Adaptive character generation and spatial expressiveness

Michael Cohen
Human Interface Lab, University of Aizu 965-80, Japan
mcohen@u-aizu.ac.jp

Abstract

Zebrackets is a system of meta-METAFONTs to generate semi-custom striated parenthetical delimiters on demand. Contextualized by a pseudo-environment in LaTeX, and invoked by an aliased pre-compiler, *Zebrackets* are nearly seamlessly invokable in a variety of modes, manually or automatically generated marked matching pairs of background, foreground, or hybrid delimiters, according to a unique index or depth in the expression stack, in 'demux,' unary, or binary encodings of nested associativity. Implemented as an active filter that re-presents textual information graphically, adaptive character generation can reflect an arbitrarily wide context, increasing the information density of textual presentation by reconsidering text as pictures and expanding the range of written spatial expression.

Adaptive Character Generation: Zebrackets

Zebrackets [Cohen 92] [Cohen 93] takes a small-scale approach to hierarchical representation, focusing on in-line representation of nested associativity, extending parentheses (also known as "lunulae" [Lennard 91]), and square brackets (a.k.a. "crotchets"), by systematically striating them according to an index reflecting their context.

Functionality. Table 1 crosses three of the dimensions currently supported by *Zebrackets*, using a LISP function (which performs a generalized "inclusive or") as a scaffolding.

index is the semantic value of the pattern being superimposed on the delimiters:

- **unique** generates a unique, incremental index for each pair of delimiters
- **depth** calculates the depth of the delimited expression in an evaluation stack, useful for visualizing expression complexity

encoding scheme refers to the way that the index is represented visually:

- **demux** named after a demultiplexer, or data selector, which selects one of n lines using $\lg_2 \lceil n \rceil$ selectors, puts a 'slider' on the delimiter. Such a mode is useful for establishing spatial references, as in (top)(middle)(bottom).
- **unary** creates a simple tally, a column of tick marks
- **binary** encodes the index or depth as a binary pattern, the most compact of these representations

The `demux` encoding mode always has exactly one band or stripe, but the `unary` and `binary` encodings have variable numbers, and use an index origin of zero to preserve backwards compatibility. Since the striations are adaptively chosen, the complexity of the delimited expression determines the spacing of the streaks. Without NFSS, the maximum number of stripes for a self-contained face is $\lg_2(\frac{256}{2}) = 7$. Otherwise, for overly rich expressions that exceed visual acuity, *Zebrackets* can be limited to a fixed striation depth, wrapping around (repeating) the indexing scheme if the delimiters exhaust the range of uniquely encodable values, as seen in the `unique` × {`demux`|`unary`} sextants.

type controls the style of the striations superimposed on pairs of delimiters:

- **background** bands drop out segments from the delimiters
- **foreground** explicitly put in black ticks, which are more legible if less inconspicuous
- **hybrid** combines these two styles, dropping out bands at all the possible slot locations, and then striping the actual index

Eventually perhaps, greyscale striations (not yet implemented) might interpolate between these approaches, causing the ticks to disappear at normal reading speed, but be visible when doing a detailed search.

Foreground *Zebrackets* only work well with thinner faces, and background *Zebrackets* only with bolder faces. Figure 1 exercises *Zebrackets* through an obstacle course of less common fonts, showing some of the legibility problems, even with figure/ground modes chosen to flatter the filigrees.

encoding	type	index	
		unique	depth
demux	background	(DEFUN ANY (LST) (COND ((NULL LST) NIL) ((CAR LST) T) (T (ANY (CDR LST)))))	(DEFUN ANY (LST) (COND ((NULL LST) NIL) ((CAR LST) T) (T (ANY (CDR LST)))))
	foreground	(DEFUN ANY (LST) (COND ((NULL LST) NIL) ((CAR LST) T) (T (ANY (CDR LST)))))	(DEFUN ANY (LST) (COND ((NULL LST) NIL) ((CAR LST) T) (T (ANY (CDR LST)))))
	hybrid	(DEFUN ANY (LST) (COND ((NULL LST) NIL) ((CAR LST) T) (T (ANY (CDR LST)))))	(DEFUN ANY (LST) (COND ((NULL LST) NIL) ((CAR LST) T) (T (ANY (CDR LST)))))
unary	background	(DEFUN ANY (LST) (COND ((NULL LST) NIL) ((CAR LST) T) (T (ANY (CDR LST)))))	(DEFUN ANY (LST) (COND ((NULL LST) NIL) ((CAR LST) T) (T (ANY (CDR LST)))))
	foreground	(DEFUN ANY (LST) (COND ((NULL LST) NIL) ((CAR LST) T) (T (ANY (CDR LST)))))	(DEFUN ANY (LST) (COND ((NULL LST) NIL) ((CAR LST) T) (T (ANY (CDR LST)))))
	hybrid	(DEFUN ANY (LST) (COND ((NULL LST) NIL) ((CAR LST) T) (T (ANY (CDR LST)))))	(DEFUN ANY (LST) (COND ((NULL LST) NIL) ((CAR LST) T) (T (ANY (CDR LST)))))
binary	background	(DEFUN ANY (LST) (COND ((NULL LST) NIL) ((CAR LST) T) (T (ANY (CDR LST)))))	(DEFUN ANY (LST) (COND ((NULL LST) NIL) ((CAR LST) T) (T (ANY (CDR LST)))))
	foreground	(DEFUN ANY (LST) (COND ((NULL LST) NIL) ((CAR LST) T) (T (ANY (CDR LST)))))	(DEFUN ANY (LST) (COND ((NULL LST) NIL) ((CAR LST) T) (T (ANY (CDR LST)))))
	hybrid	(DEFUN ANY (LST) (COND ((NULL LST) NIL) ((CAR LST) T) (T (ANY (CDR LST)))))	(DEFUN ANY (LST) (COND ((NULL LST) NIL) ((CAR LST) T) (T (ANY (CDR LST)))))

Table 1: index:{unique, depth} × encoding:{demux, unary, binary} × type:{background, foreground, hybrid} (10 pt. cmtcsc *Zebrackets*, selected to match size and font [small caps] of text)

ACTIVE INGREDIENT: Hydramethylnon [tetrahydro-5, 5-dimethyl-2(1*H*)-pyrimidinone(3-[4-(trifluoromethyl)phenyl]-1-(2-[4-(trifluoromethyl) phenyl]ethenyl)-2-propenylidene)hydrazone]

ACTIVE INGREDIENT: Hydramethylnon [tetrahydro-5, 5-dimethyl-2(1*H*)-pyrimidinone(3-[4-(trifluoromethyl)phenyl]-1-(2-[4-(trifluoromethyl) phenyl]ethenyl)-2-propenylidene)hydrazone]

ACTIVE INGREDIENT: Hydramethylnon [tetrahydro-5, 5-dimethyl-2(1*H*)-pyrimidinone(3-[4-(trifluoromethyl)phenyl]-1-(2-[4-(trifluoromethyl) phenyl]ethenyl)-2-propenylidene)hydrazone]

ACTIVE INGREDIENT: Hydramethylnon [tetrahydro-5, 5-dimethyl-2(1*H*)-pyrimidinone(3-[4-(trifluoromethyl)phenyl]-1-(2-[4-(trifluoromethyl) phenyl]ethenyl)-2-propenylidene)hydrazone]

Figure 1: Application of *Zebrackets* to a chemical formula (sans serif bold extended with background, sans serif with hybrid, sans serif demibold condensed with hybrid, "funny face" [negative inclination] with foreground)

Implementation. The implementation of *Zebrackets* comprises two aspects: a filter to generate permuted invocations of the underlying delimiters, and the delimiter glyphs themselves. The filter is composed of (an *ad hoc* collection of) `csh` and `sh` shell scripts and C and `perl` [Wall & Schwartz 91] programs. The two-pass filter parses selected text, invoked explicitly with editor utilities like Emacs' `shell-command-on-region` command [Stallman 88], or implicitly as a precompiler. In the latter case, sections of the document set off by the LaTeX [Lamport 86] pseudo-environment

`\begin{zebrackets}{`<parameters...>`}`

⋮

`\end{zebrackets}`

are replaced by zebracket invocations. This pseudo-environment is interpreted by a precompiler, like a macro processor, that replaces vanilla delimiters with zebracketed, and emits METAFONT [Knuth 86] source that will be invoked at image time.

The first pass parses the expression using a stack, establishes the maximum number of stripe slots needed, and generates the necessary METAFONT files. For the unique index mode, the maximum number of striations is the number of bits needed to represent its highest index, which is equal to $\lceil \lg_2 |\text{delimiter pairs}| \rceil$. Using the context established by the first pass, the second pass replaces each delimiter with LaTeX code invoking its respective zebracketed version by effectively traversing the underlying tree. As seen in Figure 1, different styles of delimiters (like rounded parentheses and square brackets) are handled separately, and the respective striation slots are spaced out evenly along the height of the delimiter.

For example, invoking the aliased precompiler/compiler on a document containing the contents of Figure 2 runs the `zebrackets` filter on "`(a * (b + c))`" (with arguments that mean "automatically generate (uniquely) indexed foreground-striated binary-encoded 10pt. delimiters using `cmr` base parameters"), determines that only one potential striation is needed, encodes the indices as binary patterns, replaces the source text with that in Figure 3,[1] and generates the `zpfbcmr10.mf` source,[2] as well as the appropriate `.tfm` and `.pk` files, which together yield "(a * (b + c))" at preview (`TeXview` [Rokicki 93] via `TeXMenu` [Schlangmann 92] on NextStep) or printing (`dvips` [Rokicki 92]) time.

By having indirected the glyphs one extra level, *Zebrackets* implements a meta-METAFONT. Dynamic fonts [Knuth 88] [André & Borghi 89] [André & Ostromoukhov 89] employ what is sometimes called "dynamic programming," which is basically lazy evaluation of a potentially sparse domain. Although each *Zebrackets* character is essentially determined at edit-time, and the actual specification involves human-specified ranges for zebracketing, because of the communication between document and METAFONT, character generation is context-sensitive and adaptive, since the automatic specification can be conceptually lumped together with the compilation (via `latex`) and imaging.

Currently the size of the delimiters and the name of the Computer Modern model font are

[1] Idempotency of font declarations is finessed by the `\ifundefined` condition [Knuth 84], pages 40, 308.

[2] The syntax of METAFONT terminates a token upon encountering a digit, so no numbers can be used directly as part of a font name. Therefore, the number of striations is mapped to an alpha character ('a'⇒0 stripes, 'b'⇒1 stripe, ...), which becomes, after 'z' [for *Zebrackets*], 'b' or 'p' [for parentheses, or brackets], and 'b', 'f', or 'h' [for back- or foreground, or hybrid], the fourth character in the font name.

```
\documentstyle[zebrackets]{article}
\begin{document}
⋮
\begin{zebrackets}{f,-1,-1,binary,10,cmr}
(a * (b + c))
\end{zebrackets}
⋮
\end{document}
```

Figure 2: Sample (LaTeX pseudo-environment) input

```
\documentstyle[zebrackets]{article}
\begin{document}
⋮
\ifundefined{zpfbcmr}\newfont{\zpfbcmr}{zpfbcmr10}\fi
{\zpfbcmr\symbol{0}}a * {\zpfbcmr\symbol{1}}b + c{\zpfbcmr\symbol{3}}{\zpfbcmr\symbol{2}}
⋮
\end{document}
```

Figure 3: Sample (*Zebrackets* filter) output

explicitly passed as parameters to the pseudo-environment. A more elegant approach would be to code the *Zebrackets* filter directly as a *bona fide* LaTeX environment, which could determine delimiter size and font at compile time (writing information to an `.aux` file and using something like "`\immediate\write18`"to escape to the operating system to create and invoke `mf` files). *Zebrackets*' implementation as a precompiler insulates the characters from useful positional and contextual information, like page position and current font and size. Otherwise, *Zebrackets* is compatible with (perhaps redundant) LaTeX dimensions, as overstated by Figure 4.

The *Zebrackets* filters slow down document compilation considerably. However, since they are usually image-level compatible, a document may be previewed quickly in a *Zebrackets*-less mode, while the cycle-intensive *Zebrackets* run in the background, eventually seamlessly strobing into the previewer without any layout change or page motion.

Spatial Expressiveness

The notion of a fixed alphabet font is inherently limited, even one extended into a family by techniques like weighting, italicization, emboldening, and local contextual tools like ligature and kerning. Computers offer the potential of "chameleon fonts," altered, depending on their context, to heighten legibility (readability, balance, or proportion) or evoke emotions that complement, reinforce, or amplify the words and ideas.

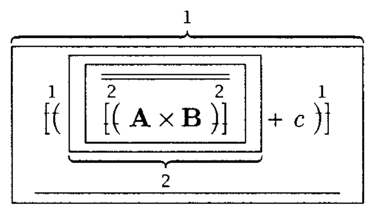

Figure 4: Celebration of nesting hyperbole: Round and rectangular tagged *Zebrackets* reinforcing interleaved (to the limits of TeX's semantic nesting stack size) tagged over- and underbraces, framing, over- and underlining, emboldening, italicization, case, natural operator precedence, and canonical left→right reading order

Zebrackets is a focused realization of adaptive character generation, useful in certain contexts, but ultimately less important than its conceptual ambitions. The logical extension of typography is arbitrarily tuned characters, calculated globally and generated uniquely. Adaptive character generation is the destiny of electronic publishing, glyphs adjusted in arbitrarily subtle ways to carry information and fit space.

***Zebrackets* and Multiple Master Typefaces.** Reading publications like Baseline and Emigre, one might think that the only computer-driven typographic innovations are on the Macintosh, using tools like Fontographer. This imbalance is perhaps because the formalization of a meta-language (and its corollary meta*-languages) is less accessible to artists than graphical techniques.

The notion of a meta-font language can be likened to Adobe Multiple Master Typeface [Adobe 92] [Spiekermann & Ginger 93] with an arbitrary number of axes, or dimensions, each corresponding to a parameter. (Selection of a base font can be thought of as setting lower level parameters.) Usually the glyph space is thought of as continuous, and the arguments, or components of the index vector, are floating point.

Zebrackets' integral characterization of the text yields a quantized specification of a font; real numbers would allow for continuous variation (within, of course, the resolution of finite precision encoding), expanding even further the ability to custom-tailor a font for a context. Such variety might manifest as arbitrarily soft typefaces, perhaps employing greyscale or dynamic effects, or tuned by the reader, to match visual acuity.

Quantification of dimensionality. Visual languages combine textual and graphical elements. Spatial expressiveness is achieved not only via effects like *Zebrackets*, but any kind of systematic control of document presentation—explicit parameters like margins, but also implicit global characteristics, like consistency or contrast of typographic features.

Words have different expressive qualities than pictures, but treating text as pictures, interpolating between 1D textual streams and 2D graphical representations, enables some of the best qualities of both. Table 2 attempts to align this spatial expressiveness with computer languages and communication modalities, suggesting that typeset documents have a dimensionality somewhere between 1 and 2.[3]

It is amusing to try to estimate the value of this non-integral dimension. We can assume that a document composed entirely of (captionless) pictures is fully 2-dimensional, and a document stripped of graphical cues, denuded ASCII, to be entirely one-dimensional, and that the interpolation between is (linearly) proportional to the fraction of the respective components, as shown in Figure 5.

Using an information theoretic assumption that a metric of a vector is proportional to its length, and that languages are Huffman encoded, so that clichéd expressions are terse, then the most expressive will be the longest, and a heuristic for spatial expressiveness is simply to compare the magnitude of a graphical file with that of the underlying text:

$$S = \frac{|graphics| - |text|}{|graphics|} + 1 \qquad (1)$$

where $|text|$ is the length of the text substrate, $|graphics|$ is the length of the graphical file (which includes all the text), and S is the dimension of spatial expressiveness. In particular, the character counts of the PostScript (`.ps`) file and the `detexed` LaTeX (`.tex`) and bibliography (`.bbl`) files can be used to characterize the relative weights of the respective components. Using this formula, a simple shell script, shown in Figure 6, calculates the dimension of this document to be about 1.94. This value is inflated by including the font encoding in the ps file, but such a dilation is an expected consequence of sharpening the granularity of the document rendering. A (perhaps not undesirable) consequence of such a definition is that dimension varies with output resolution.

In contrast, we would expect the spatial expressiveness of a graphically-challenged document, `detexed` source embraced by a minimalist compiling context to be closer to unity. As seen in Table 3, and corresponding with this intuition, small documents are mostly graphical, with dimensionality near 2, but as their textual component lengthens, they become more vector-like, with dimension closer to 1. The same text, but zebracketed, yields, as expected, higher values of spatial expressiveness, except for the lowest character counts, where the heuristic manifests artifacts of `detex` idiosyncrasies. Anyway, the test files used to generate these metrics are more than a little artificial, because empty lines are needed to prevent TeX's paragraph buffer from overflowing, and *Zebrackets*' wrap-around restriction currently makes it impossible to generate even a contrived document in which every character is unique.

The usefulness of such a metric is bounded by the validity of its model; particularly suspect is the assumption of equivalence of graphical information, yielding artifacts of an over-simplified characterization. It seems intuitive that the information in text scales according to length ("A = B" has roughly half as much information as "A = B = C"), but does, for example, a PostScript (macroscopic) `moveto` carry, on the average, the same amount of information as a same-length fraction of a (microscopic) font encoding in a document prelude? Only arguably, in a relaxed, informal sense of "spatial expressiveness." The data must be regarded as preliminary, and further analysis is indicated.

Paradigm shift: the end of fonts. As adaptive character generation becomes increasingly intricate, compressed encodings become less relevant (since each font is disposable), and the distance between the bitmap and the next least abstract representation

[3] Of course time must be considered another dimension, or design axis. Temporal techniques, like `Emacs'` flashing pairs of parentheses, will become important in ways difficult for us to imagine now, and cinematographic techniques will start to infiltrate books (as in World-Wide Web). Perhaps rotating colors through letters, or gently inflating/deflating them, will make them easier to read. And, of course, (hyperlinked) video is inherently temporal.

spatial expressiveness (dimensionality)	computer language	communication modality
1	ASCII	
	JIS, EUC [Lunde 93]	email
	Unicode	
	Rich Text Format (rtf)	typewriting
	TEX/LATEX/METAFONT	typesetting
	+ *Zebrackets*	
	device independent (dvi)	handwriting
2	PostScript (ps)	drawing & photography
	texture maps	
		painting
	Renderman (rib)	
3		sculpture

Table 2: Correspondence of dimension with hierarchy of languages and media as richness of expression

dimensionality slider	
1	2
2D partition: graphical layout, typography, ...	1D partition: textual substrate

Figure 5: Partitioning of document into graphical and textual components: the fraction of a document's graphical content determines its dimensionality

grows. Fonts as we know them will become singletons, eclipsed by transformations and geometric manipulations. Document manipulation will be organized as filters—not only conventional idioms like emboldening, underlining, size and color contexts, but also legibility sliders, path-following and space-filling constraints, visual overtones, and temporal effects. Not only will characters be morphed, but characteristics will be crossed and composed. Such promiscuous intermingling of these filters, dancing to graphical rhythms that reverberate through the document, will legitimize an intermarriage between perspectives and multiple inheritance of eclectic legacies.

An example of such local manifestation of global context, inspired by the notion of a cross-reference as a back-traversable hyperlink, can be seen in this paper's references section, whose (superimposed demux-style) zebracketed keys indicate the pages of all the respective citations. An extension to *Zebrackets* (the intricacies of which deserve another paper) automatically uses `.aux`, `.bbl`, and `.idx` files to striate the bibliographic tags for back-references, each of the delimiter slots representing a page of the document. (The body of a paper, excluding the bibliography, can be at most seven pages long, since only up to seven striations are currently encoded by *Zebrackets*.) The left delimiter points to the `\cites` and the right indicates the `\nocites`. Notice, for instance, that [Knuth 86] gets two explicit citations (one of which is here) and one invisible one.

Conclusion. The handwritten "publishing" of pre-Gutenberg scribes was arbitrarily subtle, with its attendant human caprice (and mistakes). Printing can be thought of as having rigidified this information transmission. The research described here loosens some of that determinism, not by randomizing the presented information, but by softening the digitized boundaries, thereby expanding the range of expression. Contextual fonts like *Zebrackets* indicate evolving modes of written representation, algorithmic descriptions driving adaptive displays, as style catches up to technology.

References

[Adobe 92] Adobe. Adobe Type 1 Font Format: Multiple Master Extensions. Technical report, Adobe Systems, February 1992.

[André & Borghi 89] J. André and B. Borghi. Dynamic fonts. In J. André and R. Hersh, editors, *Proc. Int. Conf. on Raster Imaging and Digital Typography*, pages 198–203, Lausanne, Switzerland, October 1989. Cambridge University Press. ISBN 0-521-37490-1.

[André & Ostromoukhov 89] J. André and V. Ostromoukhov. Punk: de METAFONT à PostScript. *Cahiers GUTenberg*, 4:123–28, 1989.

[Cohen 92] M. Cohen. Blush and Zebrackets: Two Schemes for Typographical Representation of Nested Associativity. *Visible Language*, 26(3+4):436–449, Summer/Autumn 1992.

```
#!/bin/sh
INPUTFILENAME=`basename $1 .tex`
GRAPHICS=`dvips -o "!cat" $INPUTFILENAME.dvi | wc -c`
TEXT=`cat $INPUTFILENAME.tex $INPUTFILENAME.bbl | detex | wc -c`
echo `echo 3 k $GRAPHICS $TEXT - $GRAPHICS / 1 + p | dc`
```

Figure 6: Shell script to estimate spatial expressiveness (dimensionality) of a compiled LaTeX document

target characters	pages	without *Zebrackets*			with *Zebrackets*		
		TEXT	GRAPHICS	S	TEXT	GRAPHICS	S
1	1	14	4073	1.996	49	4090	1.988
10	1	41	4193	1.990	76	6427	1.988
100	3	311	6001	1.948	346	30194	1.988
1000	23	3011	21009	1.856	3046	84223	1.963
10000	228	30011	154907	1.806	30046	226971	1.867
100000	2273	300011	1501805	1.800	300046	1661796	1.819

Table 3: Quantification of spatial expressiveness: stripped down and striped up

[Cohen 93] M. Cohen. *Zebrackets: a Pseudo-dynamic Contextually Adaptive Font*. *TUGboat: Communications of the TeX Users Group*, 14(2):118–122, July 1993. ISBN 0896-3207.

[Knuth 84] D. E. Knuth. *The TeXbook*. Addison-Wesley, 1984. ISBN 0-201-13448-9.

[Knuth 86] D. E. Knuth. *The METAFONTbook*. Addison-Wesley, 1986. ISBN 0-201-13444-6.

[Knuth 88] D. E. Knuth. A punk meta-font. *TUGboat: Communications of the TeX Users Group*, 9(2):152–168, August 1988. ISBN 0896-3207.

[Lamport 86] L. Lamport. *LaTeX: A Document Preparation System*. Addison-Wesley, 1986. ISBN 0-201-15790-X.

[Lennard 91] J. Lennard. *But I Digress: Parentheses in English Printed Verse*. Oxford University Press, 1991. ISBN 0-19-811247-5.

[Lunde 93] K. Lunde. *Understanding Japanese Information Processing*. O'Reilly & Associates, 1993. ISBN 1-56592-043-0.

[Lupton & Miller 90] E. Lupton and J. A. Miller. Type writing. *Emigre*, (15):i–viii, 1990.

[Rokicki 92] T. G. Rokicki. `dvips 5.491`, 1992.

[Rokicki 92] T. Rokicki. *NeXTTeX 3.141*, 1993.

[Rokicki 93] T. Rokicki. TeXview 3.0, 1993.

[Schlangmann 92] H. Schlangmann. TeXmenu, 1992. 4.1.

[Spiekermann & Ginger 93] E. Spiekermann and E. Ginger. *Stop Stealing Sheep & find out how type works*. Adobe Press, 1993. ISBN 0-672-48543-5.

[Stallman 88] R. M. Stallman. *GNU Emacs Manual*. Free Software Foundation, 1988.

[Wall & Schwartz 91] L. Wall and R. L. Schwartz. *Programming perl*. O'Reilly & Associates, Inc., 1991. ISBN 0-937175-64-1.