How Jumps are Incorporated into the Web version of String Art

Vertex jumps takes on a special role within the string art model. If no jump is noted, the assumption is that one moves from vertex to vertex one at a time ending in an *n*-gon. This may also be explicitly noted by saying *J* = 1. The number of vertex Jumps, *J*, is conceptualized as a constant jump pattern; a 5,2-star means you draw a series of connected lines connecting at every other vertex to create a pentagram in Part I. The *n,J*-star creates the vertex frame, VF, on which string art images are created in Part II. The difference between successive used vertices is CONSTANT in Parts I and II. In Part III we relax this assumption and provide for *Jump Sets* of two or more distinct values before repeating the pattern of jumps. The web version allows each of these possibilities.

Absent explicit instructions to the contrary (via different numbers for the parameters), the opening image for the web version of the model, playingwithpolygons.com creates an 8-point Brunes Star, the same image used to explain the difference between *n*, *S*, and *P*. The web version image description (*n,S,P*) is (4,12,30) and notes that *Total Jumps* is 0.

Images from the web version: Polygon: (4,12,30)

Total Jumps

0

These are shown because *Total Jumps* = 0 means, there is no set jump set. It DOES NOT mean that one does not move from vertex to vertex. If you click the ⬍ arrow up 1 so that Total Jumps = 1, the IMAGE DOES NOT CHANGE but the image description changes to **(4,12,30,J(1))** and a new parameter, *Jump 1*, is created beneath the *Total Jumps* box as seen to the right. **Jump 1 is the J that is used throughout Parts I and II of the book**.

Total Jumps

1

Jump 1

1

Why is *J* different? The description of the image treats *J* differently from the other three parameters. It seems odd to put the jump inside a set of parentheses. This is done because *J* can take on more than one value once jump sets are allowed (in Part III). It is also why we describe images as (*n,S,P,J*) even though *n* and *J* are used first (to create the VF) then *S* and *P* are used to create the image from the VF.

The box at the top of the jumps area, *Total Jumps*, refers to the number of jumps in a *Jump Set*. A *jump set* is a sequence of jumps and is denoted by *k* in Part III explainers. Take an example using *n* = 12. Both 1 and 5 are coprime to 12 so that if *J* = 1 you have a 12-gon, and if *J* = 5 you have a 12,5-star. But if you have *Total Jumps* = 2, *Jump 1* = 1, and *Jump 2* = 5, the VF is a long rectangle connecting 0-1-6-7-0 as you can see from this (I2,1,1,J(1,5)) image. Interesting images with larger jump sets abound (check out this *k* = 13 bird beak) but it is wise to learn to walk before you run. Master *k* = 1 first.

Do not increase *Total Jumps* past 1 until you understand the basics. When you want to create stars, or create VFs that are stars, you must have *J* > 1. That does not mean that *Total Jumps* > 1. It means that *Total Jumps* = 1 and *Jump 1* > 1.

To take the simplest example, *suppose you want to create a 5,2-star or pentagram*. Using the playingwithpolygons.com (4,12,30) Brunes Star as the starting point, this can be done in three steps.

1) To remove string art complications, so set *S* = *P* = 1.   The resulting image is a square.
2) To get a star you need 5 vertices, so set *n* = 5.   The resulting image is a pentagon.

To turn this pentagon into a pentagram, you need *J* = 2. Suppose you had not carefully read the above. It would seem natural to set *Total Jumps* = 2.

Bad 3)   If you set *Total Jumps* = 2, you end up with a pentagon. As each new jump is added that jump is set to a jump of 1. Therefore, if *Total Jumps* = 2, you have *Jump 1* = 1 and *Jump 2* = 1 which are not distinct from one another, both are jumps of 1 and could have been modeled using *Total Jumps* = 1 (or 0).

Good 3)   Set *Total Jumps* = 1 and *Jump 1* = 2.   The resulting image is a pentagram.

Where is *J* in (*n,S,P,J*)?   This has already been stated above but it is worth repeating. To set a constant number of vertex jumps, *J*, per line used to create the VF, **set *Total Jumps* = 1 and *Jump 1* = *J***.