Deriving Baskets

James Mallos, Sculptor, Washington DC, USA; jbmallos@gmail.com

Abstract

A hands-on activity that introduces at an elementary level a theme of 20-century mathematics—the exploration of algebraic structures less regular than a group—is presented. Students play with an algebraic structure that generates codes in Mullin's encoding of the plane graphs, and thus generates sequences of craft actions that produce a genus zero (handle-less) basket while working in a closed loop. Abstract algebra is an advanced topic, but this hands-on activity requires only facility in making algebraic substitutions. Starting from the algebraic structure's identity element, 1, the young algebraist derives an original basket by making a sequence of permitted algebraic substitutions, *ad libitum.* Upon declaring her derivation complete, she fabricates her basket out of folded paper strips via an easy unit weaving technique—a task made easier by the fact that the variables' letter shapes are mnemonic for the craft actions they represent. Play in this microworld reveals that all such baskets are related by a connected network of algebraic substitutions: in mathematical terms, they constitute the equivalence class of 1.

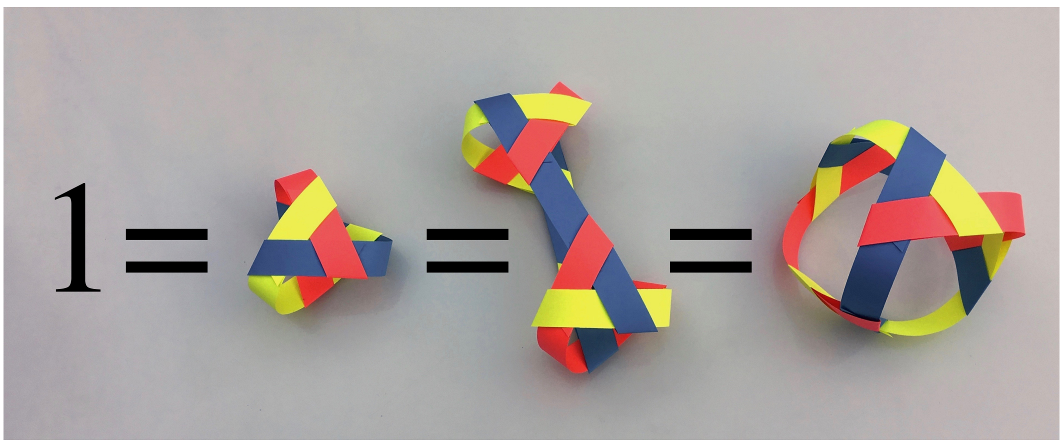

Figure 1: *By deriving expressions equivalent to 1, a young algebraist generates valid sequences of craft actions that yield a basket. Upon completing an original derivation, the sequence of letter shapes is interpreted to make the corresponding basket.*

Introduction

"We begin with the hypothesis that any subject can be taught effectively in some intellectually honest form to any child at any stage of development." [2] This bold conjecture by psychologist Jerome Bruner, a central figure in the New Math movement of the 1960's, is no doubt still controversial in math education. Yet, what Bruner describes is typically the way a young person learns engineering: by noticing and playing with one of the ubiquitous artifacts of contemporary technology, and becoming greedy for the power to alter it, improve it, or just curious enough to take it apart. This paper proposes an activity (Figure 1) that allows middle school students to play with some mathematical ideas dating from the early to mid 20th century.

Background

As a sculptor, I have been interested shapes that are 'primal' in the sense that they can be described with very little information. This led me to baskets, and the codewords (symbolizing sequences of craft actions) that describe their construction. The codewords do not actually describe shapes, but, given a basket's tendency to minimize its bending energy, a particular shape generally does spring into existence when the construction is completed. I have found that children can learn to play a word game to generate a codeword, and then follow its instructions in making a basket. Some fun can be had in this way [5].

In the past, I taught codeword generation via rewriting rules that even quite young children could learn, but the rules seemed arbitrary and 'spun out of whole cloth.' It is exciting to now realize that the laws of algebra can enforce the same rewritings, so this sort of basket making now connects in a (hopefully) inspiring way with knowledge somewhat older kids are gaining in math class.

The Algebraic Structure

The introductory study of groups usually begins with *finite* groups. These offer the advantage of being represented by a Cayley table. Also, suitable examples of finite groups can be chosen to connect with our common experience manipulating symmetrical objects in 3-dimensional space. That said, students also bring to the classroom their common experience manipulating character strings, whose concatenation, as 20-century mathematicians discovered, is tantamount to playing with infinite monoids—algebraic structures a step more general than infinite groups. Monoids generalize groups by not requiring the existence of inverses.

Given a finite set, A, called the alphabet, whose elements are called characters, the infinite set containing all character strings that can be formed from A by repeated concatenation, is designated $A*$, termed the free monoid over A. [4] That $A*$ is closed under the operation of concatenation follows by definition. No multiplication table is required here: when two elements of $A*$ are to be concatenated, they need only be written next to each other. That the operation of string concatenation is associative is likewise almost self-evident: no arrangement of parentheses alters the final order of the characters. It is less obvious that $A*$ contains an identity element, a string that, when concatenated to a second string (no matter as prefix or postfix,) returns the second string. The identity element is supplied by convention: the empty string, designated 1, the unique character string of zero-length, is asserted to be to be an element of $A*$.

Monoids become 'less free' as we add equations asserting that certain strings are equivalent to certain other strings. That observation is most obviously true when we find the product of two elements on one side of an equation and the identity element on the other—at which point we are near to discovering a commutative pair of inverses in the structure, which would bring us one step closer to a group.

In our algebraic structure for generating basket instructions we take the alphabet, A, to be:

$$A = \{\mathrm{a, b, c, d}\}.$$

The following small set of relations on $A*$ completes the specification:

$$\mathrm{ab} = 1, \quad \mathrm{cd} = 1; \tag{1}$$

$$\mathrm{ac} = \mathrm{ca}, \quad \mathrm{ad} = \mathrm{da}, \quad \mathrm{bc} = \mathrm{cb}, \quad \mathrm{bd} = \mathrm{db}. \tag{2}$$

In words, the relations (1) show that the structure has certain *one-sided inverses* which do not commute (in contrast, in a group every element commutes with its *two-sided inverse:* $xx^{-1} = 1 = x^{-1}x$); the relations (2) show that the structure has certain other pairs of elements that do commute. In effect, when combined with associativity, the relations say:

ab and **cd** are names for **1**,

a's and **b**'s can shuffle past **c**'s and **d**'s, and vice versa.

Deriving a Basket

Here is a derivation of a small basket:

$$1 \tag{3}$$

$$= \text{ab} \tag{4}$$

$$= \text{abcd} \tag{5}$$

$$= \text{acbd} \tag{6}$$

Each line in the derivation encodes a basket: 1 encodes the *null basket*, which may be understood as the artisan working around a closed loop without performing any actions (the null basket can only be seen with the mind's eye.) In the second line, the algebraist invokes an inverse pair as an equivalent of 1. In the third line, the she invokes the other inverse pair as an equivalent of 1, inserting it on the right. In the last line, she commutes a commutative pair. This derivation could be extended indefinitely. For a first experience in making a basket, you should probably declare that your derivation has demonstrated *quod erat demonstrandum* before the expression exceeds six letters.

```
   ab
    ^
  acdb
     x
  acbd
  ^
 cdacbd
   x
 cadcbd
    ^
cadabcbd
     x
cadacbbd
   x
caadcbbd
```

Figure 2: *This derivation starts with a primal basket (3-gonal hosohedron) and passes through tetrahedral and triangular-prism stages to end as a cubic basket (with some more organically-shaped stages passed on the way.) The interlinear markup makes the derivation easier to follow.*

Derivations are easier to follow if each step is hinted by markup in the space between lines: a caret for insertions, an 'x' for shuffles (Figure 2).

Making the Basket

Fabric making techniques such as crochet and net making can be worked from the same letter sequences; we will be content to make a paper basket by weaving together paper units [6]. To make a 6 letter basket only requires three 3-ply decks of weaving units, and at least nine can be cut from a 3-ply deck of letter-sized sheets, so the material needed amounts to only one sheet of colored paper per student. (The commercial Flexeez (a.k.a. Wammy) snap-together construction pieces offer an alternative modeling technique that skips over the step of learning to weave the paper units.)

Manufacturing the Weaving Units

Figure 3 shows a way to make the paper units using a 30-60-90 degree triangle (homemade from corrugated cardboard) and a paper trimmer. Three colors of paper are needed. If you have a paper trimmer, it will be convenient to fold and cut a multicolored deck of three sheets of paper at a time. The first task is to make the

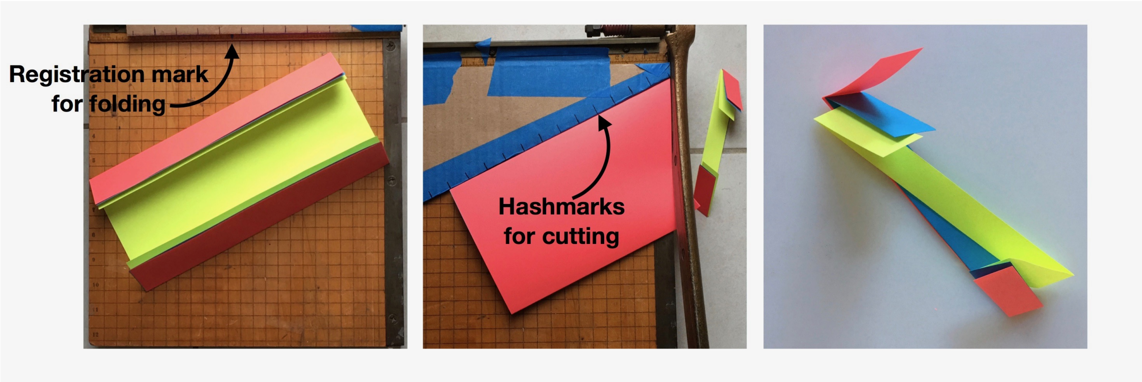

Figure 3: *A paper trimmer can make weaving units 3-at-a-time from multicolored decks of paper. To guide folding, a taped-on registration mark shows where to position the deck of letter-sized sheets. To guide trimming, hashmarks spaced along the hypotenuse of a 30-60-90 degree triangle. Cut with the folded flaps facing down to make units that match these.*

folds. Tape a registration mark on the ruler of your trimmer to show where the edge of the deck should be when it overhangs the cutting edge by the width of the desired fold.[1] Use the cutting edge to fold a crease; then unfold the crease and rotate the deck to make the parallel crease in the same way. Sharpen the creases fully by pressing them against the work surface. When you have made enough folded decks, tape the 30-60-90 triangle down to the ruler and use hashmarks on the triangle to cut equal-width, triplets of units. Make your cuts with the flaps folded underneath if you want to make units of the same handedness as in the illustrations.

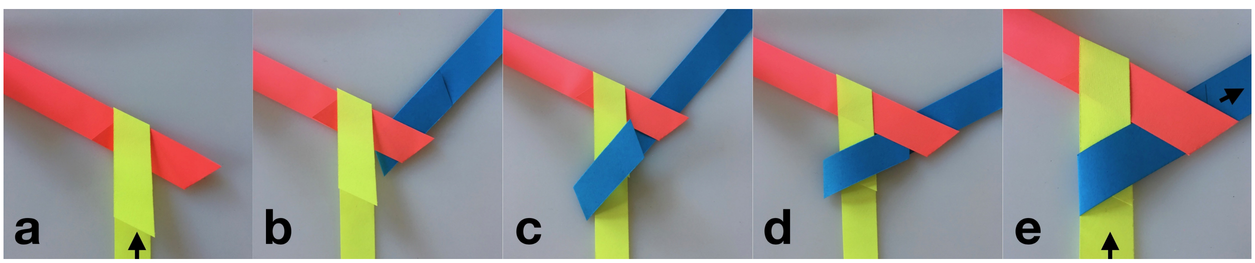

Figure 4: *To make a vertex: (a) wrap the flap of the current unit (here yellow and bearing arrow) around the folded flap of the new lefthand unit (here red), leaving a small (red) triangle of the flap exposed, (b) 'thread the needle' with the new righthand unit (here blue), (c) push the righthand unit through far enough to wrap its flap around the current unit (yellow), (d) tuck the point of flap back through the eye of the needle, (e) pull evenly on all three units to tighten the vertex: the unit with the next path color (blue) now becomes the current unit.*

Making the Vertices

The weaving units lace together three-at-a-time to build a vertex, one vertex for each letter in the code. Everything is oriented by the most recently added unit in the working path, which we call the *current unit.* The working path alternates two colors: instead of 'follow the yellow brick road' we follow the yellow/blue road. The units are always worked with the flaps facing up. Hold onto the current unit at all times until there is a new one. Figure 4 shows how the lacing is done. Before you start the next vertex, you will need to know which side the non-path color (red) will be on: 'left' or 'right' in the mnemonic name of the letter (see below)

[1]There is a definite optimum for the unit dimensions, for paper the width of US letter size, I recommend a 44 mm overhang for folding and 19 mm spacing of the hashmarks on the cardboard triangle.

contains this information. It is convenient to assemble every vertex in a counterclockwise order. Whichever color is on the left gets wrapped by the flap of the current unit (4a), leaving only a small triangular portion at its tip exposed. That leaves an opening, or 'eye of the needle', to push the folded end of the third unit through (4b), push it through far enough to wrap its flap around the current unit (4c), then tuck it back through the eye of the needle (4d), and tension all the units to pull the vertex snug (4e). The unit, left or right, with the alternate path color now becomes the current unit.

The Shape Names for the Letters

Each of the four letters, a, b, c, d, conveys two aspects: open/close and left/right. Luckily, the shapes of the letters are mnemonic. Figure 5 makes the case for renaming the first letters of the alphabet: open-left, close-left, open-right, close-right. (As a preparation for making a first basket, try spelling out your derivation this way.)

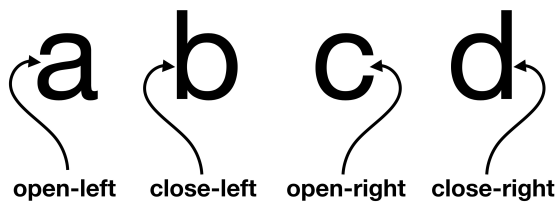

Figure 5: *Mnemonic names for the first four letters of the alphabet.*

Building Letter Sequences

Start with a yellow unit. The first letter in the expression tells you which side the non-path color (red) is on at the first vertex. Figure 6 shows the start of a basket that begins with 'a', or as that letter is read "open left." That means red goes on the left, and therefore the next path color, blue, must go on the right. Upon completion of this first vertex, blue becomes the current unit. The second letter in this example is 'c', which is read "open-right", so this time the unit on the left is the next path color, yellow, and red is on the right. The third letter is 'b', which is read "close left". 'Close' means pick up a red unit already in the work. Which one? The nearest (most recently added) available one on the side mentioned. In this case, the letter says to look on the left side. Closing is more difficult than opening, your fingers may have to get creative if space is limited. When you finally come to the last letter, you do not add any new units, the units you need are already in the work.

Some History

In 1967, Ronald C. Mullin, then a recent PhD on the faculty of the University of Waterloo, discovered a simple way to encode the topological information in any plane graph as a codeword on a four letter alphabet [7, 1]. A particularly clear description of Mullin's encoding is in [8]. Recognition of the importance of what Mullin had discovered, and his priority, has only come more recently [1]. When I showed basket making from the equivalent code at Bridges 2012 [5], it had been known for a long time that the code, also known as a shuffle of parenthesis systems [3] or a shuffled Dyck language, can describe 3-valent Hamiltonian graphs on the sphere (a dodecahedron is one example.) Mullin's sometimes neglected discovery is a reversible construction that takes *any* graph on the sphere to a 3-valent Hamiltonian graph on the sphere—and back again. It's the new universality that makes his encoding exciting [6].

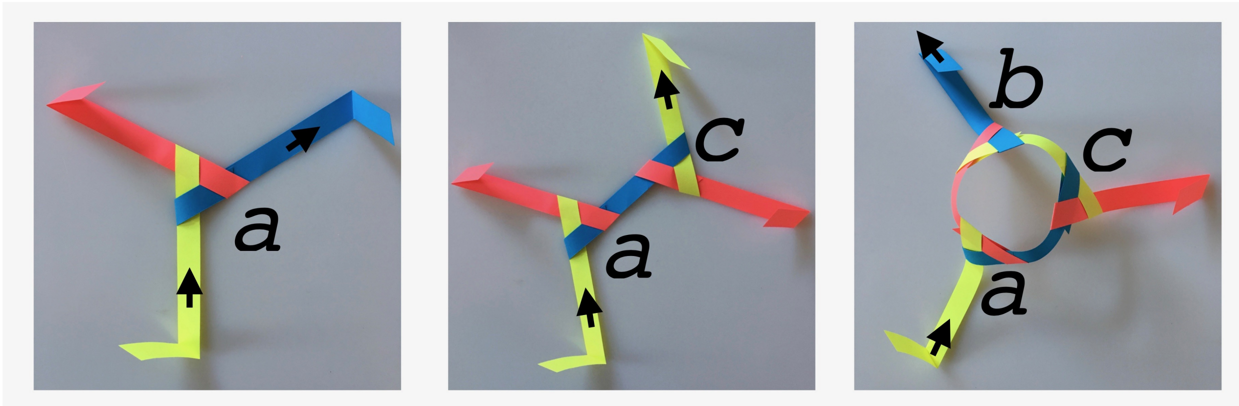

Figure 6: *Example of following a letter sequence: (left panel) open-left, (middle panel) open-left, open-right, (right panel) open-left, open-right, close-left. Path units alternate yellow/blue.*

Summary and Conclusions

I have presented a proposed a creative class activity that does not require deep knowledge, but nonetheless lets students play with some ideas of comparatively recent interest to mathematicians. I hope to report some real world experience in teaching this activity in the near future.

Acknowledgements

The author is grateful for the improvements reviewers' attentive corrections and suggestions have made to this paper.

References

[1] Olivier Bernardi. Bijective Counting of Tree-Rooted Maps and Shuffles of Parenthesis Systems. *The electronic journal of combinatorics*, 14(1):9, 2007.

[2] Jerome S. Bruner. *The Process of Education*. Harvard University Press, 2009.

[3] Robert Cori, Serge Dulucq, and Gérard Viennot. Shuffle of Parenthesis Systems and Baxter Permutations. *Journal of Combinatorial Theory, Series A*, 43(1):1–22, 1986.

[4] Martin Dowd. *Introductory Algebra, Topology, and Category Theory*. Hyperon Software, 2006.

[5] James Mallos. Evolve Your Own Basket. In Douglas McKenna Robert Bosch and Reza Sarhangi, editors, *Proceedings of Bridges 2012: Mathematics, Music, Art, Architecture, Culture*, pages 575–580, Phoenix, Arizona, 2012. Tessellations Publishing. Available online at http://archive.bridgesmathart.org/2012/bridges2012-575.html.

[6] James Mallos. A 6-Letter DNA for Baskets with Handles. *Mathematics*, 7(2):165, 2019.

[7] Ronald C Mullin. On the Enumeration of Tree-Rooted maps. *Canad. J. Math*, 19:174–183, 1967.

[8] Gilles Schaeffer. Planar Maps. *Handbook of enumerative combinatorics*, 87:335, 2015.