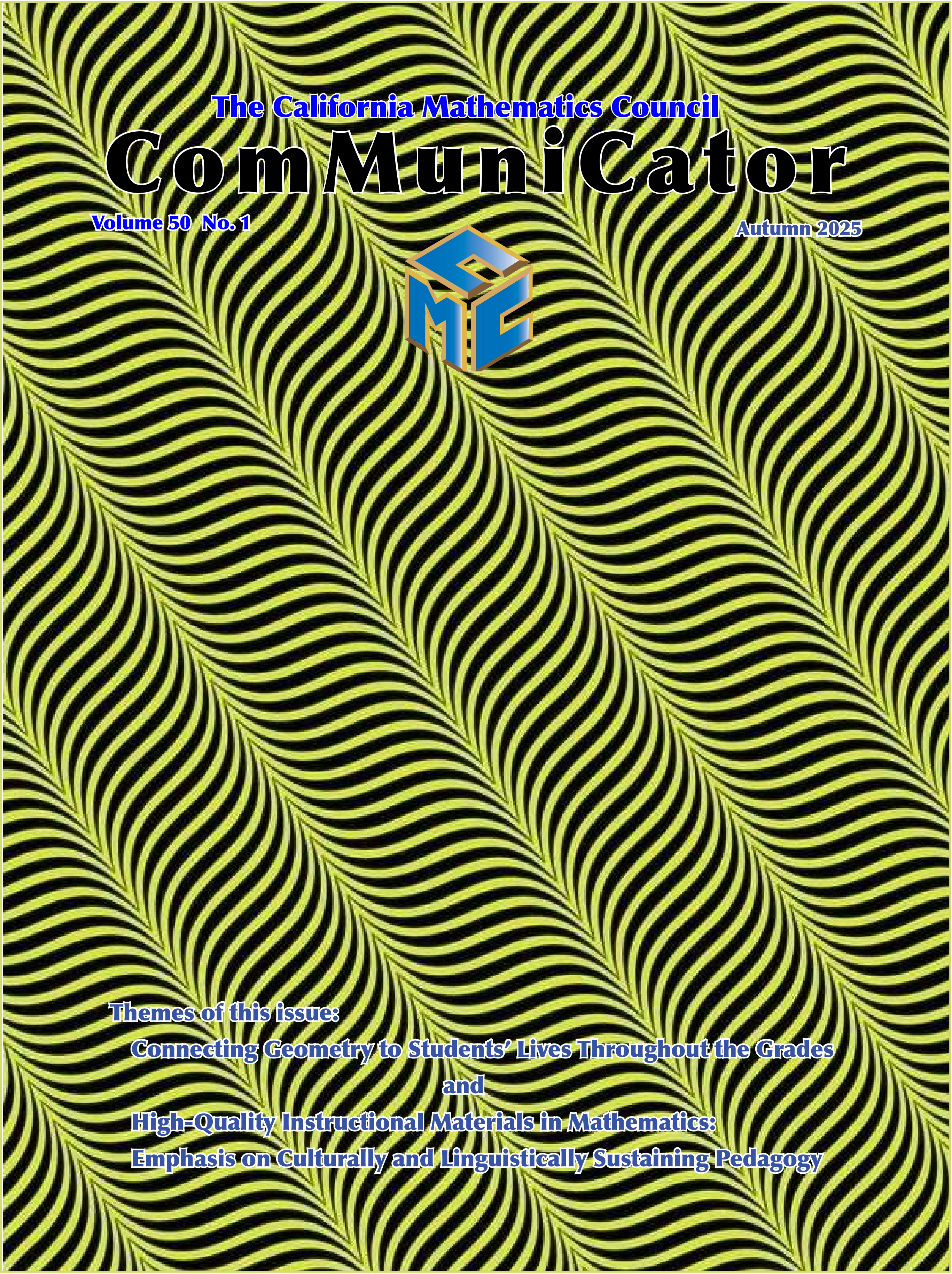

The California Mathematics Council

ComMuniCator

Volume 50 No. 1

Autumn 2025

Themes of this issue:

Connecting Geometry to Students' Lives Throughout the Grades

and

High-Quality Instructional Materials in Mathematics:

Emphasis on Culturally and Linguistically Sustaining Pedagogy

Dataflow Geometry: Advanced Geometry with Computational Thinking

by Pierre Bierre, Spatial Thoughtware; pierre@DataflowGeometry.org;
and Gregory Duran, District Math Coordinator, San Ramon Valley USD, retired

Students grow up surrounded by embedded geometry artifacts such as three-dimensional (3D) animation, Global Positioning System (GPS), and Google Earth, although they probably have not thought about the geometry within these. Students in California high schools should be able to understand the mathematical underpinnings of such everyday things in their lives. We are two mathematics educators who crossed over from technology (scientific software, electrical engineering) and brought to the classroom the perspective of how applied mathematics in industry has evolved into an interdisciplinary mathematics-computer science approach that educators now call computational thinking (Wing 2006). NCTM supports blending computational thinking (CT) into mathematics education (Perez 2018).

We will need CT to keep mathematics up-to-date, relevant, and fully engaging for 21st-century students. Through our research, we envision mathematical competency as the ability to solve a problem on paper and then translate it into an automated algorithm. We want to share our classroom success story based on our research, supported by a grant from Lawrence Livermore National Laboratory (Kagarakis 2011.) We invite you to experience the newest incarnation of that learning experience online at **youtu.be/xzCElLwM25w**

We designed a "C" Advanced Mathematics course, Algorithmic Geometry w/ Java, and conducted two pilot years teaching pre-STEM 11th–12th -graders (Bierre 2010, 2012). Considerable thought went into selecting geometry constructs that carry over smoothly into software to help novices write bug-free code. We based our instructional model on transitioning students from two-dimensional (2D) high school plane geometry to 2D and 3D vector geometry, leading to understanding technology such as GPS and Google Earth.

Algorithmic Geometry students faced various applications where they applied this knowledge:

- ✔ Apollo Mission Navigation: Given the radio-signal distances s1, s2, s3 of Apollo to each of the 3 Deep Space Network receivers at geolocations g1, g2, g3, calculate Apollo's current position in geocentric Cartesian coordinates.
- ✔ Plane Down: Specify the geolocation where to begin Search and Rescue for an international flight that lost radio contact 37% of the flight duration from Rio de Janeiro to Paris, given the geolocations of the two airports, and assuming a Great Circle, constant velocity flight path.
- ✔ Alpha Centauri Navigation (star tracking): Using camera-based directional sightings d1, d2, d3 of three reference stars + a star chart giving their galactic Cartesian-coordinate locations p1, p2, p3, calculate the current position of the starship.

A New Workstyle with Remarkable Mental Efficiencies

Every problem solved becomes useful as an automated algorithm, which can be applied to developing a solution to more complex problems. This type of solution piggybacking depends upon posing each geometry problem in its most general form (i.e., spanning all possible cases of input). The advantage of solving the general case in an automated format is that we can apply the solution to additional contexts.

Solution piggybacking helps utilize every previous problem solved as a tool. Once you solve and test your solution for correctness, you add the algorithm to your pile of tools, readying you for more complex problems. This helped our high school students show great productivity in the 145-hour mathematics course. However, this depended on having a Java expert in the classroom daily. Since this limitation made the program difficult to replicate, we revised the toolset by eliminating the

Java programming component and replaced it with a modular graphical user interface (GUI).

A New Alternative to Programming Languages

To broadened the participation of schools, teachers, and students, we replaced programming languages with modular GUIs such as Scratch (Resnick 2009). We prototyped a modular dataflow GUI, a simple way to link modules that perform desired computations. The design worked well enough to revise the first-semester 2D Geometry + CT package. Branded **DataflowGeometry2D**, we recently released a free, self-paced, online course (software and video tutorials) for curious academics, administrators, teachers, and students.

Ready for the Next Phase: DataflowGeometry2D

DataflowGeometry2D works similarly to Scratch, and we designed it explicitly for mathematics students learning advanced geometry. For example, look at how a student solves an entry-level problem: determining the midpoint of a given pair of points, p1 = [x1, y1], p2 = [x2, y2]. After developing a solution on paper, the student is ready to use the software. ***Figure 1*** illustrates the interactive computer graphics supporting problem visualization and solution testing.

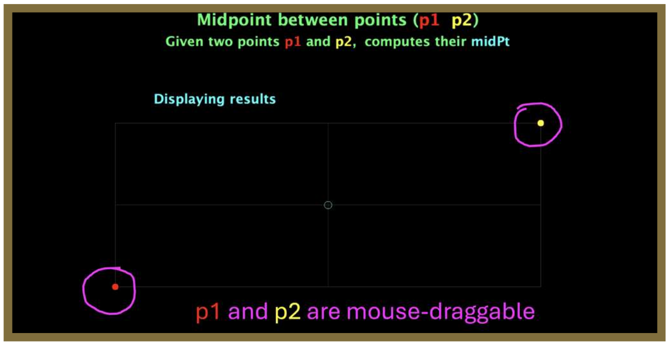
Midpoint between points (p1 p2)
Given two points p1 and p2, computes their midPt
Displaying results
p1 and p2 are mouse-draggable

Figure 1

In a second Dataflow Editor window, the application feeds the two student-manipulated test points in as numerical Givens as shown on the left in ***Figure 2***. The student constructs the midpoint calculation by linking up modules drawn from a palette of arithmetic functions.

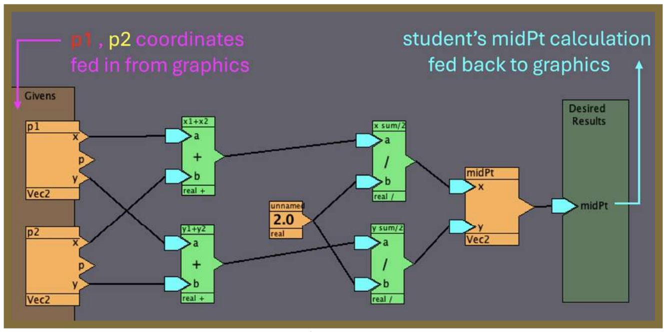

Figure 2

The student can quickly determine the correctness of the underlying sketch and dataflow implementation by judging the validity of the display location of the midpoint. By *modularizing* the solution and making it piggyback-able, we create a valuable tool to solve future problems. . Solutions piggybacking over a dozen levels equip learners to pursue advanced 2D and 3D geometry topics and projects. This develops an understanding of how spatial applications like 3D animation and GPS work. You can view a nine-minute demonstration of the Midpoint-solving process and software interaction at **youtu.be/xz-CElLwM25w**.

Other positives emerged from the Algorithmic Geometry experiment that we implemented into DataflowGeometry2D:

- ✓ The application allowed students to test their solutions for correctness using built-in computer graphics. This process developed an assessment where the instructor watches the student-developed graphics demonstration and checks off each completed lab assignment for their grade. Our students preferred and benefitted from this objective means of evaluation rather than grading their product and every student performed successfully on the final exam.
- ✓ The program provided students with the opportunity to dramatically improve their sketching skills. This ability is central to the problem-solving process since it is where the "aha" moment occurs. This sketch served as the specification for the dataflow algorithm entered into the software successfully. The Dataflow Geometry YouTube channel helps students develop a specific sketching practice.
- ✓ The course maximized the reuse of previously solved problems. Students spend the first 70 hour block working in 2D geometry, and the second 70 hours in 3D. Interestingly, they can simplify many difficult 3D problems into 2D problems, solve them using a 2D solution,

Continued on page 38☞☞

Figure 1

and then back-transform the results into 3D.

- ✔ The software simplifies the complex computation process that becomes far too complex for paper, pencil, and calculator. The software simplifies this process and offers greater problem complexity and situations that are difficult to solve with traditional techniques. This process of using programs to solve complex calculations was portrayed in the movie “Hidden Figures.” IBM mainframe/Fortran numerical programming replaced the toolset of paper and pencil and mechanical calculators to solve the problem of putting astronaut John Glenn into orbit. The women mathematician superstars on Dorothy Vaughn’s calculator team took their mathematics skills to a new power level by adopting computational thinking.
- ✔ DataflowGeometry2D, when combined with computational thinking, improved not only mathematics skills but also mathematics teaching skills. We and our students experienced an astonishing new level of problem-solving power and mental efficiency.

Our remaining work is to help you make this transition. We know from our classroom experience that students (and parents) love the modern thrust of geometry + CT, with its hands-on engagement and high relevance to 21st-century technology. College STEM professors appreciate the 3D visualization skills and advanced spatial mathematics problem-solving application.

We learned it takes a supportive administrative coalition to start and maintain progress. Teachers need support in undertaking this course. We suggest offering DataflowGeometry2D as a one-semester online Independent Study (or summer enrichment course). The idea is to gain experience and build confidence while minimizing risk.

Next Steps You Can Take

The power of this problem-solving workstyle is hard to convey until you have experienced it. We welcome your review of the courseware at **www.DataflowGeometry.org**. You will find a syllabus under the subheading **www.Dataflow Geometry.org/teachers**. The courseware (Macintosh or Windows software application+ YouTube video channel) is free for interested educators and students to study. We expect to complete DataflowGeometry3D in 2026.

Working together, we can move several steps closer to helping students better understand mathematics and the technology that impacts their lives. To quote a pragmatist-sage: “There is enlightenment at the end of the tunnel.”

References

Bierre, P. 2010. *Flexing the Power of Algorithmic Geometry*. Spatial Thoughtware.

________. 2012. “Algorithmic Geometry: 2 Year Summary Report on Dougherty Valley HS Pilot Phase.” Dougherty Valley HS. **https://algogeom.org/files/TwoYearReport.pdf**

Kagarakis, A. 2011. “Slice of Life TV 30 Piece 2011” (Tri-Valley TV). *Algorithmic Geometry Education.* **AlgoGeom.org/files/SliceOfLifeTV30Piece2011.mov**

Perez, A. 2018. “A Framework for Computational Thinking Dispositions in Mathematics Education.” *Journal for Research in Mathematics Education* 49 (4): 424–461. **https://doi.org/10.5951/jresematheduc.49.4.0424**

Resnick, M. 2009. “Programming for All (Scratch).” *MIT Media Lab 40.* **https://www.media.mit.edu/publications/programming-for-all/**

Wing, J. 2006. “Computational Thinking.” *Communications of the ACM* 49 (3, March): 33–36. **https://doi.org/10.1145/1118178.111821**

Resources

Gould, R. 2020. *Introduction to Data Science*. UCLA Center X. **https://introdatascience.org**

“Front Matter.” 2020. “Special Issue on Computational Thinking from a Disciplinary Perspective.” *Journal of Science Education and Technology* eds. Lee, I., J. Malyn-Smith, S. Grover, F. Martin, S. Pillai. 29 (1, February). **https://www.jstor.org/stable/48741975**

NCSM. 2017. “Position Paper: Computer Science and K–12 Mathematics.” NCSM. **https://www.mathedleadership.org/docs/resources/positionpapers/NCSMPositionPaper18.pdf**

Papert, S. 1980. *Mindstorms: Children, Computers, and Powerful Ideas*. Basic Books.

#CMCSouth2025

FALL INTO MATH

POST A PICTURE OF A MATH LEARNING ACTIVITY OR GAME!

As school is starting, we are all falling back into math! Comment with a picture of an activity or game that demonstrates mathematical exploration and a love f ma in Grades TK-12.

Every post is entered into a raffle for a prize! Extra roffle entry for those who TAG A COLLEAGUE!

#CMCSouth2025
#CMCFallIntoMath

#CMCFallIntoMath