The tonevalue package

Yuanhao Chen

July 25, 2021 v1.0

Contents

1 Introduction

This package provides a tikz-based solution to typeset visualisations of tone vales. In this version (v1.0), unt's model[1] is implemented. Support for more models is planned.

2 User Interface

2.1 Basic Usage

Put in your preamble

```
\usepackage[⟨tonevalue options⟩]{tonevalue}
```

then after \begin{document}, use

```
\begin{⟨name of visualisation environment⟩}[⟨visualisation environment options⟩]
  \⟨name of drawing command⟩[⟨drawing options⟩]{⟨tone value⟩}{⟨name of tone⟩}
\end{⟨name of visualisation environment⟩}
```

[1] unt. 一种直观的调值格局可视化方法 (A Novel Approach to Visualization of Tone Value Pattern). 第十四届中国语音学学术会议 (The 14th Phonetic Association of China). July 2021.

2.2 A Brief Working Example

An example of complete working code looks like

Listing 1: basic example.

```
\documentclass{article}

% load the package, and use the predefined color set
\usepackage[defaultcolors]{tonevalue}

\begin{document}

% set showlabels to true
% set range of tone values to 1 to 4
% set scale of graph to 0.8
\begin{untVisualisation}[showlabels=true, minmax={1,4}, scale=0.8]
  % T1
  \untpoint[bgcolor=1, label=left]{312}{T1}
  \untpoint[bgcolor=1]{33}{T1}
  % change in tone value
  \linkuntpoints[color=1, bend=bend right]{{312}{T1}}{{33}{T1}}
\end{untVisualisation}

\end{document}
```

with the result

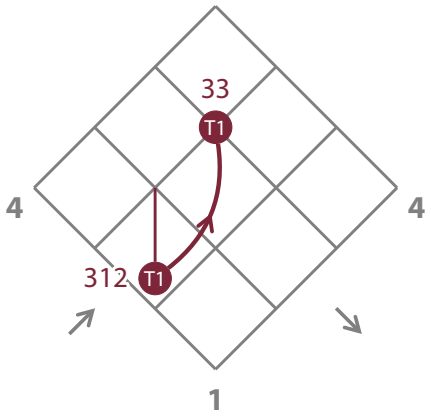

Figure 1

2.3 Details

2.3.1 Package Options

The package options can be called like in

```
\usepackage[defaultcolors, draft, fontcmd=⟨font commands⟩,
  contourlength=⟨length⟩, contournumber=⟨integer⟩]{tonevalue}
```

The effects of the options are listed below.

`defaultcolors`=no value required[2]

Use the pre-defined colour scheme designed for the four-tone and the eight-tone systems (四聲八聲) and their simplifications. The colours are chosen such that the representative character taken from each of their names falls into the category of the tone it describes, and such that the *yin* tone is of the same colour tone as but darker than its corresponding *yang* tone in the eight-tone system.

The colours are programmatically named 1 to 8, defined as `xcolor` HTML colours.

1 蘇/蘇芳色　3 朽/朽葉色　5 熨/熨斗目花色　7 竹/老竹色
2 梅/紅梅色　4 柿/柿　色　6 露/露　草　色　8 鶸/鶸萌黄

The names are taken with reference to *A Dictionary of Color Combinations*[3].

`draft`=no value required

This will speed up compilations by `\contournumber{50}` defined by `contour`.

`fontcmd`=⟨*font commands*⟩ default: `\sffamily`

This sets the font commands to use in all graphs.

`contourlength`=⟨*length*⟩ default: `0.075em`

This sets the width of contours around labels of tones to allow them stand out in the grid.

`contournumber`=⟨*integer*⟩ default: `1000`

Increase this to improve contour quality; decrease to compile faster.

2.3.2 The `untVisualisation` environment

Use this environment to draw the axes and, optionally, labels of unt's model. Later, put the drawing commands of points and lines inside this environment.

[2] 'No value required' means it could be called on its own, i.e. `defaultcolors`, or with an arbitrary string passed to it, i.e. `defaultcolors={any string}` without affecting the result.
[3] 青幻舍 (Seigensha). 配色事典 (*A Dictionary of Color Combinations*).

```
\begin{untVisualisation}[minmax=⟨range of tone values⟩,
                         scale=⟨float⟩, showlabels=⟨boolean⟩]
  \untpoint[⟨\untpoint options⟩]{⟨tone value, e.g. 3124⟩}{⟨tone name, e.g. 上⟩}
\linkuntpoints[⟨\linkuntpoints options⟩]{{⟨tone value⟩}{⟨tone name⟩}}{{⟨tone value⟩}{⟨tone name⟩}}
  \end{untVisualisation}
```

A default empty `untVisualisation` environment looks like fig. 2. A modified `untVisualisation` environment looks like fig. 1.

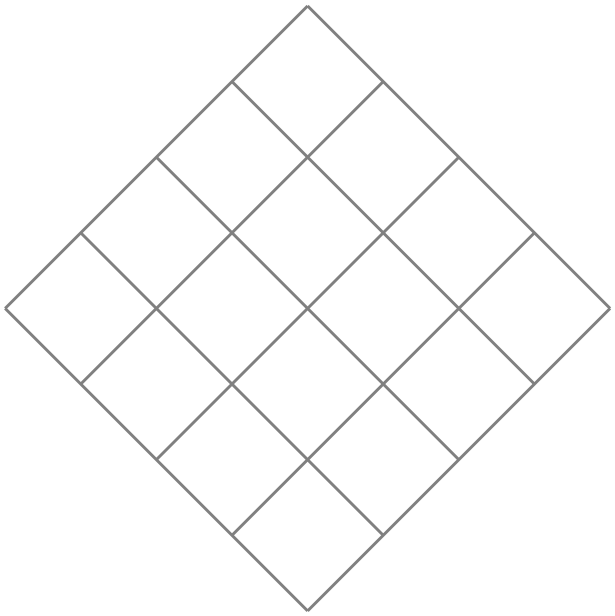

Figure 2: empty `untVisualisation`.

`minmax=`⟨*range of tone values*⟩ default: `{1,5}`

Sometimes we deal with languages whose tone values do not range from 1 to 5. Use this command to modify the minima and maxima of the axes.

`scale=`⟨*float*⟩ default: `1`

Scales the grid, but not the font size, as in fig. 3.

`showlabels=`⟨*boolean*⟩ default: `false`

Controls whether to display the labels, as in fig. 3.

2.3.3 The `\untpoint` Command

Use inside the `untVisualisation` environment to plot tone values.

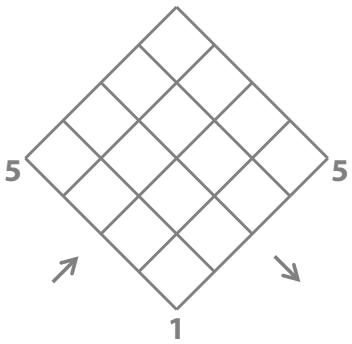

Figure 3: `untVisualisation` with labels, and scaled to factor `0.5`.

`\untpoint[`⟨*list of options*⟩`]{`⟨*tone value, e.g. 3124*⟩`}{`⟨*tone name, e.g.* 上⟩`}`

Below is a complete list of `\untpoint` options.

`stem=`⟨*boolean*⟩ default: `false`

Use `stem=true` to add a stem for turning tones.

`label=`⟨*combinations of* `above, below, left, right`⟩ default: `above`

For instance, use `label=below left` to put the label (tone value) below left of the point.

`bgcolor=`⟨*color*⟩ default: `black`

For instance, with the package option `defaultcolors` on, use `bgcolor=4` to colour the point in the *yangshang* colour.

`xshift=`⟨*length*⟩ default: `0pt`

When there are two different points at the same coordinates, use this option to slightly shift the points horizontally, e.g. `xshift=0.8em`.

`yshift=`⟨*length*⟩ default: `0pt`

The vertical variant of `xshift`.

`scale=`⟨*float*⟩ default: `1`

Scales the size of the point.

`tikzoptions=`⟨*tikz options not in the key-value format*⟩ *Unstable (this might clash with the options required to plot the point). Use at risk.* default: `{}`

For instance, use `tikzoptions={black}` to make the point completely black (the name of the tone becomes invisible), but preserving the size of the point which fits to the invisible name of the tone.

2.3.4 The `\linkuntpoints` Command

It must be called after the points involved are drawn.

`\linkuntpoints`[⟨*list of options*⟩]{{tone value 1}{tone name 1}}{{tone value 2}{tone name 2}}

Below is a complete list of `\linkuntpoints` options.

`color`=⟨*color*⟩ default: `black`

Colours the connecting line.

`bend`=⟨*bend direction*⟩ default: {}

Set `bend=bend left` or `bend=bend right` to bend the line.

2.4 A More Complicated Example

Shifts in the tone value pattern of Shanghainese in the past 150 years (fig. 4)[4], drawn with

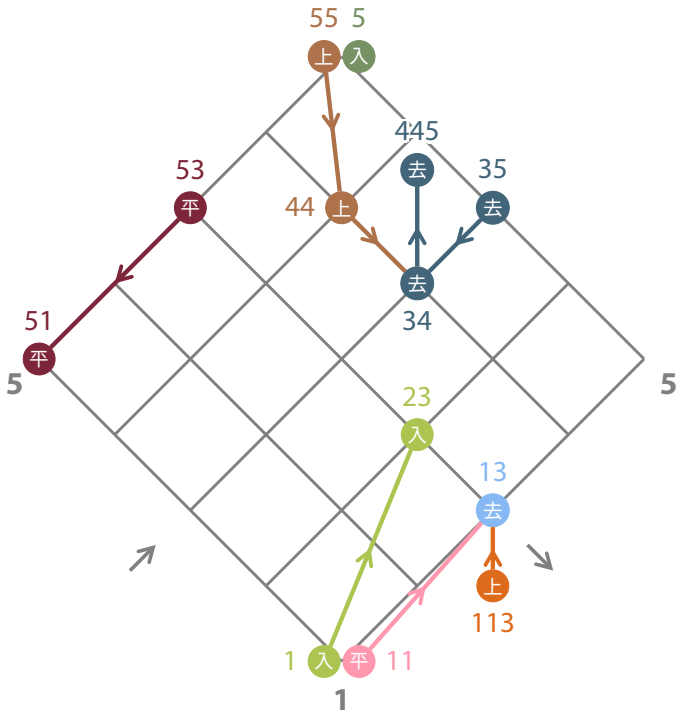

Figure 4: shifts in the tone value pattern of Shanghainese in the past 150 years.

the following code, compiled with XeLaTeX.

Listing 2: example regarding Shanghainese.

```
% !TEX program = xelatex
```

[4]unt. Ibid.

```
\documentclass{ctexart}

\usepackage[defaultcolors]{tonevalue}

\begin{document}

\begin{untVisualisation}[showlabels=true]
  % 1
  \untpoint[bgcolor=1]{53}{平}
  \untpoint[bgcolor=1]{51}{平}
  % 2
  \untpoint[bgcolor=2, label=right, xshift=0.8em]{11}{平}
  % 3
  \untpoint[bgcolor=3, xshift=-0.8em]{55}{上}
  \untpoint[bgcolor=3, label=left]{44}{上}
  % 4
  \untpoint[bgcolor=4, label=below]{113}{上}
  % 5
  \untpoint[bgcolor=5]{35}{去}
  \untpoint[bgcolor=5, label=below]{34}{去}
  \untpoint[bgcolor=5]{445}{去}
  % 6
  \untpoint[bgcolor=6]{13}{去}
  % 7
  \untpoint[bgcolor=7, xshift=0.8em]{5}{入}
  % 8
  \untpoint[bgcolor=8, label=left, xshift=-0.8em]{1}{入}
  \untpoint[bgcolor=8]{23}{入}
  %
  \linkuntpoints[color=1]{{53}{平}}{{51}{平}}
  \linkuntpoints[color=3]{{55}{上}}{{44}{上}}
  \linkuntpoints[color=3]{{44}{上}}{{34}{去}}
  \linkuntpoints[color=4]{{113}{上}}{{13}{去}}
  \linkuntpoints[color=5]{{35}{去}}{{34}{去}}
  \linkuntpoints[color=5]{{34}{去}}{{445}{去}}
  \linkuntpoints[color=2]{{11}{平}}{{13}{去}}
  \linkuntpoints[color=8]{{1}{入}}{{23}{入}}
  %
\end{untVisualisation}

\end{document}
```

3 Implementation

Listing 3: the implementation.

```
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{tonevalue}
  [2021/07/25 v1.0 LaTeX Package (Tone value: visualising tone value patterns)]

\RequirePackage{etoolbox}
\RequirePackage{listofitems}
\RequirePackage{xstring}
\RequirePackage{xkeyval}
\RequirePackage{xcolor}
\RequirePackage{tikz}
  \usetikzlibrary{positioning,decorations.markings,arrows}
\RequirePackage{contour}

% package options
  % font command
  \DeclareOptionX{fontcmd}[\sffamily]{\def\toneVisualisationFontCmd{#1}}
  % colors
  \DeclareOptionX{defaultcolors}{
    \definecolor{1}{HTML}{7E2639}
    \definecolor{2}{HTML}{FF98AF}
    \definecolor{3}{HTML}{AD724A}
    \definecolor{4}{HTML}{DE6A1C}
    \definecolor{5}{HTML}{426579}
    \definecolor{6}{HTML}{86B8F3}
    \definecolor{7}{HTML}{769164}
    \definecolor{8}{HTML}{ACC551}
  }
  % contour around numbers
  \DeclareOptionX{draft}{\contournumber{50}}
  \DeclareOptionX{contourlength}[0.075em]{\contourlength{#1}}
  \DeclareOptionX{contournumber}[1000]{\contournumber{#1}}
\ExecuteOptionsX{
  contourlength=0.075em,
  contournumber=1000,
  fontcmd=\sffamily,
}
\ProcessOptionsX*\relax
\ProcessOptionsX\relax
```

```
\def\xjoinbycomma<#1#2>{%
  \ifx\relax#1
  \else
    #1,\xjoinbycomma<#2>%
  \fi
}
\def\sendiauToListStr#1{\xjoinbycomma<#1\relax>}

% environment for unt's visualisation approach
\define@key{untVisualisation}{minmax}{\def\untVisualisation@minmax{#1}}
\define@key{untVisualisation}{scale}{\def\untVisualisation@scale{#1}}
\define@key{untVisualisation}{showlabels}{\def\untVisualisation@showlabels{#1}}
\newenvironment{untVisualisation}[1][]{
  \setkeys{untVisualisation}{minmax={1,5}, scale=1, showlabels=false}
  \setkeys{untVisualisation}{#1}
  \toneVisualisationFontCmd
  % mxn minmax
  \pgfmathparse{{\untVisualisation@minmax}[0]}
  \edef\xstart{\pgfmathresult}
  \pgfmathparse{{\untVisualisation@minmax}[1]}
  \edef\xend{\pgfmathresult}
  \pgfmathparse{{\untVisualisation@minmax}[0]}
  \edef\ystart{\pgfmathresult}
  \pgfmathparse{{\untVisualisation@minmax}[1]}
  \edef\yend{\pgfmathresult}
  \tikzpicture[scale=\untVisualisation@scale]
    \begin{scope}[rotate=45, scale=1.4142, line width=0.1em, gray]
      \foreach \x in {\xstart,...,\xend}
        \draw (\x,\ystart) -- (\x,\yend);
      \foreach \y in {\ystart,...,\yend}
        \draw (\xstart,\y) -- (\xend,\y);
    \end{scope}
    \begin{scope}[gray]
      \expandafter\ifstrequal\expandafter{\untVisualisation@showlabels}{true}{
        \node at (0,\ystart*2-0.5) {\large\bfseries\xstart};
        \node at (-\xend+0.6767+\xstart-1,\yend+0.6767+\ystart-1) {\large\
            ↪ bfseries\xend};
        \node at (\xend-0.6767-\xstart+1,\yend+0.6767+\ystart-1) {\large\
            ↪ bfseries\xend};
        \node (xAxisArrowTip) at (-\xend+0.6767+\xstart-1+\xend/2-\xstart/2,\
            ↪ yend+0.6767+\ystart-1-\yend/2+\ystart/2) {};
```

```
        \node[below left=1.25em of xAxisArrowTip] (xAxisArrowTail) {};
        \draw[line width=0.125em, -angle 60] (xAxisArrowTail) -- (xAxisArrowTip)
            ↪ ;
        \node (yAxisArrowTail) at (\xend-0.6767-\xstart+1-\xend/2+\xstart/2,\
            ↪ yend+0.6767+\ystart-1-\yend/2+\ystart/2) {};
        \node[below right=1.25em of yAxisArrowTail] (yAxisArrowTip) {};
        \draw[line width=0.125em, -angle 60] (yAxisArrowTail) -- (yAxisArrowTip)
            ↪ ;
      }{}
    \end{scope}
}{
  \endtikzpicture
}

% draw a point
\newcommand{\drawuntpoint}[9][]{
  % override tikz options, background color, coordinates, tone name, tone value
      ↪ in numbers, label position
    \node[draw, shape=circle, scale=#9*0.75, inner sep=0.1em, fill, #2, text=
        ↪ white, #1, xshift=#7, yshift=#8]
    ({#5}{#4})
    at ({#3}[0],{#3}[1])
    {#4};
  \node[#2, #6 = 0pt of {#5}{#4}] {\contour{white}{#5}};
}

\newcounter{sumOfPitchHeights}
% keys of options
\define@key{untpoint}{label}{\def\untpoint@label{#1}}
\define@key{untpoint}{tikzoptions}{\def\untpoint@tikzoptions{#1}}
\define@key{untpoint}{bgcolor}{\def\untpoint@bgcolor{#1}}
\define@key{untpoint}{xshift}{\def\untpoint@xshift{#1}}
\define@key{untpoint}{yshift}{\def\untpoint@yshift{#1}}
\define@key{untpoint}{scale}{\def\untpoint@scale{#1}}
\define@key{untpoint}{stem}{\def\untpoint@stem{#1}}
% drawing interface
\newcommand{\untpoint}[3][]{
  % options, tone value in numbers, tone name
  \setkeys{untpoint}{label=above, tikzoptions={}, bgcolor=black, xshift=0pt,
      ↪ yshift=0pt, scale=1, stem=false}
  \setkeys{untpoint}{#1}
```

```
  \StrGobbleRight{\sendiauToListStr{#2}}{2}[\sendiaulistStr] % readlist cannot
      ↪ parse trailing comma
  \readlist\sendiaulist{\sendiaulistStr}
  \edef\len{\listlen\sendiaulist[]}

  \ifnum0\len=1
    \drawuntpoint[{\untpoint@tikzoptions}]{\untpoint@bgcolor
        ↪ }{0,{#2}[0]*2}{#3}{#2}
      {\untpoint@label}{\untpoint@xshift}{\untpoint@yshift}{\untpoint@scale}
  \else
    \setcounter{sumOfPitchHeights}{0}
    \pgfmathparse{\len-2}
    % calculate the sum of pitch heights
    \foreach \pitchHeightIndex in {0,...,{\pgfmathresult}} {
      \pgfmathparse{{#2}[\pitchHeightIndex]}
      \addtocounter{sumOfPitchHeights}{\pgfmathresult}
      \pgfmathparse{{#2}[\pitchHeightIndex+1]}
      \addtocounter{sumOfPitchHeights}{\pgfmathresult}
    }
    % draw the point
    \drawuntpoint[\untpoint@tikzoptions]
      {\untpoint@bgcolor}
      {{-({#2}[0])+{#2}[\len-1]},{\thesumOfPitchHeights/(\len-1)}}
      {#3}{#2}
      {\untpoint@label}{\untpoint@xshift}{\untpoint@yshift}{\untpoint@scale}
    % draw the stem
    \expandafter\ifstrequal\expandafter{\untpoint@stem}{true}{
      \draw[\untpoint@bgcolor, line width=0.1em] ({-({#2}[0])+{#2}[\len
          ↪ -1]},{#2}[0]+{#2}[\len-1]) -- ({#2}{#3});
    }{}
  \fi
}

% link points
\define@key{linkuntpoints}{color}{\def\linkuntpoints@color{#1}}
\define@key{linkuntpoints}{bend}{\def\linkuntpoints@bend{#1}}
\newcommand{\linkuntpoints}[3][]{
  \setkeys{linkuntpoints}{color=black, bend={}}
  \setkeys{linkuntpoints}{#1}

  \begin{scope}[
      decoration={
```

```
        markings,
        mark=at position 0.5 with {\arrow[scale=0.875]{angle 60}}}
    ]
    \draw[postaction={decorate}, line width=0.15em, \linkuntpoints@color] (#2)
        ↪ to [\linkuntpoints@bend] (#3);
  \end{scope}
}

\endinput
```