Formatting mesostic poems à la John Cage

David Bellows

Abstract

LaTeX is useful not only for producing beautifully typeset math and science papers but can be just as useful for the typesetting demands of modern poetry. This includes a style of poetry called *mesostics* that was created by the poet Jackson Mac Low and embraced and extended by the late composer John Cage; it is Cage's approach to mesostics that I will be addressing in this paper. I will show how I use LaTeX to automatically format my own computer-generated mesostics.

1 Introduction

Mesostics are a type of poetry and formatting that the composer John Cage used throughout the latter part of his career to create poetry by "reading through" a large source text looking for words containing the letters of the *spine* or *mesoword*, where both the source text and the spine were chosen for their significance to him. The resulting text would then be formatted in a way similar to an acrostic but with the spine word running through the middle. An example demonstrates this best:

```
          nearLy napping,
            camE a
          tappiNg,
            as Of
       gently Rapping,
  at my chambEr door.
```

In the above example (generated by my software), the source text is Edgar Allan Poe's *The Raven*. The spine is the name "Lenore" (from the poem) and can be seen as being the only letters in caps and running down the middle.

The software I use goes through whatever source text is given looking for words in the order they occur in the spine word (e.g., "Lenore") and selects extra words on either side of the spine word according to various rules and options. It then formats the results in one of three styles. The *regular* style, as you see above, was by far the most common style Cage used.

My software, The Platonic Music Engine (`www.platonicmusicengine.com`) (PME), exists to recreate all artifacts of human culture algorithmically with a high degree of human interaction, allowing the user to create unique works that bear a superficial similarity to existing ideas.

These artifacts of culture include music, visual art, poetry, divination, gaming, and anything else I can come up with.

In this case my software is being used to generate mesostics based on the user's choices (source text, spine, various styles of formatting, various other rules and options) and then printing out the result using LaTeX.

My software does all the heavy lifting. Not only does it find the mesostics, it also generates a `tex` file with all the necessary formatting in place. A better TeXnician would offload more work onto (La)TeX but I am not that person. I am also not much of a programmer; I just barely manage to get things working. Nonetheless, this particular workflow works very well for me. I generate the results the user wants and then use various external programs like LaTeX, Csound, LilyPond, etc., to handle all the output and make me look good.

All the programs that I use for output compile their results from text files. This is an extremely handy method of generating content.

The PME is released under the GNU Affero GPL v3 and can be downloaded from the above website. I welcome any additions to or comments about the software.

2 Regular style of mesostic

In my initial attempts at formatting, I used the `alltt` package (such as in the above example) and had my software figure out the spaces needed at the beginning of each line to make everything work out.

Since `alltt` uses a monospaced font, the process was pretty straightforward. Unfortunately it doesn't match Cage's published versions which used non-monospaced fonts and still managed to get everything perfectly spaced. I do not know how this was done in the various books Cage published that contained mesostics, but I'm guessing that a typesetter manually adjusted the letters as needed.

Since the goal of my software is to create content without the user having to tweak any of the output, I had to use an automated method but something better than what I was doing.

After struggling on my own for a while, I did what many of us do and asked for help from the TeX forum at stackexchange.com. Several people provided some great help that automated the formatting of mesostics but then I ran into various problems.

The overarching problem was that I couldn't understand those solutions at all. Anything beyond commands like `\textit` and there's a good chance I'm not going to be able to follow your solution.

The two basic problems I ran into is that unlike with Cage's original formatting, I wanted to allow

doi.org/10.47397/tb/43-3/tb135bellows-mesostic

the spine letters to be in bold. I feel that this helps them show up better.

I also noticed that in the *regular* style, Cage had the spine letters perfectly centered where the center of each letter ran down through the center of the other spine letters (you can typically only see this with the letter "I" but it does depend on the font being used). But in a different style, the left side of each spine letter was aligned with the other spine letters (see the *Merce Cunningham* style below).

I could not figure out how to make both of those changes in any of the offered solutions and I didn't want to keep bothering people about this stuff. So I went back to trying to figure it out myself.

I couldn't get tables to work. I can't remember why, but the kind of formatting I needed was too fine and I just couldn't get the level of control I needed.

So then I started using various multi-column packages. These worked better. The idea is that all the words to the left of the spine letter would be in one column set to flush right and then the spine letter and the rest of the words would be in the right column with no space between the columns.

I ran into the same problems as before with centering and bold faced fonts until I found a rather old package, `parallel`. In addition to separating the page into two columns, it makes it easy to use custom widths for the columns. See figure 1 for an example generated by my software.

near**L**y napping,
cam**E** a
tappi**N**g,
as **O**f
gently **R**apping,
at my chamb**E**r door.

Figure 1: Example of a PME mesostic using the regular style.

Here is the code for the first line:

```
\newlength{\charwidth}
\setlength{\charwidth}{\widthof{L}}
\begin{Parallel}{.5\textwidth - .5\charwidth}
     {.5\textwidth + .5\charwidth}
  \ParallelLText{\hspace*{-5cm}\raggedleft{near}}
  \ParallelRText{\textbf{L}\mbox{y napping, }}
\end{Parallel}
```

The software uses LaTeX to calculate the width of each spine letter ("L", in this example) and sets this value to the variable `\charwidth`. This value is then used to work out the space between the two columns such that the center of the letter is in the center of the page. Each line uses this same approach so that the spine letter is always perfectly centered regardless of its width.

I added some negative `\hspace` to the left column to allow any extra long lines before the spine letter to run off the page instead of breaking at the end of the line and messing up the formatting. Cage imposes a 45 character limit on either side of the spine letter (which my software follows) which should limit this from happening with this style, but see below for where it became an issue.

For some reason the `\hspace` hack wouldn't work on the right side of the column where I had to use `\mbox` instead. And I couldn't get `\mbox` to work on the left side, thus this hybrid solution.

3 Merce Cunningham style

John Cage created a set of mesostics for his partner, the dancer and choreographer, Merce Cunningham. For this one set of mesostics he devised a different style of formatting.

The basic idea was the same but each letter of each mesostic was subject to chance operations determining the typeface to be used for it, the font, size, weight and so on. According to Cage's description of the process, he had available to him over 700 typefaces with which to generate these mesostics.

Cage made each letter touch each other horizontally and spaced things in such a way that each line would touch the line before and the line after it. Cage intended for the resulting mesostics to look like Cunningham dancing, but felt like they ended up looking more like waterfalls. See figure 2 for an example published by Cage; the spine used in this mesostic is "Cunningham". Cage did not provide an unstylized version of the text and was probably fine with this particular style of mesostic being illegible as well as mostly incomprehensible.

I chose this example deliberately because it shows more overlap and collisions between the lines than most of these mesostics. I haven't studied the entire collection in depth but it appears that having the dot on the 'i' overlap is the most common type of collision.

See figure 3 for an example of how my software generates this style of mesostic using the same source text and spine as in my previous example. As in Cage's versions, there are no included words to the sides of the spine word.

The code for the first line:

```
\usepackage[letterspace=-150]{microtype}
...
\begin{Parallel}{.5\textwidth}{.505\textwidth}
 \linespread{2}\selectfont\lsstyle
 \ParallelLText{\hspace*{-10cm}\raggedleft
   {{\fontsize{11}{1em}\selectfont
```

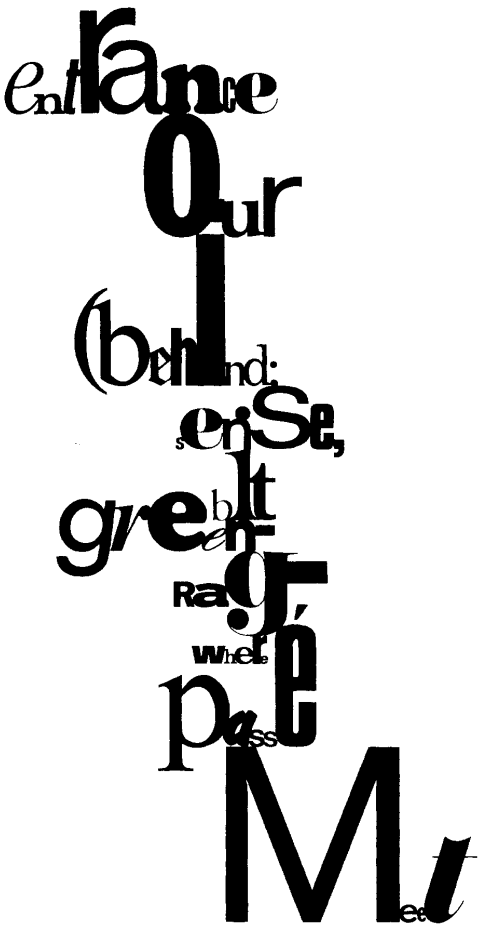

Figure 2: Example of a mesostic in the Merce Cunningham style, by John Cage. The spine is 'Cunningham'. (Page 150 from *M: Writings '67–'72* © 1973 by John Cage. Published by Wesleyan University Press. Used by permission.)

```
      {\textrm{\textit{n}}}}%
   {\fontsize{13}{1em}\selectfont
      {\texttt{\textit{\textbf{e}}}}}%
   {\fontsize{11}{1em}\selectfont
      {\textrm{\textit{a}}}}%
   {\fontsize{14}{1em}\selectfont {\texttt{r}}}%
  }}
 \ParallelRText{{\fontsize{36}{1em}\selectfont
   {\textsf{\textit{\textbf{l}}}}}%
 \mbox{{\fontsize{23}{1em}\selectfont {\texttt{y}}}%
 { }%
}}
\end{Parallel}
```

In the preamble for this style, you see my loading of the `microtype` package with a `\letterspace` of −150. I chose this number through experimentation and it seems to work well, getting the letters to touch each other horizontally but without overlapping too much. I expect that different typefaces might require different values.

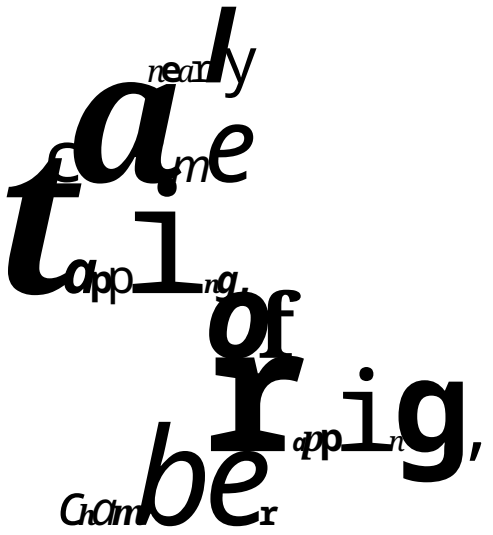

Figure 3: PME version of a mesostic in the Merce Cunningham style.

The `\linespread` command is calculated in my software based on the point size of the tallest letter in each line and multiplying that by a "magic" number that is user configurable. I experimented and settled on what I think is a good default value but the user is allowed to change this value if they want more or less space between each line which will decrease or increase the number of collisions and overlap between lines.

Another subtle change in these mesostics over the regular ones is that the spine letters, while still in the center of the page, are no longer perfectly centered with each other. Instead, the left sides of each letter are lined up and in the center of the page. If you look closely at figure 2 you can see this.

I didn't come up with any cool mathematical way to make this work so I just have the left column end at the halfway part of the page. This should mean that the spine letter starts at that point. As you can see in figure 3, this doesn't always line up perfectly. I don't know how to fix this within my software or via (LA)TEX, so I might just have to be satisfied with close enough.

You can also see in the code that I extend the `\hspace` well into the left margin. Because of the potential size of the letters in this style, you will get examples that go off the page on both the right and left sides. On an aesthetic level I think this is okay and Cage would have been fine with it.

One of the interesting problems I ran into with this style was that using different typefaces caused

all sorts of spacing issues. For example, combining Garamond with Helvetica created inconsistent spaces between letters and made centering the spine letter in this style much worse. So the unfortunate solution is to use only one typeface (Cage used many typefaces). For these examples I used Noto, which is freely available via a LaTeX package. It comes with both serif and sans serif fonts along with italics, bold, and bold italics. Other typefaces could be used but I just happen to like how this one looks by default.

The rest of the code works as with the previous example. On a personal note, I am very proud with how well this particular style turned out. It is not as elegant as Cage's versions, but given the limitations imposed on me by my skills and the nature of the project, I think my software produces pretty convincing results.

4 Ezra Pound style

The final style that I recreated from Cage is based on mesostics he generated by using Ezra Pound's *Cantos.* This was the easiest of the three. See figure 4 for an example of how my software generates mesostics in this style. I changed the spine to "Poe" to make the mesostic fit better in this publication.

seParate dying ember wrOught its thE floor
Presently my sOul grEw

Figure 4: PME version of a mesostic in the *Cantos* style.

The basic idea is that each mesostic now occupies a single line that is flush right. You still have the capital letters spelling out the spine, but they are no longer aligned vertically.

```
\begin{flushright}
 \normalsize
  se\textbf{P}arate dying ember wr\textbf{O}ught its
   th\textbf{E} floor \linebreak
  \textbf{P}resently my s\textbf{O}ul gr\textbf{E}w
   \linebreak
\end{flushright}
```

Probably the most interesting thing going on here is that the code uses the `\normalsize` command. In order to try to make sure each line fits on the page, my software adjusts the font size based on the number of characters being used in the line. This is calculated once based on the longest line and then is used for all the lines to ensure a consistent look.

5 Final thoughts

Formatting Cage's mesostics for my software was a challenge. It was a fun challenge and one where I believe the results justify the effort that went into it.

I wish I were better with (La)TeX and could turn this into a package or figure out how to add it to an existing package, as I hope someone, someday might have a need for it.

I think there are two interesting ideas to take from this paper. One is that (La)TeX, despite stereotypes to the contrary, can be extremely useful outside of STEM fields. It is an excellent tool for people working in the humanities and the arts.

The other idea is that much can be accomplished by people who don't have a tremendous amount of technical knowledge. I don't know what I'm doing most of the time but still manage to achieve my desired results. Obviously this is because of how well designed TeX and friends are and, of course, the amazing ecosystem that has grown up around it over the decades.

If I can do this stuff so can any other artist, composer, poet, gamer, ...

Finally, based on a suggestion from *TUGboat* editor Karl Berry, I am including one final mesostic (figure 5). This one uses the text of this article for the source text and the word "TUGBOAT" for the spine.

i ran inTo
always line Up perfectly.
formattinG of mesostics
at my chamBer
each letter Of
pAper).
verTically.

The spine
be in one colUmn set to
separate dyinG
on the numBer
sOurce text is
embrAced
my desired resulTs.

Figure 5: Mesostic using this paper as the source text and the word "TUGBOAT" as the spine.

◇ David Bellows
davebellows (at) gmail dot com
https://www.platonicmusicengine.com