CAS AND CURRICULUM: REAL IMPROVEMENT OR DÉJA VU ALL OVER AGAIN?

AL CUOCO
PAUL GOLDENBERG

1. Introduction

This paper contains the reflections of two aging pedagogues who, over the past three decades, have been using technology to help people understand mathematics. Seven years ago, we described our experiences in [2]:

> People who have been around for a while remember the early naïve optimism that surrounded the use of computers in mathematics education. Only a few years ago, computer and software companies told us that if we could just bring our classes to the lab, if we could just let our students play with any one of a hundred educational "programs," then all our troubles would go away. Students would discover mathematics for themselves, they'd "get it" by working through carefully orchestrated sequences of activities and games, we'd never have to deal with those ugly paper and pencil algorithms again, and classrooms would be happy places in which students basked in the enlightenment that came from the screen of an Apple IIe.
>
> It never happened, of course, although there were ample opportunities to learn how technology might (or might not) help students learn mathematics. . . . But with each new technological advance, people are again seduced into thinking that the golden age of technology in mathematics education is just around the corner. Witness the zeal that has surrounded the evolution of the graphing calculator and the excitement and wonder that we all show about the potential educational uses of the Internet.

And here we go again. CAS environments have been used for over a decade in undergraduate mathematics, and now, with the availability of

This paper draws on earlier work (especially [3, 6, 19, 13]). Special thanks are due to Ken Levasseur.

these media on hand-held devices, they are gradually making their way into precollege (upper secondary) programs. Especially in the United States, where jumping on bandwagons has a longstanding and quasi-respectable tradition in education, we're hearing about yet another golden age. Like all predictions of this type, the motto seems to be "if the machine can do it, why bother teaching it?" So, well-known educators are proclaiming that facility with algebraic calculation is unnecessary and that we can do away with those tortuous pages of factoring, simplifying, and solving.[1]

Experience tells us that this elated optimism will eventually evolve into something much less grandiose and that CAS environments will take their place alongside other useful computational media as enhancements to, rather than replacements for, the essential role that technical fluency plays in mathematical understanding.

In this paper, we provide a framework for designing curriculum in a way that uses CAS environments to obtain this added value. The framework is based on our uses of Mathematica, Maple, and the system on the TI-89—in our own work and with secondary students and high school teachers. It's also heavily influenced by our more mature work with Logo, Geometer's Sketchpad, and Cabri. Most of the paper is devoted to some examples of ways to use CAS environments that illustrate the features of our framework.

Two preliminary remarks are necessary before we begin:

1. When we use the term CAS, we are referring to CAS *environments* that are fully extensible languages, with all the features of a high-level programming language *and* the inclusion of formal algebraic expressions as first-class objects—general purpose packages like Mathematica, MatLab, and Maple (and, to a lesser extent, the version of Derive on the TI-89, which lacks some features of the larger packages). We don't mean special purpose packages or symbolic calculators that do not allow for user-defined functions.
2. Most CAS implementations contain sophisticated graphical routines, mathematical functions that can handle very large inputs with accuracy, special-purpose packages that, for example, can calculate eigenvalues, or values of the Riemann ζ-function or other infinite sums. We'll make little mention of these features. We believe that a major part of the educational value of CAS environments comes not from using these fancy enhancements but

[1] One of us was at a meeting of (mathematics) curriculum developers a few years ago when someone made the comment that algebra is dead, causing a roaring round of applause from the audience.

from the process of building more complex mathematical operations and structures out of more basic ones, through an extensible programming language that has formal expressions as first-class objects. Many CAS environments provide as built-in primitives some of the user-invented models that we describe in the rest of this paper. Clearly, for our purposes, these are beside the point.

2. A Framework for Curriculum Design

We see three broad (and overlapping) ways that CAS environments can help us produce more effective mathematics curriculum.

An Algebraic Calculator: CAS technology can be used to make tractable and to enhance many beautiful classical topics, historically considered too technical for high school students. This is the use of technology that reduces computational overhead and that allows students to easily perform calculations that would be impossible (or overly distracting) without the technology.

An Algebra Laboratory: CAS technology can be used to experiment with algebraic expressions in the same way that calculators can be used to experiment with numbers: generating data, making patterns apparent, and giving students the raw data from which they can generate conjectures. CAS environments provide teachers and students with general purpose tools for finding regularity in data, or for imposing regularity when no simple patterns can be found. CAS technology also has the potential to bring a renewed and modern emphasis on formal algebra—that is, the algebra of forms—to school mathematics (see [3] and [6] for more on this theme).

A Modeling Tool for Algebraic Structures: CAS technology allows students to build models of algebraic objects that have no faithful physical counterparts. This use of technology adheres to our view that building a computational model for a mathematical structure helps one build the mental constructions needed to interiorize that structure [2, 18]. Furthermore, such computational models are *executable*, so that students can build working models of mathematical systems, turning the mathematician's thought experiments into actual experiments. What CAS environments add to other modeling environments is the facility to perform generic calculations with algebraic *expressions*—polynomials, rational functions, and formal power series. Since formal algebraic expressions are the "universal" objects in algebra, CAS environments provide a medium for expressing abstract algebraic structure.

The rest of the paper gives some examples of how each of these pieces of the taxonomy play out in mathematical contexts. As with all taxonomies, no truly interesting examples illustrate exactly one feature. Instead, we have chosen several examples, each of which *highlights* a particular use of a CAS—as a calculator, laboratory, or modeling system. But each of our examples illustrates some of the other, non-highlighted features as well.

3. Some examples

In this section we give describe some "circles of ideas" that illustrate different aspects our framework. Each of these has been used, either with high school students [12], with teachers [13], or with prospective teachers [14].

3.1. Example 1: Fitting Functions to Tables.

Let's begin by using a CAS as an algebraic calculator. Many curricula ask students to find a "rule" that can be used to generate a table. For example, suppose we want to find a function f that agrees with this table:

Input	Output
0	-5
1	-6
2	7
3	166
4	843
5	2770

Lagrange Interpolation builds a polynomial (of minimal degree) that agrees with any finite set of input-output pairs.

We start with the following beast of an expression for our solution, call it f, as a sum of products:

$$\begin{aligned} f(x) = {} & Ax(x-1)(x-2)(x-3)(x-4) + Bx(x-1)(x-2)(x-3)(x-5) \\ & + Cx(x-1)(x-2)(x-4)(x-5) + Dx(x-1)(x-3)(x-4)(x-5) \\ & + Ex(x-2)(x-3)(x-4)(x-5) + F(x-1)(x-2)(x-3)(x-4)(x-5) \qquad (*) \end{aligned}$$

The numbers A–F will be determined in a minute. Each product is formed by taking

$$x(x-1)(x-2)(x-3)(x-4)(x-5)$$

and "dropping" one factor. Since each product is a polynomial of degree 5, the whole expression will be of degree at most 5.

Why write f is such a messy way? Well, it allows you to easily calculate $f(n)$ for any n between 0 and 5. For example, from the table, we want $f(0)$ to be -5. We calculate like this:

$$\begin{aligned}-5 &= A\,0(0-1)(0-2)(0-3)(0-4) + B\,0(0-1)(0-2)(0-3)(0-5)\\ &\quad + C\,0(0-1)(0-2)(0-4)(0-5) + D\,0(0-1)(0-3)(0-4)(0-5)\\ &\quad + E\,0(0-2)(0-3)(0-4)(0-5) + F\,(0-1)(0-2)(0-3)(0-4)(0-5)\end{aligned}$$

But look: all the terms except the last have a factor of 0, so they all vanish. We get

$$-5 = F(0-1)(0-2)(0-3)(0-4)(0-5) = -120F$$

So, $F = \frac{5}{120} = \frac{1}{24}$.

Next, let $x = 1$. We want $f(1)$ to be -6. But when we replace x by 1 in expression $(*)$, the only term to survive is the "E" term, and we get:

$$-6 = E(1-0)(1-2)(1-3)(1-4)(1-5) = 24E$$

So $E = -\frac{1}{4}$.

Similarly, we can pick off the other missing coefficients by replacing x by 2 (producing $D = -\frac{7}{12}$), 3 (producing $C = \frac{83}{6}$), 4 (producing $B = -\frac{843}{24}$), and 5 (producing $A = \frac{277}{12}$). Substituting in $(*)$, we get:

$$\begin{aligned}f(x) &= \frac{277}{12}x(x-1)(x-2)(x-3)(x-4) - \frac{843}{24}x(x-1)(x-2)(x-3)(x-5)\\ &\quad + \frac{83}{6}x(x-1)(x-2)(x-4)(x-5) - \frac{7}{12}x(x-1)(x-3)(x-4)(x-5)\\ &\quad - \frac{1}{4}x(x-2)(x-3)(x-4)(x-5) + \frac{1}{24}(x-1)(x-2)(x-3)(x-4)(x-5)\end{aligned}$$

A CAS simplifies this to $x^5 - 3x^3 + x^2 - 5$. Try it for yourself.

Lagrange Interpolation is more general than other methods (like Newton's difference formula [13]), because it applies to any finite set of input-output pairs. It can be stated like this:

> **Lagrange Interpolation:** An explicit way of writing the polynomial in x of minimal degree which takes the values b_i at the points a_i, $1 \le i \le r$ is

$$f(x) \;=\; b_1\frac{1}{p_1(a_1)}\,p_1(x) + b_2\frac{1}{p_2(a_2)}\,p_2(x) + b_3\frac{1}{p_3(a_3)}\,p_3(x) + \cdots + b_r\frac{1}{p_r(a_r)}\,p_r(x)$$

> where $p_i(x)$ is the polynomial defined by

$$\begin{aligned}p_i(x) &= (x-a_1)(x-a_2)\cdots(x-a_{i-1})(x-a_{i+1})\cdots(x-a_r)\\ &= \frac{\prod_{k=1}^{r}(x-a_k)}{x-a_i}\end{aligned}$$

The proof, except for the "minimal degree" part, is implicit in the example that opened this section, and the fact that f is of minimal degree follows from classical advanced high school algebra.[2] Clearly, we've used the CAS in a very basic way here—to simplify algebraic expressions. But even at this level, students have access to a new idea (a new tool) that would be difficult to use without the technology. We'll return to this example shortly.

3.2. **Example 2: Sums of Powers.** Suppose we are looking for a simple (polynomial) formula that lets you calculate the sum of the first n squares. Consider the function S_2, defined on positive integers, so that

$$S_2(n) = 0^2 + 1^2 + \dots (n-1)^2$$

We want a polynomial $s_2(x)$ so that $s_2(n) = S_2(n)$ for all positive integers n.

Because

$$\begin{aligned} S_2(n+1) - S_2(n) &= S_2(n+1) \qquad\qquad - \quad S_2(n) \\ &= \left(0^2 + 1^2 + \dots n^2\right) - \left(0^2 + 1^2 + \dots (n-1)^2\right) \\ &= n^2 \end{aligned}$$

it can be shown [13] that a cubic polynomial s_2 will agree with S_2 at positive integers. Similarly, it can be shown that there is a polynomial s_m of degree $m+1$ so that

$$s_m(n) = \sum_{k=0}^{n-1} k^m$$

for positive integers n. Indeed, the method from the last section (or other polynomial interpolation methods) can be used to find the s_m (see [13], chapter 5 for details). But our goal here is to investigate the sequence of polynomials *after* they have been generated. And to generate them, we can use a CAS.

Around 1690, Jakob Bernoulli wrote

> "With the help of [these formulas] it took me less than half of a quarter of an hour to find that the 10th powers of the first 1000 numbers being added together will yield the sum 91409924241424243424241924242500"

[2]In the example above, if g were any polynomial of degree less than f that agreed with f at the six inputs in the table, $f - g$ would be a polynomial of degree at most 5 with six roots, so it would be 0.

While this is an amazing feat to carry off by hand in $7\frac{1}{2}$ minutes, a scientific calculator can compute the sum in seconds. And *Mathematica* can compute it in less than one second:

```
In[2]:=
Sum[k^10,{k,0,1000}]

Out[2]=
91409924241424243424241924242500
```

But what makes a CAS different from a calculator is its ability to do *generic* calculations; it can compute *indefinite* sums, giving us an explicit form for s_{10}:

```
In[3]:=
Expand[Sum[k^10, {k, 0, n - 1}]]


Out[3]=
```

$$\frac{5\,n}{66} - \frac{n^3}{2} + n^5 - n^7 + \frac{5\,n^9}{6} - \frac{n^{10}}{2} + \frac{n^{11}}{11}$$

This is exactly the facility we want: we can use a CAS to generate the "data" from which we can seek out regularity, pattern, and conjecture.

Here's a table of summatory polynomials up to 10:

m	$s_m(x)$
0	x
1	$-\frac{x}{2}+\frac{x^2}{2}$
2	$\frac{x}{6}-\frac{x^2}{2}+\frac{x^3}{3}$
3	$\frac{x^2}{4}-\frac{x^3}{2}+\frac{x^4}{4}$
4	$-\frac{x}{30}+\frac{x^3}{3}-\frac{x^4}{2}+\frac{x^5}{5}$
5	$-\frac{x^2}{12}+\frac{5x^4}{12}-\frac{x^5}{2}+\frac{x^6}{6}$
6	$\frac{x}{42}-\frac{x^3}{6}+\frac{x^5}{2}-\frac{x^6}{2}+\frac{x^7}{7}$
7	$\frac{x^2}{12}-\frac{7x^4}{24}+\frac{7x^6}{12}-\frac{x^7}{2}+\frac{x^8}{8}$
8	$-\frac{x}{30}+\frac{2x^3}{9}-\frac{7x^5}{15}+\frac{2x^7}{3}-\frac{x^8}{2}+\frac{x^9}{9}$
9	$\frac{3x^2}{20}+\frac{x^4}{2}-\frac{7x^6}{10}+\frac{3x^8}{4}-\frac{x^9}{2}+\frac{x^{10}}{10}$
10	$-\frac{5x}{66}-\frac{x^3}{2}+x^5-x^7+\frac{5x^9}{6}-\frac{x^{10}}{2}+\frac{x^{11}}{11}$

Let's look at this and, following Bernoulli, see what we can see. Here are some (completely unjustified) observations, based on these 11 polynomials.

- It seems that the coefficient of x^{m+1} in $s_m(x)$ is $\frac{1}{m+1}$.
- For $m > 0$, the coefficient of x^m in $s_m(x)$ is $-\frac{1}{2}$.
- For $m > 1$, the coefficient of x^{m-1} in $s_m(x)$ is $\frac{m}{12}$.
- For $m > 2$, the coefficient of x^{m-2} is, well, 0.
- For $m > 3$, look at the x^{m-3} term:

m	coefficient of x^{m-3} in $s_m(x)$
4	$-\dfrac{1}{30}$
5	$-\dfrac{1}{12}$
6	$-\dfrac{1}{6}$
7	$-\dfrac{7}{24}$
8	$-\dfrac{7}{15}$
9	$-\dfrac{7}{10}$
10	-1

There are several methods for finding regularity in these numbers. For example, we could put them all over a common denominator (360) and look for patterns in the numerators.

Or, equipped with a CAS, we can find a polynomial of lowest degree (using Lagrange Interpolation, say) that agrees with this table. If we do that, we find an especially suggestive form: The table is matched with

$$-\frac{1}{720}\, m(m-1)(m-2)$$

We could keep going, but look at what we have so far:

$$s_m(x) = \frac{1}{m+1}\, x^{m+1} - \frac{1}{2}\, x^m + \frac{m}{12}\, x^{m-1} + 0\, x^{m-2} - \frac{1}{720}\, m(m-1)(m-2)\, x^{m-3} + \dots$$

We're guessing that, after some playing around with this, Bernoulli's keen eye noticed that you could write the coefficients this way:

$$s_m(x) = \frac{1}{m+1}\Bigg(x^{m+1} - \frac{1}{2}(m+1)\, x^m + \frac{1}{12}(m+1)m\, x^{m-1}$$

$$+\, 0(m+1)m(m-1)\, x^{m-2} - \frac{1}{720}\,(m+1)m(m-1)(m-2)\, x^{m-3} + \dots \Bigg)$$

$$= \frac{1}{m+1}\left(\binom{m+1}{0} x^{m+1} - \frac{1}{2}\binom{m+1}{1} x^m + \frac{1}{6}\binom{m+1}{2} x^{m-1}\right.$$

$$\left. +\, 0\binom{m+1}{3} x^{m-2} - \frac{1}{30}\binom{m+1}{4} x^{m-3} + \dots \right)$$

And a conjectured pattern emerges: Perhaps the coefficient of x^{m+1-k} in $s_m(x)$ is a constant multiple of

$$\frac{1}{m+1}\binom{m+1}{k}$$

Well, it's true. And the "constant multiples" in the conjecture have become known as *Bernoulli numbers*, denoted by B_k. We've calculated a few just now, and here are a few more:

k	0	1	2	3	4	5	6	7	8	9	10
B_k	0	$-\frac{1}{2}$	$\frac{1}{6}$	0	$-\frac{1}{30}$	0	$\frac{1}{42}$	0	$-\frac{1}{30}$	0	$\frac{5}{66}$

This is just the beginning. This particular topic takes one into some beautiful mathematical investigations (see [13], for example), but the more general point is that a CAS allows one to play with algebraic forms in the same way a scientist experiments with physical phenomena—the environment is a tool for generating algebraic data, for transforming the data, for finding or imposing regularity in the data, and for forming and testing conjectures.

3.3. **Example 3: Formal Algebra.** Over the past 20 years, precollege algebra in the US has followed what has come to be known as the "functions approach." Essentially, polynomials are thought of as functions and where the "letters" are *variables*, and a great deal of school algebra is occupied with learning to use polynomial functions to model situations and with learning to use Cartesian graphs as the main representational tool for these functions.

There's another approach to elementary algebra in which in which polynomials are *forms* and the letters are considered *indeterminates*. This point of view treats, for example, the set of polynomials in one variable with rational coefficients as a system in its own right, with its own internal logic, structure, and means for carrying out calculations (more on this in the next section and in [3, 15]). This kind of algebra fell out of fashion in the 1990s, but it is exactly the point of view supported by algebraic calculations in CAS environments. The algebraic calculator in a CAS deals only with the *forms* of expressions. The interpretation of the forms is up to the user.

When one doesn't concentrate on "the meaning of x," algebraic identities are established, not by showing that two functions are the same or that two graphs coincide, but by showing that both sides of the identity can be reduced to the same normal form via the "rules of algebra." Calculations carried out in this way have several uses in modern mathematics. We'll look at one example from "abstract" algebra in the next section. In this section, we look at an example from combinatorics.

Many elementary and middle school curricula ask students to experiment with probability. One of the most popular experiments has to do with throwing dice. For example, an activity to accompany a grade 5 curriculum asks students the following kinds of questions.

1. When tossing a single die, how many different numbers can show up? What is the most likely number?
2. When tossing two dice, how many different pairs can show up? How many different sums can show up? What is the most likely sum?

Many students realize that the possible sums for two dice (problem 2 above) are $2, 3, \ldots, 12$. It's common to see them make a table showing what can happen on each die:

Die 1→ Die 2↓	1	2	3	4	5	6
1	2	3	4	5	6	7
2	3	4	5	6	7	8
3	4	5	6	7	8	9
4	5	6	7	8	9	10
5	6	7	8	9	10	11
6	7	8	9	10	11	12

This not only tells you that 7 is the most most likely sum, it also tells you how the other sums are *distributed*:

Sum	2	3	4	5	6	7	8	9	10	11	12
Frequency	1	2	3	4	5	6	5	4	3	2	1

Teachers often ask extension questions:

1. When tossing three dice, how many different triples can show up? How many different sums can show up? What is the most likely sum?
2. When tossing n dice, how many different n-tuples can show up? How many different sums can show up? What is the most likely sum?

Formal algebra is a useful tool in analyzing the distribution of sums on n dice. Let's look at the case $n = 2$ from this point of view. Suppose we take the distribution for two dice:

Sum	2	3	4	5	6	7	8	9	10	11	12
Frequency	1	2	3	4	5	6	5	4	3	2	1

and hang it out on a clothesline[3] of xs:

$$x^2 + 2x^3 + 3x^4 + 4x^5 + 5x^6 + 6x^7 + 5x^8 + 4x^9 + 3x^{10} + 2x^{11} + x^{12}$$

The general term here is

$$\{\text{number of ways you can get } s \text{ as a sum}\} \times x^s$$

Why would you do such a thing? Well, it's just another code, another representation, of the information in the above table. It may even be easier to write down. It certainly takes less space. And you can "do algebra" with this expression. For example, we can (by hand or CAS)

[3]This image of a clothesline is due to Herbert Wilf [9].

factor it to discover that

$$\begin{aligned}x^2 + 2x^3 + 3x^4 + 4x^5 + 5x^6 + 6x^7 + 5x^8 + 4x^9 + 3x^{10} + 2x^{11} + x^{12} &=\\ x^2(x+1)^2(x^2+x+1)^2(x^2-x+1)^2 &=\\ \left[x(x+1)(x^2+x+1)(x^2-x+1)\right]^2 &=\\ (x+x^2+x^3+x^4+x^5+x^6)^2\end{aligned}$$

Hence the "distribution polynomial" for two dice is the square of the distribution polynomial for one die. This fact may seem surprising at first. We've had good luck helping people see thorough the surprise with prompts like these:

1. Explain, as if you were showing a student, how you'd multiply out
$$\left(x+x^2+x^3+x^4+x^5+x^6\right)^2$$
   (a) How, for example, would you get the coefficient of x^6 in the product?
   (b) How would you get the coefficient of x^7 in the product?
   (c) How would you get the coefficient of x^k in the product? Give your answer as a set of instructions.
2. Explain "why" the expansion of
$$\left(x+x^2+x^3+x^4+x^5+x^6\right)^2$$
   should give you the distribution for the sums that show up when rolling two dice.
3. Use your CAS to get the distribution polynomial for the possible sums when you throw three dice. Explain how you know it "works."
4. How could you use your CAS to get distribution polynomial for the possible sums when you throw k dice. Explain how you know this polynomial "works."
5. Use the result of problem 4 (or anything else you'd like to use about the algebra of polynomials) to get formulas for:
   (a) the number of possible "events" you can get when you throw k dice.
   (b) the most likely sum when you throw k dice.

The insight emerges that the distribution of sums when k dice are thrown can be read off from the coefficients of

$$\left(x+x^2+x^3+x^4+x^5+x^6\right)^k$$

When $k = 3$, a CAS produces

$$x^3 + 3\,x^4 + 6\,x^5 + 10\,x^6 + 15\,x^7 + 21\,x^8 + 25\,x^9 + 27\,x^{10} + 27\,x^{11} \\ +25\,x^{12} + 21\,x^{13} + 15\,x^{14} + 10\,x^{15} + 6\,x^{16} + 3\,x^{17} + x^{18}$$

Like all good representations, this distribution polynomial (or, as it's more commonly known, *generating function*) supports contextually meaningful interpretations of transformations on the representation itself. In this case, the transformations on the representations are algebraic calculations—substituting, factoring, and rearranging to uncover "hidden meaning" [3].

For example, replacing x by 1 gives the sum of the coefficients which can be interpreted as the number of possible events. And writing the polynomial above as

$$\left(x + x^2 + x^3 + x^4 + x^5 + x^6\right)^3$$

we see that the sum of the coefficients (an *invariant* of the particular form taken on by a polynomial) is 6^3. More generally, we find that the number of events possible when k dice are thrown is 6^k.

As another example, notice that the number of even sums is the same as the number of odd sums in the examples we have looked at so far. Will this always be the case? In terms of the generating functions, we wonder if the number of even coefficients is the same as the number of odd coefficients. One way to check is to realize that the excess in parity of even over odd (or *vice-versa*) is the value of the expression you get when x is replaced with -1. And this value is again invariant under change of form. Again, writing the generating function as

$$\left(x + x^2 + x^3 + x^4 + x^5 + x^6\right)^k$$

we see that the value when $x = -1$ is always 0, so there are as many odd coefficients as even ones, and there are as many odd dice-sums as even ones. One more example: Notice that the coefficients in our generating functions (for $k = 2$ and 3, at least) are "symmetric from the middle." In fact, they seem to be triangular numbers for awhile, but then they get "clipped."[4] A classical result in algebra shows that a polynomial has this property if and only if its roots come in reciprocal pairs. More precisely, the result states that, a polynomial whose zeros are the reciprocals of the zeros of the polynomial

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots a_1 x + a_0$$

[4]Ken Levasseur has built a lovely visual explanation of this fact in *Mathematica* that can be downloaded from `www2.edc.org/cme/showcase/`.

is

$$a_0x^n + a_1x^{n-1} + \dots a_{n-1}x + a_n$$

provided that $a_0 \neq 0$.[5] A polynomial is thus invariant under this "coefficient reversal" if and only of its zeros come in reciprocal pairs. Our generating function for k dice can be written as

$$x^k(1 + x + x^2 + x^3 + x^4 + x^5)^k$$

Now $1+x+x^2+x^3+x^4+x^5$ clearly has the symmetry property in its its coefficients, so its zeros come in reciprocal pairs (in fact, they are the non-trivial sixth roots of unity). Hence the zeros of

$$(1 + x + x^2 + x^3 + x^4 + x^5)^k$$

come in reciprocal pairs (they just form k copies of the zeros of $1 + x + x^2 + x^3 + x^4 + x^5$). Hence this kth power has the symmetry property in *its* coefficients. And multiplying it by x^k will not change this fact.

Of course there are "dice-theoretic" arguments for each of these results, but our point here is that this generating function representation allows one to map algebraic results onto combinatorial insights. And the CAS plays an important role behind the scenes here. It allows students to experiment with algebraic calculations, to make substitutions, to solve equations, and to gather evidence for conjectures, without getting detoured by the computational overhead.

There's also a shift in emphasis in the approach to "symbol manipulation." Clearly, there's less of a focus on the ability of students to accurately and quickly expand products and simplify terms in "by hand" algebraic calculations. Instead, the focus is on questions like "If you performed the following algebraic calculation, what would you get as the coefficient of x^6?" Clearly the two skills—fluency in performing algebraic calculation and fluency in *imagining* algebraic calculations—are related, but the precise nature of this relationship is not clear. What *is* clear is that students can develop fluency and skill in performing algebraic calculations *without* the abilities to reason about calculations or to do algebraic thought experiments.

Furthermore, the facility of the CAS to deal with *infinite* series supports this formal algebraic reasoning in very general combinatorial settings. Here is one concluding example of a problem that can be approached this way:

[5]The second polynomial can be written as $g(x) = x^n f\left(\frac{1}{x}\right)$. It follows that the zeros of g are the reciprocals of the zeros of f).

In US currency, there are coins worth 1, 5, 10, 25, and 50 cents. The number of ways to make change for \$$n$ is the coefficient of x^n in

$$\begin{aligned}(1+x+x^2+\dots)(1+x^5+x^{10}+\dots)(1+x^{10}+x^{20}\dots)\\(1+x^{25}+x^{50}+\dots)(1+x^{50}+x^{100}+\dots) =\end{aligned}$$

$$\frac{1}{1-x}\cdot\frac{1}{1-x^5}\cdot\frac{1}{1-x^{10}}\cdot\frac{1}{1-x^{25}}\cdot\frac{1}{1-x^{50}} =$$

$$\frac{1}{(1-x)(1-x^5)(1-x^{10})(1-x^{25})(1-x^{50})}$$

and a CAS can be used to experiment with this expression.

3.4. **Example 4: Structural Similarities.** One of the most useful algebraic habits of mind is to look out for and pay attention to calculations that "feel the same." For example, consider Lagrange interpolation again. In general, we're looking for a polynomial f so that

$$\begin{aligned}f(a_1) &= b_1\\ f(a_2) &= b_2\\ &\vdots\\ f(a_r) &= b_r\end{aligned}$$

By the remainder theorem,[6] another way to say this is that we want f to satisfy

$$\begin{aligned}f(x) &= (x-a_1)q_1(x)+b_1\\ f(x) &= (x-a_2)q_2(x)+b_2\\ &\vdots\\ f(x) &= (x-a_r)q_r(x)+b_r\end{aligned}$$

[6]**The remainder theorem**: If f is a polynomial, then the remainder when $f(x)$ is divided by $x-a$ is $f(a)$. Divide $f(x)$ by $x-a$ to get a quotient q and a remainder r. Since the degree of $x-a$ is 1, the degree of r is 0; that is, r is a *number*, and $f(x)=(x-a)q(x)+r$. Replace x by a. You get $f(a)=(a-a)q(a)+r=r$

where q_i is the quotient you get when $f(x)$ is divided by $(x - a_i)$. Put another way, we want f to satisfy the *congruences:*

$$\begin{aligned} f(x) &\equiv b_1 \pmod{(x-a_1)} \\ f(x) &\equiv b_2 \pmod{(x-a_2)} \\ &\vdots \quad \vdots \\ f(x) &\equiv b_r \pmod{(x-a_r)} \end{aligned}$$

In other words, we are looking for a polynomial that simultaneously leaves r prescribed remainders modulo r different polynomials of the form $x-a$. This looks very much like a classical problem in arithmetic, where we look for an integer that simultaneously leaves r prescribed remainders modulo r different relatively prime integers.[7] *That* problem is solved by a famous and ancient result:

> **The Chinese Remainder Theorem:** Suppose $m_1, m_2, \ldots, m_r$ are relatively prime integers. An explicit way of writing a solution to the set of simultaneous congruences
>
> $$\begin{aligned} y &\equiv b_1 \pmod{m_1} \\ y &\equiv b_2 \pmod{m_2} \\ y &\equiv b_3 \pmod{m_3} \\ &\vdots \quad\quad \vdots \\ y &\equiv b_r \pmod{m_r} \end{aligned}$$
>
> is
>
> $$y = b_1(n_1)^{-1}n_1 + b_2(n_2)^{-1}n_2 + b_3(n_3)^{-1}n_3 + \cdots + b_r(n_r)^{-1}n_r$$
>
> where
>
> $$\begin{aligned} n_i &= n_1 n_2 \ldots n_{i-1} n_{i+1} \ldots n_r \\ &= \frac{\prod_{k=1}^{r} n_k}{n_i} \end{aligned}$$
>
> and $(n_i)^{-1}$ is the (reduced) multiplicative inverse of n_i modulo m_i. Furthermore, every solution to this system is congruent to this one modulo $\prod_{k=0}^{r} m_k$.

We've left out several details (how you know that n_i has an inverse modulo m_i, for example), because we want to stress the analogy with Lagrange Interpolation. You get the sense that these theorems "feel" the same. Compare the statement of the theorem with that of Lagrange

[7] "The reminder when my age is divided by 3 is 2. When I divide it by 5, the remainder is 3. When I divide it by 7, the remainder is 1. How old am I?

interpolation on page 5. They both involve a construction of a linear combination of products of factors, each factor leaving one term out. The $x - a_i$ are irreducible, so they are certainly relatively prime. And it turns out that the $\frac{1}{p_i(a_i)}$ in the statement of Lagrange Interpolation are just the multiplicative inverses of the $p_i(x)$ modulo $x - a_i$.

This sameness goes deeper than the fact that both theorems have similar statements—it comes from basic structural similarities between the ordinary integers and the system of polynomials in one variable with, say, rational coefficients. Both of these systems are *Euclidean rings*: they have a "long division" algorithm, and you can use Eulcid's algorithm to calculate the gcd of two elements (integers or polynomials). Students practice the features of this structural similarity throughout their K–12 experience. In elementary school, they factor integers into primes, divide with remainder, and find gcds and lcms. They do similar things in high school algebra, except they do them with polynomials instead of with integers.

A CAS can be used to make this structural similarity explicit. One can write simple CAS routines that implement the Chinese Remainder Theorem's solution to simultaneous congruences, providing a computational "proof" of the Chinese remainder theorem. Then, by changing a few basic functions—that compute quotient and remainder, for example—so that they work with polynomials instead of integers, the *same* routines will implement Lagrange interpolation. Details are in [19], and the complete Mathematica code is at `www2.edc.org/cme/showcase/`.[8] To get a flavor the role of the CAS, here's a brief sketch of the highlights.

Suppose we want to build a model of $\mathbb{Z}/m\mathbb{Z}$, the ring of integers modulo m. Following the modeling method outlined in [5], we build some basic functions that construct computational representations of the objects in our system (congruence classes modulo m) and that select pieces of that representation (a particular representative of a congruence class, say).

A congruence class modulo m is represented by a pair $\{a, m\}$ where a is the "distinguished" representative of the class (the integer between 0 and $m - 1$ that represents the class). In our model, `class` is the basic constructor, and the selectors `rep` and `modulus`, when applied to a class, produce the distinguished representative and the modulus, respectively.

In [1]:=
```
modulus [ class [15,6] ]
```

[8] Or, it *will* be by the time of the meeting.

Out [1]=
```
6
```
In [2]:=
```
rep [ class [15,6] ]
```
Out [2]=
```
3
```
In [3]:=
```
class [15,6] == class [21,6]
```
Out [3]=
```
True
```

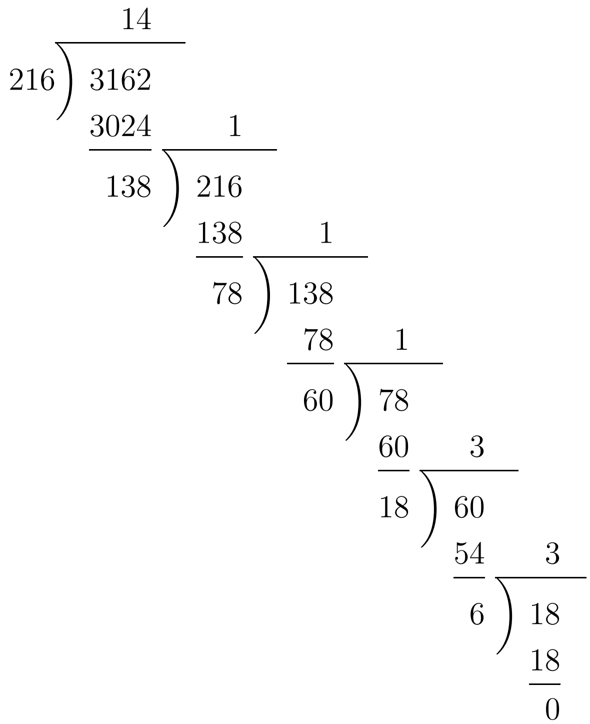
Two classes are added and multiplied by performing the corresponding operations in $\mathbb{Z}$ with the representatives of the classes and then forming the class of the results:

```
add[c_,d_] :=
   class[ rep [c] + rep [d],modulus[c]]

mult[c_,d_] :=
  class [ rep [c] * rep [d],modulus[c]]
```

A class has a multiplicative inverse if its representative is relatively prime to its modulus. To see this, one uses the fact that in $\mathbb{Z}$, one can compute the greatest common divisor of two integers via Euclid's algorithm, modeled here with two lines of *Mathematica* code.

```
gcd[a_,b_] := gcd[mod[b,a],a]

gcd[0,b_] := b
```

The algorithm is carried out below for 216 and 3162:



So, $\texttt{gcd}(216, 3162) = 6$.

A careful analysis of this algorithm (through several calculations like this) shows that the gcd of two integers a and b can always be written as a linear combination of a and b with integer coefficients. In fact, one can abstract off the regularity from this analysis of specific calculations to produce computational models of the the coefficients in the linear combination. That is, one can model functions `fcoeff` and `scoeff` such that

$$\texttt{fcoeff}(a, b) \cdot a + \texttt{scoeff}(a, b) \cdot b = \gcd(a, b)$$

Mathematica models for `fcoeff` and `scoeff` look like this:

```
fcoeff[0,b_] := 0

fcoeff[a_,b_] := scoeff[ mod[b,a],a ] - quot[b,a]* fcoeff[ mod[b,a],a ]



scoeff[0,b_] = 1

scoeff[a_,b_] := fcoeff[ mod[b,a],a ]
```

The details of the analysis that leads to these models are in [19].

Now, if $\gcd(a, m) = 1$, we have

$$\texttt{fcoeff}(a, m) \cdot a + \texttt{scoeff}(a, m) \cdot m = 1$$

Reducing this equation modulo m, we find

$$\texttt{fcoeff}(a, m) \cdot a \equiv 1 \pmod{m}$$

Hence a has a multiplicative inverse modulo m (namely $\texttt{fcoeff}(a, m)$). This leads us to a *Mathematica* model for the reciprocal of a class:

```
recip[c_] :=  class[ fcoeff[ rep[c], modulus[c] ], modulus[c] ]
```

A *Mathematica* model for the Chinese remainder theorem takes two inputs, lists of the same length (say, r):

- a list ℓ of "targets" b_i, and
- a list m of (relatively prime) moduli m_i

To find an integer y that is congruent to b_i modulo m_i, we calculate

$$\texttt{CRT}(\ell, m) = \sum_{k=0}^{r} b_k \cdot (n_k)^{-1} \cdot n_k \pmod{\prod_{k=0}^{r} m_k}$$

where n_k is the "partial product"

$$n_k = \frac{\prod_{j=0}^{r} m_j}{m_k}$$

and $(n_k)^{-1}$ is the representative of the reciprocal of the class of n_k in $\mathbb{Z}/m_k\mathbb{Z}$.

In Mathematica, this becomes

```
CRT[l_,m_] := mod[Apply[Plus,l*inv[pp[m],m]*pp[m]],Apply[Times,m]]
```

Here, pp computes the list of partial products for a list of moduli:

```
pp[m_] := Table[Apply[Times, m ]/m[[i]],{i,1,Length[m]}]
```

And inv maps the reciprocal function over the cross product of ℓ and m and then forms the list of distinguished representatives of the reciprocals (these are the $(n_k)^{-1}$):

```
inv[l_,m_] := Map[rep,Map[recip,cp[l,m]]]
```

The complete and documented *Mathematica* code is at `www2.edc.org/cme/showcase/`. To find an integer that satisfies

$$\begin{aligned} y &\equiv 1 \pmod{3} \\ y &\equiv 0 \pmod{2} \\ y &\equiv 3 \pmod{5} \\ y &\equiv 4 \pmod{77} \\ y &\equiv 5 \pmod{13} \end{aligned}$$

we type

```
 In[29]:=
CRT[{1,2,3,4,5},{3,2,5,77,13}]

 Out[29]=
928
```

One can check that 928 has the desired properties.

Everything in this section so far could be carried out in any good programing language (see [7] for Logo models). But the value added is that *the same models work without modification for polynomials in one variable.*

Instead of the integers $\mathbb{Z}$, consider the ring of polynomials in one variable with rational coefficients, $\mathbb{Q}[x]$. As so many high school students can testify, one can perform long division with two polynomials, obtaining a quotient and a remainder (whose degree is smaller than that of the divisor). This allows one to calculate via Euclid's algorithm the gcd of two polynomials, and everything else follows. All that's necessary is an adjustment to the functions `mod` and `quot`, so that they compute quotients and remainders in $\mathbb{Q}[x]$ rather than in $\mathbb{Z}$. Details are in [19]. With these adjustments, suppose, as on page 9, we want to fit a function to this table:

4	$-\frac{1}{30}$
5	$-\frac{1}{12}$
6	$-\frac{1}{6}$
7	$-\frac{7}{24}$
8	$-\frac{7}{15}$
9	$-\frac{7}{10}$
10	-1

We are looking for a polynomial f so that

$$f(x) \equiv -\frac{1}{30} \pmod{(x-4)}$$
$$f(x) \equiv -\frac{1}{12} \pmod{(x-5)}$$
$$f(x) \equiv -\frac{1}{6} \pmod{(x-6)}$$
$$f(x) \equiv -\frac{7}{24} \pmod{(x-7)}$$
$$f(x) \equiv -\frac{7}{15} \pmod{(x-8)}$$
$$f(x) \equiv -\frac{7}{10} \pmod{(x-9)}$$
$$f(x) \equiv -1 \pmod{(x-10)}$$

In *Mathematica*, we type

```
CRT[{-1/30,-1/12,-1/6,-7/24,-7/15,-7/10,-1},{x-4,x-5,x-6,x-7,x-8,x-9,x-10}]
```

and the CAS returns

$$-\frac{x}{360}+\frac{x^2}{240}-\frac{x^3}{720}$$

In this example, we used the CAS as a medium for expressing the "sameness" of the Chinese remainder theorem and Lagrange interpolation. But this modeling process is much more general. For example, every finite number system can be built up from our model of $\mathbb{Z}/n\mathbb{Z}$. And every algebraic extension of $\mathbb{Q}$ can be realized as a modular system obtained from $\mathbb{Q}[x]$ by reducing modulo some polynomial. It will take a fair amount of work to make this approach tractable to beginning students, but we see a great potential in it for introducing students to major themes in abstract algebra.

3.5. **Conclusion.** We're not advocating that the mathematical topics we've discussed in this paper —Lagrange interpolation, summatory polynomials, generating functions, and structural similarities between integers and polynomials—should be integrated into the K–16 mathematics curriculum, although we have used precisely these topics with some success in our teaching. What's important to us is that the discussion about the ways in which CAS environments can be used moves past the extremist "all or none" debates that we often hear in US education circles and moves towards a program of experiments and research to better understand the potential of these computational media for helping students think mathematically. Our examples are meant only to illustrate some possibilities of CAS use that supplement, rather than supplant, traditional algebra curricula.

Other work along these lines is underway. The NCTM recently published a book of essays about CAS use in education, with contributors from over 10 countries [1]. Several curricular experiments at the high school level are ongoing [4, 8, 11, 12, 14, 16]. And the different factions of mathematics education—education researchers, teachers, mathematicians, and others—are making progress, at least in public reports and pronouncements, on integrating the perspectives that exist in the mathematics community [10], so that it's more likely now than in the past that suggestions for curricular change take multiple points of view into account.

We are currently developing a high school program that will make essential use of CAS environments [12], and we're using the framework outlined in section 2 as we develop the blueprint for this program. We'll close this paper with some of the questions with which we are grappling in this endeavor:

Performing calculations versus imagining them: What are the precise connections between between technical fluency in performing calculations and skill in reasoning about them? How much (and what kind of) "hand" calculation is needed in order for students to "see" algebraic identities like

$$(1 + x + x^2 + \cdots + x^6)(x - 1) = x^7 - 1$$

Function $\leftrightarrow$ form: How can students come to flexibly use the two "ways to think about" algebraic expressions—as algebraic functions and as algebraic forms—without getting bogged down in distinctions (variable versus indeterminate, for example) that are best left for more experienced users of algebra? What new methods of investigation and proof are available when one is used to moving back and forth between function and form?

Finding, imposing, expressing regularity: "Looking for patterns" has almost become a mantra in mathematics education. How can students learn to spot regularity in algebraic calculations that will allow them to develop general theories? What tools can be integrated into precollege curricula that will allow students to *impose* regularity on "stubborn" data? How can CAS-rich curricula help students find and express regularity among algebraic *systems* (as opposed to regularity in sequences of numbers or expressions) in ways that preview important ideas from abstract algebra?

References

[1] Fey, J. et.al. (eds) *Computer Algebra Systems* NCTM:Reston, 2003.

[2] Cuoco, A. and P.Goldenberg. "A Role for Technology in Mathematics Education." *Journal of Education*, 178:2, 1996.

[3] Goldenberg, E.P."Algebra and Computer Algebra." In *Computer Algebra Systems* (J. Fey et al, eds) NCTM:Reston, 2003.

[4] Zehavi, Z. and Mann, G. "Task Design in a CAS Environment: Introducing (In)Equations." In *Computer Algebra Systems* (J. Fey et al, eds) NCTM:Reston, 2003.

[5] Abelson, H., and Sussman, G,J. *Structure and Interpretation of Computer Programs*, MIT Press: Cmbridge, 1985.

[6] Cuoco, A. and Levasseur, K. "Classical Mathematics in the Age of CAS." In *Computer Algebra Systems* (J. Fey et al, eds) NCTM:Reston, 2003

[7] Cuoco, A. *Investigations in Algebra*. MIT press: Cambridge, 1990.

[8] Hibbard, A. and K. Levasseur, *Exploring Abstract Algebra with Mathematica*, Springer-Telos: New York, 1999.

[9] Wilf, H. *generatingfunctionology*. Academic Press. New York, 1994.

[10] Cuoco, A. "L'insegnamento della Matematica negli Stati Uniti." *Bollettino UMI, La Matematica nella Societ e nella Cultura*, Serie VIII, Vol. VI-A, Aprile 2003.

[11] Fey, J. et. al. *Concepts in Algebra.* Everyday Learning:Chicago, 1995.

[12] EDC, *Mathematical Themes*, a 4-year high school curriculum being developed at EDC with funding from NSF.

[13] EDC, *The Mathematical Companion*, In preparation.

[14] Greenes, C.. Cuoco, A., *et.al. Making the Connections*. A course for prospective teachers. Preprint available at
`http://www2.edc.org/connect/`

[15] Cuoco, A. "Thoughts on Reading Artigue's 'Learning Mathematics in a CAS Environment'." *International Journal of Computers for Mathematics Learning*, 7:3, 2002.

[16] Cuoco, A. and M. Manes, "When Memory Fails: Putting Limitations to Good Use," *The Mathematics Teacher*, 94(6).

[17] Boole, G., *A Treatise on the Calculus of Finite Differences*, Macmillan: London, 1860.

[18] Harel, I. and S. Papert *Constructionism* 1991 Ablex Publishing Corporation Norwood, NJ

[19] Cuoco, A. "Constructing the Complex Numbers." *International Journal of Computers for Mathematics Learning*, 2:(155–186), 1997.

Center for Mathematics Education, EDC, 55 Chapel Street, Newton MA 02458
E-mail address: `alcuoco@@edc.org, paulg@@edc.org`