

increase in *blacker* is a decrease in the diameter of the path forming the o.

An increase in *blacker* should cause the pen width to increase, but it should not cause the pen's path to change. The extra width should be distributed equally on both sides of the stroke on the assumption that the printer will erode both sides of the stroke equally. Unfortunately, I do not see any easy way to modify the METAFONT code to make it behave this way. The best results would be obtained by moving all the points that are positioned relative to a pen edge outward by *.5blacker*. For example, we could change the second statement above to

$$y_{1r} - .5blacker = h + vround 1.500;$$

This would require an enormous amount of error prone work, however, because every point in the full set of Computer Modern fonts would have to be studied and a large proportion changed. A more tractable approach would be to remove all references to *blacker* in the definitions of pen widths and to modify such commands as **penstroke** and **filldraw stroke** to broaden their strokes by *blacker* automatically. This would limit the number of changes that would have to be made to the code, but it would have the disadvantage of nullifying the carefully planned rounding of pen widths, perhaps ruining the fonts in other ways.

Not wishing to tackle either of these projects immediately, I decided to live within the limits of the existing parameters. After many experiments I arrived at a compromise set of parameters:

$$blacker := .6;$$

$$fillin := -.3;$$

$$o_correction := .6;$$

At this value of *blacker*, most of the characters keep their original sizes, but it is not quite enough to compensate for the thinning inherent in the printer. The rather extreme setting of *fillin*, which thickens diagonals, seems to correct the remaining faint spots. (At one point I tried to use *o_correction* to enlarge the shrunken bowls of *blacker* = .75, but with hindsight I should have known better. The results were not the kind of thing that I would take outside the privacy of my own home.) There are still some ugly features in the resulting fonts, particularly an inconsistency in the weights of characters. Nevertheless, I feel that this set of parameters is considerably better than the ones that result from the "conjectural" parameters, and also better than the "am" fonts they replace. There is room for improvement, however, as well as a

challenge for the next generation of METAFONT designers.

Copies of these fonts ready for downloading to the Xerox 2700 can be obtained from me or preferably from

Margot Nelligan
Xerox Printing Systems Division
880 Apollo Street
El Segundo, CA 90245.

Updated Computer Modern Fonts for the LN03

John Sauter

Included with TUGboat volume 8 number 1 was the usual errata sheet for the T_EX programs and documentation. Among the corrections to *Computers and Typesetting, volume E*, were changes to the parameters. The effect of these changes is to change the shapes of some of the Computer Modern characters.

I have been making available "alternative" versions of the Computer Modern parameter files since TUGboat volume 7 number 4, so these changes to volume E make my files obsolete. Fortunately, the files are easily fixed. Anyone who got a tape from me not marked METAFONT version 1.3 (or later) please make the following changes. I have taken some liberties with the spacing in order to fit the corrections into TUGboat's columns. You can use whatever spacing you wish when you change the files, except that a comment that starts with % must end on the same line as the %.

In COMPUTE_CMR.MF, starting at line 128, change four lines from

```
%elseif design_size < 12:
    ((design_size*15)+150)
else: ((0.020812520812*
    design_size*design_size) +
(14.5421245421*design_size) +
(152.49750249))fi)/360pt#;
to
elseif design_size < 40:
    ((-0.23934398934*design_size*
    design_size) +
(20.265567765*design_size) +
(121.278721278))
else: (548.951048934)fi)/360pt#;
```

Replace line 167, which looks like this:

```
else: ((design_size*9.4696969696)+
      236.36363637)fi)/360pt#;
```

with these four lines:

```
elseif design_size < 30:
  ((-0.4995004995*design_size*
   design_size) +
  (25.989010989*design_size) +
  (110.059940059))
else: (440.179820179)fi)/360pt#;
```

At line 173, change these three lines

```
else: ((0.020812520812*design_size*
      design_size) +
  (14.5421245421*design_size) +
  (222.49750249))fi)/360pt#;
```

to the following four lines:

```
elseif design_size < 45:
  ((-0.23934398934*design_size*
   design_size) +
  (20.265567765*design_size) +
  (191.278721278))
else: (618.557692303)fi)/360pt#;
```

Lastly, change five lines starting at line 269 from

```
serif_drop#:=
  % vertical drop of sloped serifs
  (17.28 pt looks strange)
  (if design_size < 12: (design_size*4)
  else: ((design_size*design_size*
        2.62445887445) -
  (design_size * 53.738095238) +
  314.935064935)fi)/360pt#;
```

to

```
serif_drop#:=
  % vertical drop of sloped serifs
  (if design_size < 12: (design_size*4)
  else: ((design_size*design_size*
        0.0228937728937) +
  (design_size * 3.49633699633) +
  2.74725274725)fi)/360pt#;
```

In COMPUTE_CMSS.MF there is only one change:
replace three lines starting at line 47 with two lines.

The original lines look like this:

```
(if design_size < 8:
  ((design_size*235)+10)
elseif design_size < 9:
  ((design_size*470)-1870)
elseif design_size < 10: (2360)
```

and the new lines look like this:

```
(if design_size < 9:
  ((design_size*230)+50)
elseif design_size < 10:
  ((design_size*240)-40)
```

These changes affect only the Computer Modern Roman font at 17 point and the Computer Modern Sans-serif and Slanted Sans-serif fonts at 9 point, so only the CMR17, CMSS9 and CMSSI9 fonts need to be recompiled.

The errata sheet in TUGboat volume 8 number 1 also contained some changes to METAFONT, turning METAFONT version 1.0 into METAFONT version 1.3. At the time I wrote this article I had not yet finished testing all of the Computer Modern fonts with the LN03, but from what I have seen so far METAFONT version 1.3 produces slightly thinner diagonal lines than version 1.0 in some cases, which seems to improve the appearance of the Computer Modern fonts on the LN03.

It is still my intention to put these alternative parameter files on the Stanford tape, along with the pixel and GF files for the LN03. I have recently gotten an encouraging message from David Fuchs, but at the time I wrote this I was still not certain that I would be successful. Therefore, I renew my offer to send anyone running VAX/VMS who has an LN03 and the Stanford tape a VMS Backup copy of everything necessary to print documents using the Computer Modern fonts on the DEC LN03: command files, alternative parameter files, and the resulting .TFM and pixel files for the DEC LN03. If you write me I'll send you a magnetic tape by return mail. If you can't read 6250 BPI tapes be sure to let me know, since that is my default density: it lets me use a smaller tape.

If I succeed in getting these files onto the Stanford tape I'll write another TUGboat article withdrawing this offer; to avoid hassle for the T_EX community I'll fill requests until that article is published.