Evolve Your Own Basket

James Mallos
Sculptor
3101 Parker Avenue
Silver Spring, MD, 20902, USA
E-mail: jbmallos@gmail.com

Abstract

By playing a word game, participants "evolve" an original basket design through a sequence of spelling mutations. They then learn how to weave their newly-evolved basket through the easy-to-learn technique of *unit weaving*. This fun game has connections to formal language theory, graph coloring, genetics, evolutionary theory, and physics.

Background

Making curved, closed surfaces in the fabric arts can be a headful of trouble. Curvature is typically obtained by counting *increases* and *decreases*, and full closure is typically obtained, in the last resort, by sewing. In this workshop we will learn a weaving technique that weaves closed baskets idiomatically, with no thought to increases, decreases, or sewing. The weaving is directed by "words" written in the four-letter alphabet {**u, n, d, p**} superficially resembling the {A, C, G, T} alphabet of DNA. The approach we will take in the workshop is to first master the simple rules for "evolving" words in this *undip* language; to each evolve such a word; and to then learn to weave the baskets our unique words describe.

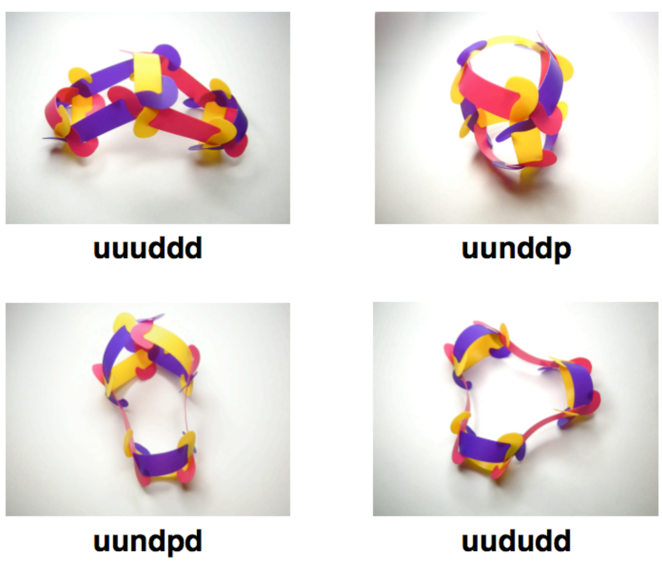

Figure 1: *The four basket shapes named by the 70 undip words of length 6.*

Formal Languages

Definitions. A *formal language* is a set of *words* (character strings) spelled using only letters from a finite set of letters called an *alphabet.* The *length* of a word is the number of characters in the word. For example, the undip word **nnuuppdd** (which names a cube-like basket) is a word of length 8. By convention, every formal language includes the unique, zero-length word called the *empty word,* symbolized by the Greek letter **ε**.

A formal language is called *algebraic* if its membership is completely specified by a set of rewriting rules. Undip is an algebraic language. Thus, undip is the set of character strings {**ud, np, udud, udnp** ... etc.} containing all the words that can be formed by recursively applying the rewriting rules that will be specified below. The empty word serves as the starting point in generating a language from its rewriting rules.

Undip

Undip is an algebraic language on the alphabet {**u, n, d, p**}. Taking note of the direction of their strokes, we will call **u** and **d** *up letters,* and **n** and **p** *down letters.*

The Re-Writing Rules for Undip. The following re-writing rules suffice to form all the words, and *only* the words of undip:

> **Insertion Rule:** ***ud*** *or* ***np*** *can be inserted anywhere.* (In other words, those sequences can be inserted before, after, or between the letters of any word.)
>
> **Shuffle Rule:** *Where an up and a down letter are adjacent, their order can be reversed.*

Inverses of the Re-Writing Rules. It is easy to see that *shuffle* is its own inverse: applying it twice to the same letters returns the original word. *Insertion,* on the other hand, has a distinct inverse: *deletion.* Deletion can be used can to generate a shorter undip word:

> **Deletion Rule:** *The subwords* ***ud*** *or* ***np*** *can be deleted from any word.*

Questions and Exercises. Is there an undip word of odd length? The number of undip words of length 2n is given by the product of consecutive Catalan numbers, which are namely the numbers in the sequence {1, 1, 2, 5, 14, 42, 132 ...}. Can you name the (1 x 1 = 1) undip word of length 0? See if you can find all (2 x 5 = 10) undip words of length 4. Which of these ten words are just one edit (one "mutation") apart?

Playing the Word Game

Starting Genesis-like from the empty word, we will, each at our own whimsey, apply the above two (or optionally three if we use deletion) rewriting rules recursively to "evolve" a unique undip word. To document the evolution of our words, we will use proofreaders' marks as exemplified by the notes reproduced below. Notice that after indicating an edit with proofreaders' marks, a clean, edited copy is written out on the next line below—ready for another "mutation." The first line of an evolution is invariably an insertion into the empty word.

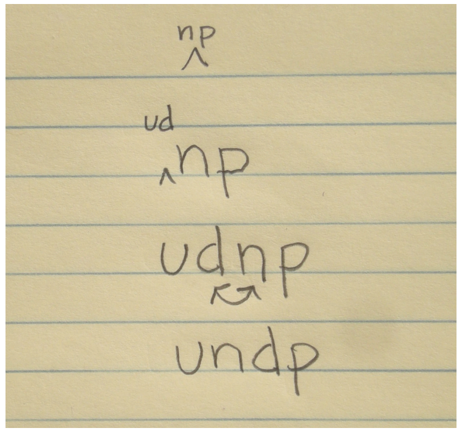

Figure 2: *Markup for the evolution of an undip word of length 4.*

Unit Weaving

Weaving One Crossing at a Time. The basket weaving technique we will use to make baskets from our undip words is called unit weaving. All you need to learn about unit weaving is how to join three unit-weavers, called *twogs,* together in a Y-shaped join which we will call an *event.* Figure 3 shows the steps. Twogs (like playing cards) should always be held with the proper side facing you. Viewed from that proper side, the end of the twog looks like a curled right hand (see first frame in Figure 3.)

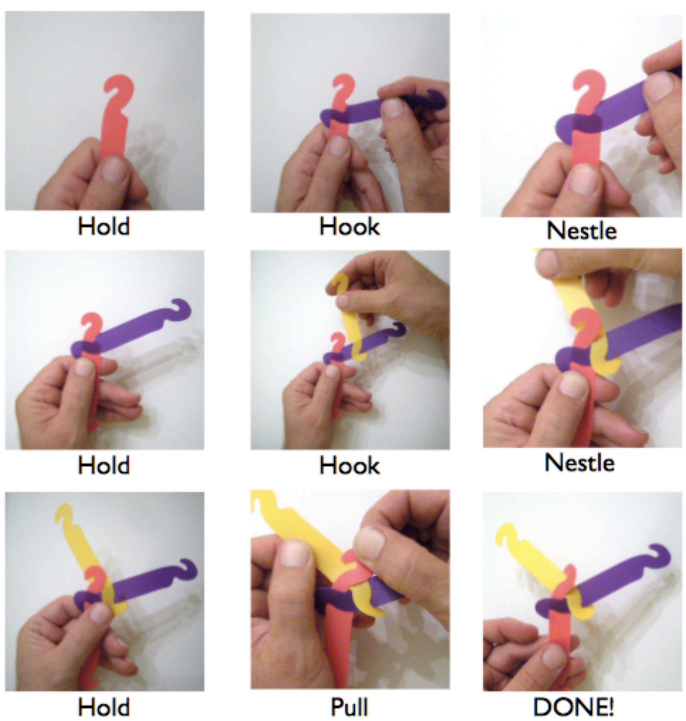

Figure 3: *Three twogs join together in a Y-shaped join called an event.*

Having reached the "done!" frame in Figure 3, the next step would to take hold of either the yellow or purple twog (i.e., follow either the left or right branch of the Y,) and repeat the assembly process to build a second event connected to the first. Practice making a few events. Can you make one without looking?

Questions: Can you find a mirrored or "left-handed" way to put twogs together? Is an event built in the left-handed way actually different an event built in the right-handed way? Do you think twogs can be used to weave a nonorientable surface such as a Mobius strip or a Klein bottle? Why, or why not?

Baskets as Feynman Diagrams

Weaving Electron Stories. Many things will be made more vivid and easier to remember if we see the baskets we are going to weave as Feynman diagrams [1]. That is, in each basket we will read a simple story of an electron and the photons it emits and absorbs. In quantum mechanics the probability of an event depends on a sum taken over all possible ways the event can occur—a "sum over histories," in Feynman's phrase, that lead up to the event. A Feynman diagram isolates just one of these histories for consideration. As such, we are never so lucky that a single Feynman diagram corresponds to a real phenomenon—full accuracy always requires an infinity of ever more complicated diagrams—but weavers *are* lucky enough that a Feynman diagram can correspond to a real basket.

Twogs come in three colors, and there will always be three different-colored twogs joined at each event. Choose the lightest color (yellow) as your *photon color.* The other two colors, pink and purple, become by default your *electron colors.* At each event the electron emits or absorbs a photon, thereby experiencing a change in energy and momentum. Our electron changes colors at each event (pink to purple, or purple to pink) as an emblem of these changes.

Let's look at the basket being woven in Figure 4, and try to read it as the story of an electron. The first frame in Figure 4 says our electron starts out in its pink color, emits a photon to the left, and continues on in its purple color. The second frame says that the electron experiences a second event where it emits a photon to the right, its color changes back to pink, and it continues on.

The third frame shows a third event where now the electron absorbs a photon on its left. Which photon does it absorb? (In this particular basket the answer is quite easy because only one photon has been emitted on the left side.) Because our baskets always have spherical (a.k.a., planar) topology, we can always use this simple rule:

> **First-to-Hand Rule:** *to find the correct photon to absorb, run a hand backwards along the work on the side you are looking for a photon. The first that comes to hand is the correct one.*

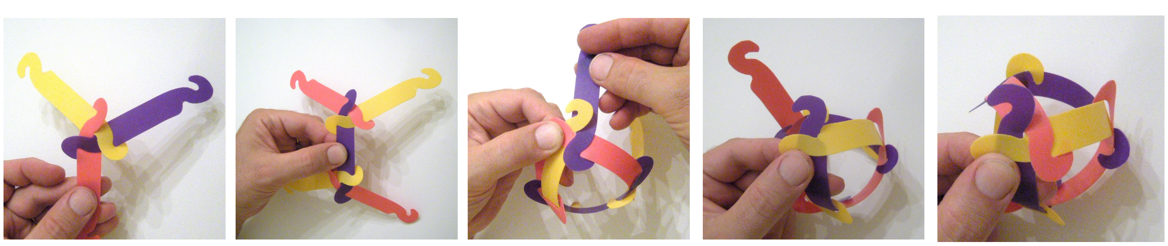

Figure 4: *A basket as a Feynman diagram: the lightest (yellow) twogs represent photons.*

The fourth frame in Figure 4 shows a fourth event in the electron's story—the absorption of a photon on the right—but in a partially completed state. This event is a little different because the final photon is being absorbed, thus completing our basket, and taking our story back to its beginning. Notice that the last twog joined is actually the other end of the very first twog. In last frame the work is nearly finished—we just need to place the remaining (purple) hook behind the last-added (pink) twog. (Weaving the last event is always a little fussy because all the twogs are already anchored in the work.)

Reading Undip Words

Bottoms Up. We now know how to make a basket that tells the story of an electron, all we need to add is the ability to read such a story in an undip word. That part is easy. Rotate your word so that you are reading from the bottom up. Open letters emit photons, closed letters absorb them. Can you guess on which side the emission or absorption occurs? That's all there is to it.

Figure 5: *Undip words are read sideways, from the bottom up. Open letters emit photons, closed letters absorb them. Can you guess to which side?*

One more rule will be handy in getting your photons to emit and absorb on the side you want:

Left-is-Last Rule: *the color you want on the **l**eft side is the **l**ast added in building the event.*

How it Works

Every closed surface can be approximated by a triangulation, or, roughly speaking, as a polyhedron with faces that are all triangles. Such a triangulation has a dual having three edges meeting at every vertex. Given any triangulated surface, it is the dual that can be unit-woven. Taking the icosahedron as an example of a triangulated surface, Figure 6 shows the close relationship between a triangulation (the

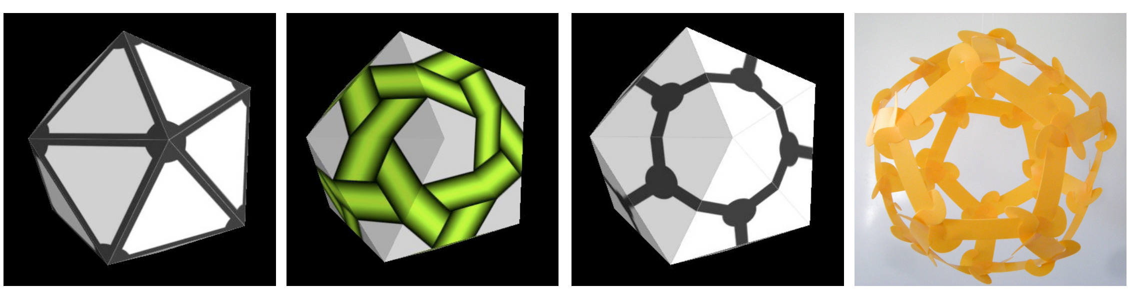

Figure 6: *A triangulated surface, a woven basket of the same shape, the dual of the triangulation, and a unit-woven basket of the same shape.*

icosahedron), a kagome basket basket of the same shape, the dual (the dodecahedron), and a unit-woven basket of the same shape. This same way, every closed surface can be approximated by both a classical kagome basket and a unit-woven basket.

A special sort of dual is needed for us to read the story of an electron in it: it must have a *Hamilton circuit*. A Hamilton circuit is a closed path that visits every vertex without re-traversing any edge. A chosen Hamilton circuit becomes the path of the electron in the story. Most 3-regular graphs have Hamilton circuits. If there are none, or one is too difficult to find, Gopi and Eppstein [2] have described an algorithm that edits a graph in subtle ways in order to create a Hamilton circuit.

Any polyhedron of spherical topology can be stretched to lie flat on the plane with all but one face showing. This is called a *Schlegel diagram* of the polyhedron. We will only need to see the edges of the missing face, and they are still present as the perimeter of the Schlegel diagram. Figure 7 shows a Schlegel diagram of a cube, with a Hamilton circuit highlighted in red. We can stretch the diagram still more to turn the Hamilton circuit into a circle. Choose some mid-edge on the Hamilton circle as a starting location for the electron's story, and choose a direction (clockwise or counter-clockwise) to travel. Each time we encounter a black edge for the first time, it is recorded as a photon emission, the second time it is recorded as a photon absorption. When we complete the circuit, we will have recorded an undip word that describes the basket. This technique for describing a planar map was described by Cori et al. in 1986 [3].

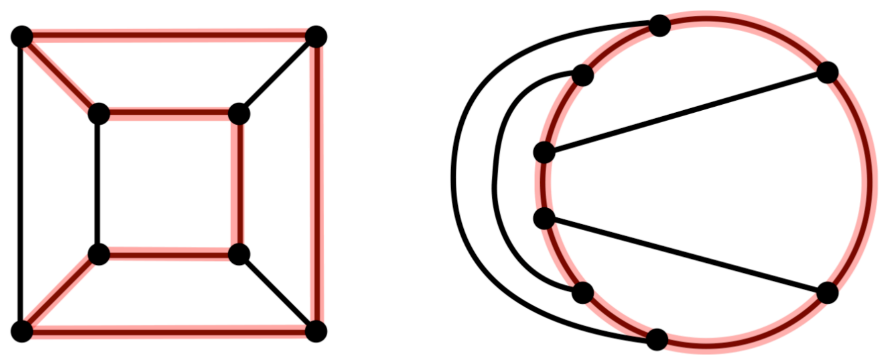

Figure 7: *A Schlegel diagram can be deformed so that a particular Hamilton circuit becomes a circle.*

Questions: Try finding a Hamilton circuit in a graph that can be drawn on the plane without crossings. Does every such graph have a Hamilton circuit? If you find one, can you stretch your drawing to show the Hamilton circuit as a circle? Has another student woven a basket of the same shape as yours following a different word? What is the smallest number of "mutations" that can turn the one word into the other?

References

[1] R. Feynman, *QED: The Strange Theory of Light and Matter,* (1985), Princeton Univ. Press.

[2] M. Gopi and D. Eppstein, "Single-strip triangulation of manifolds of arbitrary topology", *Proc. 25th Eurographics,* (2004).

[3] R. Cori, S. Dulucq, G. Viennot, "Shuffle of parenthesis systems and Baxter permutations", *Journal of Combinatorial Theory Series A,* 43:1, (1986).