Macros

A Page Make-up Macro

Joost Zalmstra*
David F. Rogers**

In a previous article in TUGboat (Ref. 1) it was pointed out that using TEX to produce heavily illustrated mathematical, engineering and scientific books containing large numbers of complex display equations and tables poses problems. The reason is that publishers have stringent requirements for the placement of figures with respect to the text while neither Plain TEX nor LATEX provides good control of this placement.

Currently a set of macros is being developed to help in solving this problem. The macros are based on a FIFO queue. Basically, the algorithm is quite simple. When making up a page the FIFO queue is checked. If the queue is not empty and if the first figure on the queue fits on the current page, it is removed from the queue and placed on the current page at the top. If the queue is still not empty, and the second figure on the queue also fits on the page it is removed from the queue and placed at the bottom of the current page. If the figures do not fit on the current page, the page is made-up normally and the process repeated for the next page. If a new figure is encountered, and if the queue is not empty, the figure is placed on the queue. If the queue is empty, and if the figure fits on the current page, it is placed on the current page, either at the top or at the bottom. If the figure does not fit on the current page, it is placed on the queue and the page made-up normally.

In implementing the algorithm in Plain TEX, a form of floating keep with the following properties is introduced:

> The figure and any caption must be placed in a `\vbox`.
>
> The order in which these `\vbox`-es appear in the text is maintained.
>
> No more than two `\vbox`-es appear on one page, one at the top and one at the bottom of the page.

These characteristics satisfy the criteria presented by Rogers in Reference 1. The floating keep is implemented as a macro `\addfigure` that is called with one parameter: the number of a box-register that contains the figure `\vbox`. If the `\vbox` can not appear on the current page (either there are already two figures on this page or there are figures left over from a previous call to `\addfigure`) then `\addfigure` puts the `\vbox` on a list (FIFO queue) implemented as the box-register `\figlist`. Otherwise, it tries to fit the box on the current page, taking into account `\pagetotal`, `\pagegoal` and the size of the figure. The figure size includes the space separating the `\vbox` from the text. If the figure won't fit, it is added to `\figlist`. If the figure fits on the page the `\vbox`, together with the space separating it from the text, is inserted on the current page using Plain TEX's `\topinsert`. The queue (`\figlist`) is implemented using TEX's box operations. An element (figure) is added to the queue by placing it at the top of the existing vertical list of boxes. The 'head' of the queue is accessed via TEX's `\unvbox` and `\lastbox` commands.

The output routine splits the insertion, if necessary, and places the first figure at the top of the page and the second, if present, at the bottom. After the `\shipout` for the current page occurs, the output routine checks `\figlist` to see if there are any figures left. If there are, it attempts to insert a maximum of two figures on the next page, again using `\topinsert`. Before calling `\topinsert` the total size of the figures on the page is checked to insure that it does not exceed `\dimen\topins`. If this is not done, `\topinsert` would shift the insertion to the next page, thus producing a figure on that page without `\addfigure` knowing about it.

The space separating the figure from the text is specified by two skip registers: `\topfigureskip` and `\bottomfigureskip`. The first specifies the `\vskip` below a figure placed at the top of the page, the second the `\vskip` above a bottom figure. Note that this `\vskip` must not be included in the `\vbox` offered to `\addfigure`.

There are several refinements to the basic algorithm that have been implemented and are worth mentioning:

> If the amount of text remaining on a page after insertion of the figure(s) is less than a certain amount (specified by the dimen register `\minpagesize`) then no text is placed on the page. If two figures appear on that page they are placed at the top and bottom of the page.

*Vrije Universiteit, Amsterdam, The Netherlands, `mcvax!nat.vu.nl!jjaz@uunet.uucp`

**United States Naval Academy, Annapolis, MD, USA, `dfr@usna.mil`

If there is only one figure on that page, it appears at the bottom of the page.

The switch \ifbottomforce is used to force a bottom figure. If \bottomforcetrue then if \addfigure finds that it can fit a figure on the current page as the first figure (which would normally be placed at the top of the page) then it forces the figure to the bottom of the page (by inserting an empty \vbox first, which appears at the top of the page). This feature can be used to insure that the figure appears after the first reference to it.

If the \vbox offered to \addfigure has a height of exactly \vsize then it is considered to be a page sized figure. No extra \vskip is added. \pageinsert is used to insert it into the text.

For completeness, a set of macros to create the space for a figure and to set the figure caption is also presented. These macros are called from within the macro \figspace. The \figspace macro has four calling parameters:

The vertical dimension of the white space to be left for the figure.

The horizontal dimension of the white space to be left for the figure.

The figure number.

The figure caption.

If the horizontal size of the figure (parameter 2) is smaller than a certain value specified by \sidearttest then \sidefig is called and the caption is placed to the right of the figure (separated from it by \sideartspace). Otherwise \normalfig is called and the caption appears below the figure separated by \artsurround.

The \addfigure and \figspace macros are combined by calling \figplace with the same parameters as \figspace. This creates space for the figure, sets the caption, places the result in a \vbox and calls \addfigure. Note that \addfigure and \figspace are completely independent. Changing the \figspace-macros has no influence on the working of \addfigure.

The macros have been used both with test files and with production text (Ref. 2). The production text consisted of three separate files. The complexity of the page make-up task is indicated by the details given in Table 1. Page size was 46 pc by 29 pc set 10/12 using the Computer Modern Fonts. The figure space quoted in Table 1 does not include space for the figure captions nor that for \topfigureskip, \bottomfigureskip or \artsurround. The data in Table 1, plus allowance for these spaces, indicates that approximately 25% of the available space is occupied by illustrations. This fact, plus the large numbers of display equations, many of which are 4 to 8 row matrices within an \eqalign, illustrates the complexity of page make-up for these files.

Table 1. Page Make-up Complexity

	File 1	File 2	File 3
No. Pages	40	106	40
No. Figs.	16	50	16
Total Figure Space	357 pc	1255 pc	280.5 pc
Average Figure Space	22.3 pc	25.1 pc	17.5 pc
Smallest Figure	11 pc	6.5 pc	5 pc
No. Display Equations	295	380	250
No. Tables	1	0	10
No. Direct Calls to \normalfig	0	7	2

The \figplace macro performed very well. Calling \figplace immediately after the first reference to the figure often resulted in acceptable placement of the figure. Placement of the figure can be changed by changing the location of the call to \figplace in the source file. Occasionally when TEX was particularly recalcitrant, it was necessary to call \normalfig or \sidefig directly.

There are still some small bugs known to us and some improvements that can be made.

Since the vertical size of a figure caption is not known *a priori*, there is no way to initially exactly specify a full page figure. At least one pass through TEX must be made. The vertical size of the figure caption is then measured and the value of the first parameter in \figplace adjusted accordingly. To solve this problem \figplace is called with a vertical size (parameter 1) of 0. The \figspace macros then change the vertical size to produce a full page figure. Directly incorporating this capability into \figplace to maintain complete independence of the two macros is desirable.

If \raggedbottom is used, the vertical size of pages may vary. However, \figplace always makes full page figures exactly the specified \vsize.

Verso and recto pages may then be unbalanced (of different vertical lengths). The workaround is to call `\figplace` within a `\vbox` of the required size. However, if the required size is greater than the specified `\vsize`, an overfull `\vbox` results and TEX prints the appropriate message. The message can be ignored since the page is set and output correctly.

`\figplace` is limited to a maximum of two figures on a page. However, difficult make-up situations or reader convenience may dictate that more than two figures be placed on the page. For example, it may be desirable to group several small figures together at either the top or bottom of the page. Here, the workaround is to use `\figspace` directly, place the small figures together in a `\vbox` and call `\addfigure` directly, e.g.,

```
\setbox0=\vbox{
     \figspace{#1a}{#2a}{#3a}{#4a}
     \figspace{#1b}{#2b}{#3b}{#4b}
              }
\addfigure0
```

The `\vbox` containing the figures is added to `\figlist` and processed in sequence.

The `\sidefig` macro assumes that the illustration is left justified on the page with the caption set to the right. A companion macro with the illustration right justified on the page with the caption set to the left is needed.

A macro to set two or more small figures side-by-side with the captions below is also needed. Here, the calling sequence might be

```
\sidebysidefig#1#2#3#4#5#6#7#8
```

where items 1–4 correspond to those of `\figplace` for the lefthand figure and 5–8 to those of `\figplace` for the righthand figure.

A major problem is the fact that `\addfigure` must be called between paragraphs. The reason is that `\addfigure` calculates the space available on the current page from `\pagetotal` and `\pagegoal`. However, at the moment that these are evaluated they have the current value at the start of the paragraph.

Despite the problems remaining, the macros seem to do the job. Hopefully they will be useful to other people and will stimulate further discussion of the solution to the page-makeup challenge.

References

1. Rogers, David F., A Page Make-up Challenge, TUGboat, Vol. 9, No. 3, pp 292–293, December 1988.
2. Rogers, David F. and Adams, J. Alan, *Mathematical Elements for Computer Graphics* 2nd Edition, McGraw-Hill, New York, 1989.

The macros figplace.tex

```
% name: figplace.tex
% programmer: Joost Zalmstra
% version/date: Version(1.8)  of 89/16/1 08:30:00
% language: Plain \TeX
% purpose: Implementation of a floating keep that allows
%          better control of the placement of figures
% history: Based on David Rogers tests and comments the original macro
%          was modified as follows:
%          Added separate \topfigureskip and \bottomfigureskip to partially
%          solve the problem of unequal spaces between a topfigure and the
%          text and a bottomfigure and the text.
%          Corrected problem of loss of all subsequent figures if a
%          figure + caption is larger than \vsize.
%          Added code to approximately center the text between a topfigure and
%          a bottomfigure on the same page.
%          Added code to push a figure larger than \vsize - \minpagesize to the
%          bottom of the page.
%          Added code to force a single figure on the page to be at the
%          bottom of the page.
% declaration of some registers
```

```
\newif\ifbottomforce      \bottomforcetrue
\newbox\figlist
\newcount\figuresonpage   \figuresonpage=0
\newcount\figuresonlist   \figuresonlist=0
\newskip\topfigureskip    \topfigureskip=1.25 pc
\newskip\bottomfigureskip \bottomfigureskip=1.5 pc
\newdimen\pagesize
\newdimen\minpagesize     \minpagesize=5 pc
\newdimen\figuresizeonpage
\input figspace.tex%        % \figspace is the macro that actually sets the white space
%                           % for the art and the figure caption.
%                           % It is independent of \figplace.
% user interface for inclusion of figures
\catcode`@=11 %             %  borrowing macros from plain TeX
% Note: the redefinition of \raggedbottom has nothing to do with
%       the implementation of \addfigure. It is only placed here
%       because \raggedbottom uses @ as a letter
\def\raggedbottom{\topskip10pt plus 24pt \r@ggedbottomtrue}
\def\figplace#1#2#3#4{%
  \setbox0=\vbox{%
    \figspace{#1}{#2}{#3}{#4}
  }
  \addfigure0
}
% add figure in box #1 to the document
\def\addfigure#1{%
    \dimen0=\ht#1
    \ifdim\dimen0>\vsize%            % be sure it will fit
      \message{figure to big}
      \setbox#1=\vbox to \vsize{\box#1}
    \fi
    \ifnum\figuresonpage>1%          % already two figures on this page
      \addtolist{#1}%                % so save for later
    \else\ifnum\figuresonlist>0%     % figlist is not empty
      \addtolist{#1}%                % so save for later
    \else%                           % no figures left over, so try to fit this one
      \dimen0=\ht#1
      \ifdim\dimen0<\vsize%          % not a pagesize figure
          \advance\dimen0 by \dp#1%  % size of the box
          \ifnum\figuresonpage=0
            \advance\dimen0 by \topfigureskip%      % extra space below
          \else
            \advance\dimen0 by \bottomfigureskip%   % extra space above
          \fi
      \fi
      \dimen1=\dimen0
      \advance\dimen0 by \pagetotal
      \ifdim\dimen0>\pagegoal%            % no space on this page
        \addtolist{#1}%                   % so save for later
      \else
        \advance\dimen1 by \figuresizeonpage
        \ifdim\dimen1>\dimen\topins%      % exceeding maximum insertion size
            \addtolist{#1}%               % so save for later
        \else%                            % insert figure
          \ifnum\figuresonpage=0%         % force a bottom insert
            \ifbottomforce%               % by adding an empty box first
              {\setbox1=\vbox{%
                 \vskip -\topfigureskip}% compensate for figureskip
              \figureinsert1}
            \fi
          \fi
          \figureinsert{#1}%              % insert the figure
        \fi
      \fi
    \fi\fi
}
% \addtolist adds \box#1 to the FIFO queue \figlist
\def\addtolist#1{%
    \global\setbox\figlist=\vbox{%
        \nointerlineskip
        \box#1
        \unvbox\figlist}
    \global\advance\figuresonlist by 1
}
% \figureinsert inserts \box#1 using Plain \TeX's \topinsert
% and updates the registers \figuresizeonpage and \figuresonpage
\def\figureinsert#1{%
    \dimen0=\ht#1
    \advance\dimen0 by \dp#1
    \ifdim\ht#1<\vsize%                           % not a full page figure
        \ifnum\figuresonpage=0
           \advance\dimen0 by \topfigureskip
           \insert\topins{\vbox{%
             \unvbox#1\vskip\topfigureskip}}%     % skip below the figure
        \else
           \advance\dimen0 by \bottomfigureskip
           \insert\topins{\vbox{%
             \vskip\bottomfigureskip\unvbox#1}}%  % skip above the figure
         \fi
    \else%                                        % full page figure
        \pageinsert
            \unvbox#1
        \endinsert
    \fi
    \global\advance\figuresonpage by 1
    \global\advance\figuresizeonpage by \ht#1
    \global\advance\figuresizeonpage by \dp#1
    \ifnum\figuresonpage=0
       \global\advance\figuresizeonpage by \topfigureskip
    \else
       \global\advance\figuresizeonpage by \bottomfigureskip
    \fi
}
% see if there is a figure in the list that will fit on the next page
% return \box0 containing the figure or void if no figure will fit
\def\checkfiglist{%
    \ifnum\figuresonlist>0 %                 % there are figures on the queue
```

```
        \global\setbox\figlist=\vbox{%
            \unvbox\figlist
            \setbox0=\lastbox%             % \box0 contains first figure
            \global\advance\dimen0\ht0
            \ifdim\ht0<\vsize
                \global\advance\dimen0 by \dp0
                \ifnum\figuresonpage=0
%                                          % topfigure may loose \figureskip
                \else
                   \global\advance\dimen0 by \bottomfigureskip
                \fi
            \fi
            \ifdim\dimen0>\vsize%          % test if it will fit
                 \vbox{\box0}%             % no room ; replace the figure
            \else
                \ifdim\dimen0>\dimen\topins
                    \vbox{\box0}%          % exceeding maximum insertion size
                \else%                     % remove it from the queue
                    \global\advance\figuresonlist by -1
                \fi
            \fi
            \global\setbox0=\box0
        }
    \else%                                  % no figures left on the queue
        {\global\setbox0=\box0\voidb@x}     % make box0 void
    \fi
}
% output routines
% note: these are just modifications of the Plain \TeX output routines
\output{\myoutput}
\def\myoutput{\shipout\vbox{\makeheadline\pagebody\makefootline}%
  \advancepageno
% reset \addfigure registers
  \global\figuresonpage=0
  \global\figuresizeonpage = 0pt
  \dimen0=0pt%                      % \dimen0 contains \figuresizeonpage
  \checkfiglist%                    % check if there is figure waiting that will fit
  \ifvoid0%                         % nothing left
  \else
      \figureinsert0%               % insert this figure
      \global\advance\dimen0 by \topfigureskip
      \checkfiglist%                % check for a second figure
      \ifvoid0\else
          \figureinsert0
       \fi
  \fi
  \ifvoid255\else\unvbox255\penalty\outputpenalty\fi%   % put unused page back on list
  \ifnum\outputpenalty>-\@MM \else\dosupereject\fi}
% actual contents of this page
\def\pagecontents{%
    \pagesize=\ht255
    \advance\pagesize by \dp255%          % determine size of box 255
    \ifnum\figuresonpage>0%               % check for insertion of figures
        \unvbox\topins%                   % place the figure(s) on top
        \ifdim\pagesize>\minpagesize
           \ifnum\figuresonpage=2
               \setbox0=\lastbox%                % place the second figure in \box0
           \fi
           \setbox1=\lastbox
           \unvbox1%                             % remove surrounding box
        \else%                                   % no text is produced on this page
          \ifnum\figuresonpage=1
             \setbox0=\lastbox
             \vfill%                             % figure becomes bottomfigure
             \unvbox0%                           % remove surrounding box
             \unskip%                            % remove \topfigureskip
          \fi
        \fi
    \fi
% topfigure is placed on this page
% put the text on this page if it is large enough
     \ifdim\pagesize>\minpagesize
        \dimen@=\dp255 \unvbox255%       % the real page contents
        \ifr@ggedbottom \kern-\dimen@\vfil\fi
     \else
        \ifdim\ht255=0pt%                         % box is empty
        \else
           \vfil%                                 % too small so leave for next page
        \fi
     \fi
% Produce the second figure, if present
    \ifnum\figuresonpage=2
        \box0%                                    % second figure
    \fi
% produce possible footnotes
    \ifvoid\footins\else%                         % footnote is present
      \vskip\skip\footins
      \footnoterule
      \unvbox\footins\fi
}
\catcode`@=12%                                    % at signs no longer letters
```

The macros figspace.tex

```
% name: figspace.tex
% programmer: David F. Rogers
% version/date: Version(1.6) of 89-16-01 08:30:00
% language: Plain \TeX
% purpose: To provide white space for stripped in art and set the caption for the art
% history: Modified by Joost Zalmstra to work more efficiently with the \figplace macro
%          Added code to check for nearly page size figure, expand it to fill the page
%          and print a message to that effect.
%          Added code to check for figure + caption larger than page size, to
%          reduce the figure size (parameter #1) so that the art plus caption just
%          fills the page and to print a message to that effect.
% define top and bottom cornerrules to indicate the size of the art
\newdimen\ruleht     \ruleht=0.5pt%        % take the rules out for final
\newdimen\ruledpth   \ruledpth=0pt%        %  production by setting
\newdimen\rulelength \rulelength=1pc%      % \ruleht=0pt and \rulelength=0pt
\def\settopcornerrules#1{\hbox to #1{\vrule width\rulelength height\ruleht
        \hfil \vrule width \rulelength}
    \hbox to #1{\vrule height\rulelength width\ruleht depth\ruledpth
        \hfil \vrule width\ruleht depth\ruledpth}}
\def\setbotcornerrules#1{%
        \vskip  -2\rulelength%
        \hbox to #1{\vrule height\rulelength width\ruleht depth\ruledpth%
        \hfil \vrule width\ruleht depth\ruledpth}%
    \vskip -\rulelength%
    \hbox to #1{\vrule width \rulelength height\ruleht%
        \hfil \vrule width \rulelength}
}
% define a figure caption macro
% #1 is the figure number #2 is the caption
% the caption is to be set in a `box' left and right justified 1em to
% the right of the figure number
% the size to the box containing the word Figure, its number and the 1em
% skip is found in box0
% box1 is \hsize less the width of box0.
% a \vtop is used along with an \halign to obtain the flush left and right
% effect.
\def\figcap#1#2{{%
    \setbox0=\hbox{{\bf Figure #1}\hskip 1em}%
    \setbox1=\vtop{\advance \hsize by -\wd0 \noindent
                   \spaceskip=.3em plus .2em minus.2em
                   #2}
    \halign{## & ## \cr
            \box0 & \box1 \cr}}
    \leavevmode
}
% define a figspace macro
% A 1.5pc space is required above the art.
% If a figure is more than 13pc wide then the caption is to be set
% below the art with a 1.5pc space between the art and the figcap.
\newdimen\artsurround  \artsurround=1.5pc%   % space between art and the caption
\newdimen\sidearttest  \sidearttest=13pc%    % if more than set side caption
\newdimen\sideartspace \sideartspace=2em%    % horizontal space to caption
\def\figspace#1#2#3#4{%
    \ifdim #2 > \sidearttest%
        \normalfig{#1}{#2}{#3}{#4}
    \else
       \sidefig{#1}{#2}{#3}{#4}
    \fi
}
% this is the case when the figure is wider than \sidearttest (13 pc's)
\def\normalfig#1#2#3#4{%   #1 is vertical size in pc's
%                          #2 is horizontal size in pc's
%                          #3 is the figure number -- used in figcap
%                          #4 is the figure caption -- used in figcap
```

```
    \setbox0=\vbox{\figcap{#3}{#4}}
    \dimen0=\ht0
    \advance\dimen0 by \dp0
    \advance\dimen0 by \artsurround
    \dimen1=#1
    \ifdim\dimen1 = 0 pt%            % figure should fill the rest of the page (jz)
       \dimen1 = \vsize
       \advance\dimen1 by -\dimen0
       \message{expanding figure to \the\dimen1}
    \else
       \advance\dimen0 by \dimen1
       \ifdim\dimen0>\vsize%         % figure + caption larger than page size (jz)
            \advance\dimen0 by -\vsize
            \advance\dimen1 by -\dimen0
            \message{warning: reducing vertical figure size to \the\dimen1}
       \fi
    \fi
    \leftline{%                       % left justify the figure
       \vbox to \dimen1{%
           \settopcornerrules{#2}%   % set top corner rules
            \vfill%                   % push the figcap down to bottom
            \setbotcornerrules{#2}%  % set bottom corner rules
        }%
    }%
    \nointerlineskip
    \vskip \artsurround%              % put space between figure & caption
    \box0%                            % set the caption
}
% this is the case when the caption is to be set to the right
\def\sidefig#1#2#3#4{%     #1 is vertical size in pc's
%                          #2 is horizontal size in pc's
%                          #3 is the figure number -- used in figcap
%                          #4 is the figure caption -- used in figcap
    \ifdim#1>\vsize%      % figure + caption larger than page size (jz)
       \dimen0=\vsize
       \message{warning: reducing vertical figure size to \the\dimen0}
   \else
       \dimen0=#1
   \fi
   \setbox0=\vbox to \dimen0{%
       \advance \hsize by -#2%                % reduce \hsize by the horiz.
       \advance \hsize by -\sideartspace%     % size and blank space between
       \vfill%                                % push the figcap down to bottom
       {\figcap{#3}{#4}}%                     % of the box
    }
    \setbox1=\hbox to #2{\vbox to \ht0{
       \settopcornerrules{#2}%                % set top corner rules
       \vfill%                                % push the figcap down to bottom
       \setbotcornerrules{#2}%                % set bottom corner rules
    }}%
    \hbox{\box1 \hskip \sideartspace \box0}%   actually do the setting
}
```