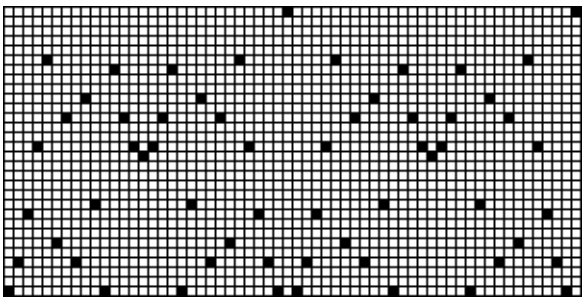
Sequence Normalization

T-sequences (threading and treadling sequences) derived from integer sequences by modular reduction [1] often have missing values. Taken as-is, such t-sequences would not utilize all the available shafts or treadles and might require more loom resources than are available.

An example is t-sequence for the squares modulo 30:

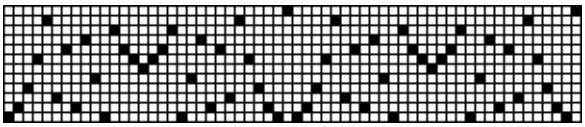


n^2 mod 30

Although the values run from 1 to 30, 18 values are missing; there are only 12 different values and this t-sequence, if appropriately modified, could be used with 12 shafts or 12 treadles.

There are several ways this t-sequence could be modified so that the values run from 1 to 12. One is *compression*, in which values are moved down to fill in the gaps — in other words, just striking out the blank rows.

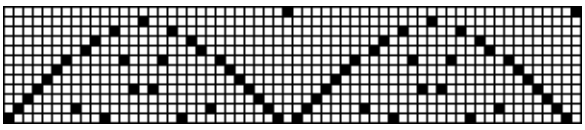
For the example above, the result is



n^2 mod 30 compressed

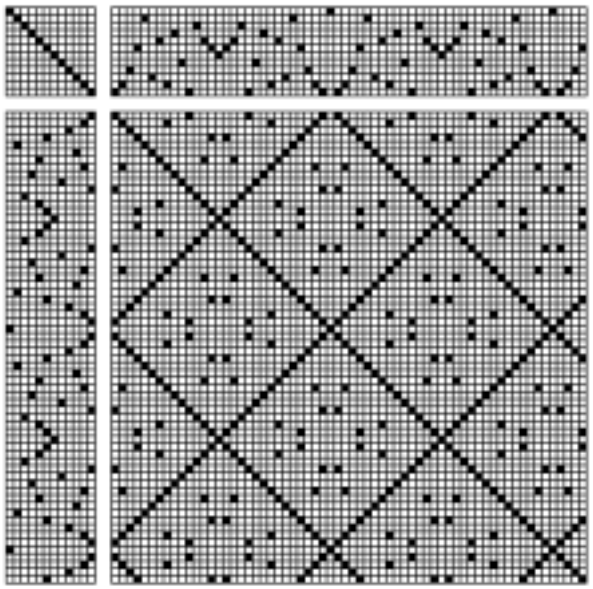
Another method, called *normalization*, changes the values as they first appear from left to right to the integers in order: 1, 2, 3, ….

For the example above, the result is

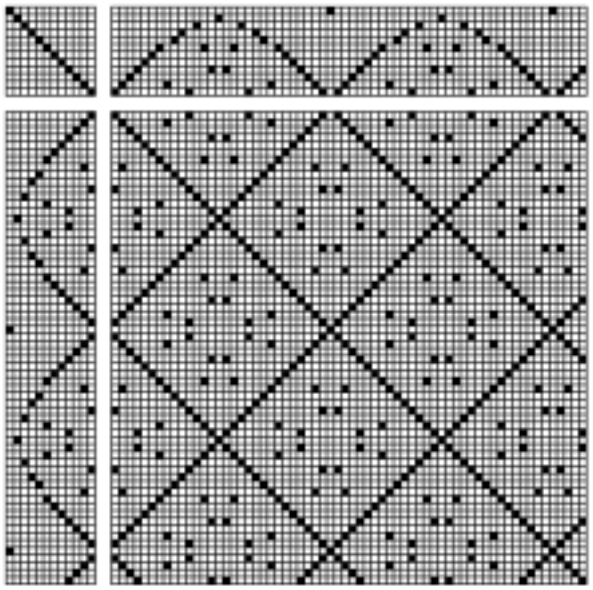


n^2 mod 30 normalized

Compressed and normalized t-sequences are, in general, very different in appearance, as this example illustrates. Yet they are simply different permutations (rearrangements) of rows of the same pattern. This is dramatized by the fact that if used in drafts with a direct tie-up and treadled as drawn in, they produce the same drawdown:



compressed

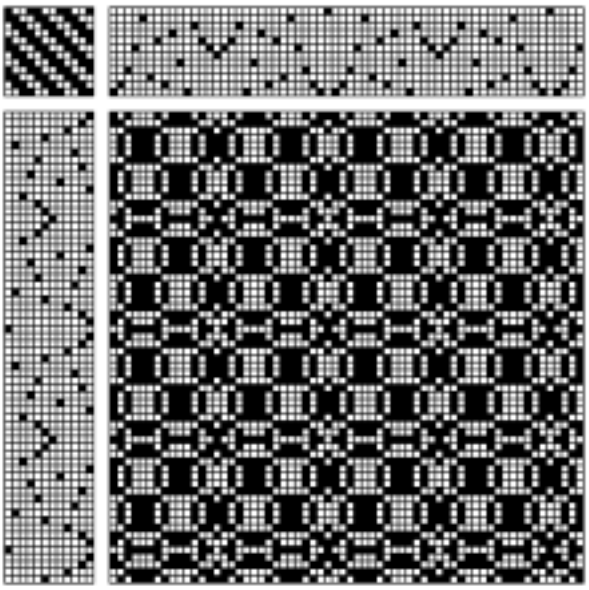


normalized

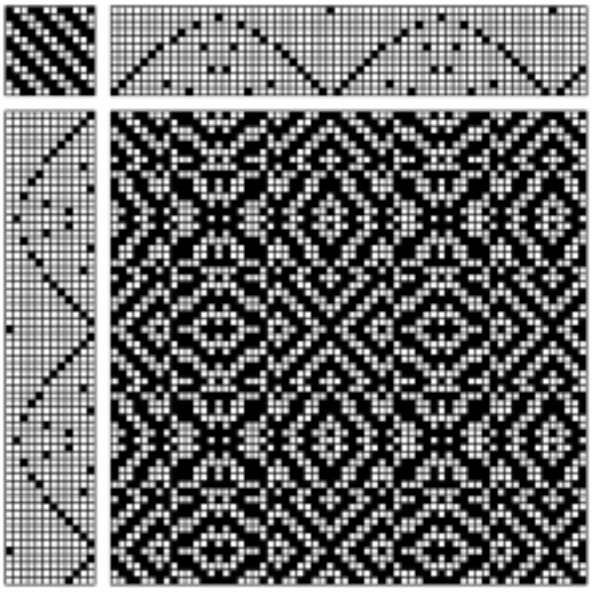
The original, uncompressed sequence also produces the same drawdown.

December 24, 2003; last revised August 1, 2004

If other tie-ups are used, however, the drawdowns generally are very different. Here are the results for a 2/2 twill tie-up:



compressed



normalized

Normalization has several advantages over compression. Because values are reassigned in order starting at 1, normalized t-sequences generally have more coherence and appear visually more attractive than compressed sequences. One special consequence of this is that if all the values in a repeat of the sequence are different, the normalized form is a straight draw and can be dismissed as aesthetically uninteresting.

Alternatively, the unnormalized version of such a sequence can be obtained by permuting the values in a straight draw.

Normalization can be applied to any sequence, not just ones with missing values. For example, normalizing the compressed sequence used in the examples here produces the same results as normalizing the uncompressed sequence.

In general, normalization provides a good method for comparing sequences.

Finally, note that normalization converts sequences with 0s and negative values to sequences with all positive values.

Reference

1. *Drafting with Sequences*, Ralph E. Griswold, 2004:
http://www.cs.arizona.edu/patterns/weaving/webdocs/gre_seqd.pdf

Ralph E. Griswold
Department of Computer Science
The University of Arizona
Tucson, Arizona