The `vowelchart` package

Vowel charts with TikZ

Daniel Bunčić
University of Cologne
daniel.buncic@uni-koeln.de
https://github.com/danielbuncic/vowelchart

Version 1.0 (21 November 2026)

Contents

1 Introduction **2**
1.1 Features . . . 2
1.2 Requirements . . . 2
2 Quick start: The general structure **3**
3 Options: Properties of the vowel chart **3**
3.1 Where options can be set . . . 3
3.2 Shape . . . 4
3.3 Grid style . . . 6
3.4 Size . . . 6
3.5 Colors . . . 7
3.6 Font and font size . . . 8
3.7 Line width and dot size . . . 8
4 The commands: Placing vowels **9**
4.1 Placement options . . . 9
4.2 `\putvowel` . . . 9
4.3 `\putcvowel` . . . 10
4.4 `\putipavowel` . . . 11
4.5 `\putdiphthong` . . . 11
4.6 `\putipadiphthong` . . . 13
5 Using TikZ in the vowel chart **14**
5.1 Nodes . . . 14
5.2 The coordinate system `vowel cs` . . . 14
5.3 Using the chart's settings in your own code . . . 15
5.4 Further possibilities . . . 16
5.5 Changing the white areas for colored backgrounds . . . 16
6 Compatibility with `vowel` and `pst-vowel` **17**
6.1 Moving from `(pst-)vowel` to `vowelchart` . . . 17
6.2 Legacy: absolute coordinates with `\putvowel*` . . . 18
7 A known problem and some solutions **19**
8 The mathematics behind it **21**
8.1 How `vowelchart` turns relative into real coordinates . . . 21
8.2 From absolute `vowel` coordinates to relative `vowelchart` coordinates . . . 22
8.3 Converting many coordinates in a spreadsheet . . . 23
9 Examples **24**
Appendix: Short reference of commands and options **28**

1 Introduction

1.1 Features

The package `vowelchart`, as the name suggests, helps you draw vowel charts for linguistics. It is indebted to the creators of two older packages, whose work it continues:

- `vowel` (2001) by Fukui Rei, distributed with `tipa`, draws the charts with LaTeX-picture-like primitives. Vowels are placed with `\putvowel` and `\putcvowel`.
- `pst-vowel` (2009) by Alan Munn adds PSTricks nodes to `vowel`, so that arrows (e.g. for diphthongs) can be drawn between vowels.

However, the package `vowelchart` differs from the preceding packages in several important respects:

- **Vector graphics:** In `vowel`, the slanted lines consist of many tiny squares, which show up as steps at higher resolutions. In `vowelchart`, every line is a real vector path and stays smooth at any zoom level.
- **Nodes and TikZ:** Every vowel is a TikZ node. You can draw arrows, shaded areas and annotations with ordinary TikZ commands in the same picture, using a special coordinate system that works for every chart shape.
- **Exact size:** The chart can be given an exact width or height.
- **More shapes and grids:** In addition to the shape options of `vowel`, there are a symmetric and a slanted quadrilateral, and all four corners can be set by hand. Apart from the grid styles of `vowel`, you can have a triangular central zone without a middle line or a central column resulting in a 3×3 division.
- **Relative coordinates:** Vowels are placed with abstract coordinates that follow the shape of the chart, so the same code works in every shape.
- **Convenient commands:** `\putipavowel` places a vowel simply defined by its IPA symbol (with Unicode or X-SAMPA input) at its usual position in the chart, `\putdiphthong` draws a diphthong arrow between two positions, and `\putipadiphthong` draws it between two vowels given by their IPA symbols or between any mixture of IPA vowels and TikZ nodes.
- **Colors and line widths:** Foreground and background colors, the line width and the dot size can be set for each chart.
- **Settings at several levels:** All options can be given for a single chart, for all following charts in a document, or as package options (Section 3.1).

1.2 Requirements

The package loads `tikz`, `contour` and `xparse`, and needs a LaTeX kernel from 2020 or later (expl3). With XeLaTeX or LuaLaTeX, choose a font that contains the IPA symbols (for instance Libertinus, Charis SIL or DejaVu). With pdfLaTeX, the IPA characters must be made available by an encoding or font package.

If you load `vowelchart` together with `vowel` or `pst-vowel`, load it *after* them (see Section 6.2).

2 Quick start: The general structure

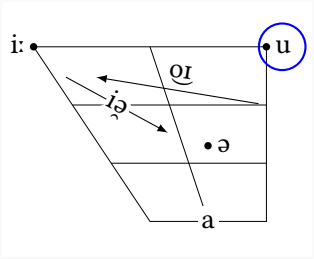
In order to draw vowel charts, you first have to tell LaTeX to use this package:

```
\usepackage{vowelchart}
```

If the package `vowelchart` is not part of your LaTeX distribution, all you have to do is copy the file `vowelchart.sty` to the same folder as your main `.tex` file.

Then you can create an environment `vowelchart`, which automatically draws a chart. Within this environment, you can then use any of the commands `\putvowel`, `\putcvowel`, `\putipavowel`, `\putdiphthong`, and `\putipadiphthong` to place vowels on the chart. You can also use TikZ commands to draw in the chart. See the following example.

```
\begin{vowelchart}
  \putvowel[r]{ə}{1.3}{1.7}
  \putcvowel[l]{i:}{1}
  \putipavowel{u}[vowelu]
  \putipavowel{'a'}
  \putdiphthong[diphlabelpos=on]{iə̯}{0.1}{0.5}{0.9}{1.5}
  \putipadiphthong{*}{o}{I}
  \draw[blue, thick] (vowelu) circle (0.4);
\end{vowelchart}
```



You can change the appearance of the vowel chart by adding parameters. The options you have for this, and the places where you can set them, are laid out in Section 3. The exact syntax and properties of the commands for placing vowels are presented in Section 4. Finally, since the vowel chart is a `tikzpicture`, you can use TikZ commands to easily add graphical elements (and text) to the chart, see Section 5.

3 Options: Properties of the vowel chart

3.1 Where options can be set

All options are given as a comma-separated list of keys, optionally with values (`key=value`). Such a list can be used at three levels:

```
% 1. for a single chart
\begin{vowelchart}[<options>]
  <vowel commands and TikZ commands>
\end{vowelchart}

% 2. for all following charts
\vowelchartset{<options>}

% 3. for all charts of the document, when loading the package
\usepackage[<options>]{vowelchart}
```

At the beginning of every chart, first the options given when loading the package and by `\vowelchartset` are applied, in the order in which they were given, and then the options of the chart itself. Within a list, options are processed from left to right. Thus more specific settings always override more general ones, and later settings override earlier ones. Example:

```
\usepackage[slanted, fg=blue!50!black]{vowelchart}   % default for all charts
...
\vowelchartset{linewidth=.3pt, fontsize=10pt}        % for all following charts
...
\begin{vowelchart}[normal, fg=black, width=5cm]      % this chart only
```

Some remarks:

- `\vowelchartset` acts globally, even inside a group, and the settings accumulate. You cannot delete a default, but you can override it in a chart with the opposite option, e.g. `normal` for the default shape, `ipanew` for the default grid style, `four` for the default number of vertical levels, or `bg=white`. An empty value (e.g. `labelbg=`) restores the default of the colors for labels (Section 5.5).
- Shape options set all four corners, so a predefined shape given later overrides corners given before (Section 3.2).
- A bare dimension such as `6cm` is a shorthand for `width=6cm`.
- Values containing a comma have to be put in braces, e.g. `tl={0,0.5}`.

A complete list of all options with their defaults is given in Section 9.

3.2 Shape

The shape is defined by four corners: top left (`tl`), top right (`tr`), bottom left (`bl`) and bottom right (`br`). The unit is one quarter of the width of the default chart; the y axis points downwards. The standard shapes (shown in Figure 1 and Table 1) set all four corners.

Option	tl	tr	bl	br
normal (default)	0, 0	4, 0	2, 3	4, 3
rectangle	0, 0	4, 0	0, 3	4, 3
triangle	0, 0	4, 0	2, 3	2, 3
symmetric	0, 0	4, 0	1, 3	3, 3
slanted	0, 0	3.6, 0.7	2.4, 2.7	4, 2.7

Additionally, you can set the corners of the quadrangle manually.

```
tl={<x>,<y>}   tr={<x>,<y>}   bl={<x>,<y>}   br={<x>,<y>}
```

Corners not mentioned keep the values of the chosen shape, so you can start from a standard shape and change a single corner. Since a shape option overwrites *all* four corners, it has to come before individual

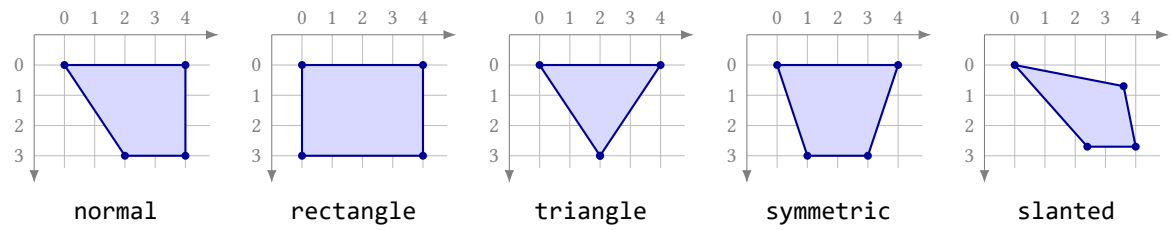
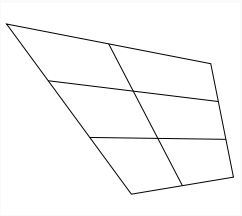


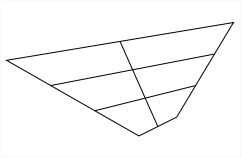
Figure 1: The coordinate system used for defining the corners (the y axis points downwards)

corner options. The braces are required, because the comma would otherwise separate options. Values are evaluated by pgfmath, so mathematical expressions are allowed.

```
\begin{vowelchart}[slanted, bl={2.2,3},
↪  width=3cm]\end{vowelchart}\quad
```



```
\begin{vowelchart}[tl={0, 2/3}, tr={4, 0}, bl={7/3, 2},
↪  br={3, 5/3}, width=3cm]\end{vowelchart}
```



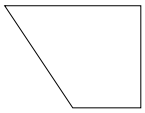
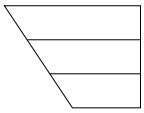
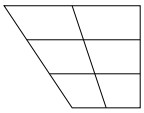
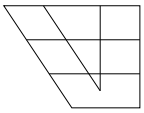
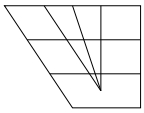
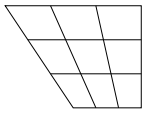
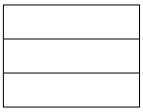
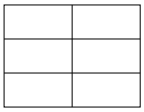
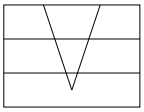
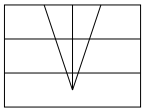
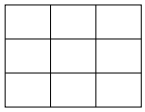
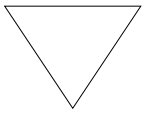
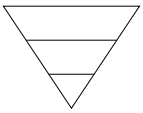
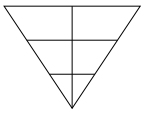
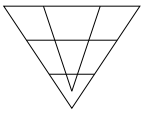
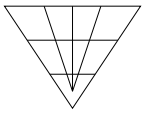
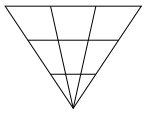
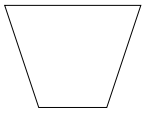
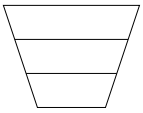
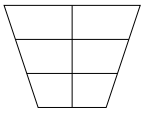
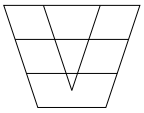
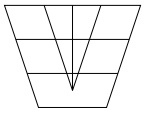
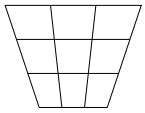
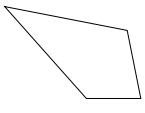
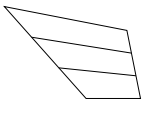
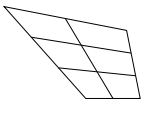
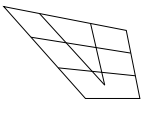
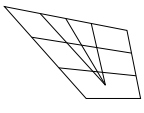
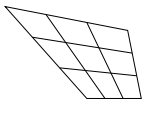
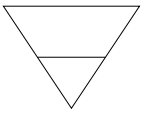
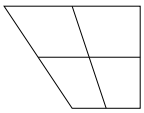
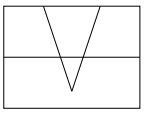
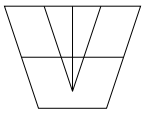
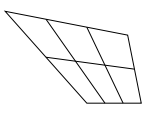
	plain	simple	ipanew	central-zone	standard	central-col
normal (0,0) (4,0) (2,3) (4,3)						
rectangle (0,0) (4,0) (0,3) (4,3)						
triangle (0,0) (4,0) (2,3) (2,3)						
symmetric (0,0) (4,0) (1,3) (3,3)						
slanted (0, 0) (3.6, 0.7) (2.4, 2.7) (4, 2.7)						
three (with various shapes)						

Table 1: Options for chart shapes and grid styles

3.3 Grid style

Option	Lines
`plain`	outline only
`simple`	outline and horizontal line(s)
`ipanew` (default)	`simple` plus a vertical center line (as in the current IPA chart)
`centralzone`	`simple` plus a triangular central zone
`standard`	`centralzone` plus a vertical line through the middle of the central zone (the name `standard` has been kept from the `vowel` package)
`centralcol`	`simple` plus two vertical lines, resulting in a 3 × 3 division of the vowel space
`four` (default)	four heights are distinguished (i.e. two horizontal middle lines are drawn in all styles except `plain`)
`three`	only three heights (i.e. only one horizontal middle line)

If you need additional lines, you can easily add them with `\draw` commands, using the convenient relative coordinates explained in Section 4.2 (see Figure 2) via the coordinate system `vowel cs` (Section 5.2).

Grid styles and shapes are independent of each other and can be combined freely. Combinations of shapes and grid styles are shown in Table 1.

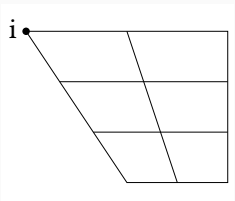
3.4 Size

```
width=<dimension>     height=<dimension>     <dimension>
```

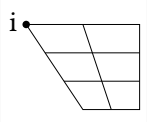
`width` sets the exact width and `height` the exact height of the quadrilateral grid (excluding overhanging labels and dots). A bare dimension such as `[6cm]` is a shorthand for `width`. There is only one scale for both directions; the chart is never distorted. If both are given, the last one wins. The default is `width=8em`.

Because `em` refers to the font size of the chart (the font size at the beginning of the environment, after applying the options `font` and `fontsize`), commands such as `\small` or `\Large` before the chart scale it together with its symbols unless the size is set with an absolute measure of length. The default line widths and dot sizes are absolute (Section 3.7).

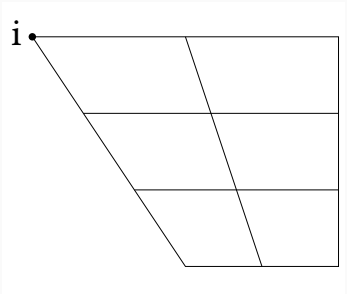
```
\begin{vowelchart}[height=2cm]
  \putipavowel{i}
\end{vowelchart}
```




```
\begin{vowelchart}[width=1.5cm]
  \putipavowel{i}
\end{vowelchart}
```



```
{\Large
  \begin{vowelchart}
    \putipavowel{i}
  \end{vowelchart}
}
```



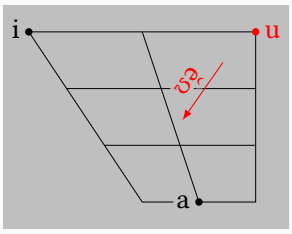
3.5 Colors

Option	Colors
`fg`	the foreground color of lines, dots and labels (default: the current text color, usually black)
`bg`	the background color (default: white)
`labelbg`	the color of the area cleared behind labels (default: = `bg`; can also be `off`; see Section 5.5)
`contour`	the color of the contour around labels (default: = `labelbg`; can also be `off`; see Section 5.5)

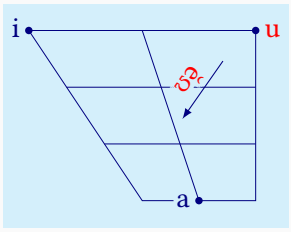
By default, everything is drawn in the current text color on a white background. You can change this by giving color definitions in the variables `bg` for the background color and `fg` for the foreground color. The background color is used for a rectangle behind the whole chart and also for the areas that clear the lines around the labels; it should therefore be the color of the surrounding if you use a transparent-looking chart. Both options accept all color expressions of `xcolor`, e.g. `bg=cyan!15`.

Note that you can use color within vowel charts as well. If you use `\color` before one of the commands for placing vowels, both the dot or arrow and the label will appear in this color. If you want to change the color of the label separately from the corresponding dot or arrow, you can use `\color` or `\textcolor` within the label argument of the `\put`... command. Inside the chart, `\color{vowelchartfg}` restores the foreground color of the chart (Section 5.3).

```
\begin{vowelchart}[bg=lightgray,width=3cm]
  \putipavowel{i}
  {\color{red}
    \putipavowel{u}
    \putdiphthong{ʊ̯}{1.7}{0.5}{1.1}{1.6}
  }
  \putipavowel{a'}
\end{vowelchart}
```

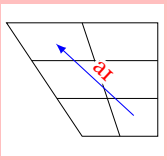



```
\begin{vowelchart}[fg=blue!50!black, bg=cyan!15,width=3cm]
  \putipavowel{i}
  \putvowel[r]{\color{red} u}{2}{0}
  \putdiphthong{\textcolor{red}{ʊ̯}}{1.7}{0.5}{1.1}{1.6}
  \putipavowel{a'}
\end{vowelchart}
```



```
\colorbox{pink}{%
  \begin{minipage}{5cm}
    \small A standard vowel chart in a surrounding with a
    ↪  different background color: \medskip \\
    \begin{vowelchart}[width=2cm]
      \color{blue}
      \putdiphthong{\textcolor{red}{aɪ}}{1.5}{2.5}{0.5}{0.5}
    \end{vowelchart}
  \end{minipage}
}
```

A standard vowel chart in a surrounding with a different background color:

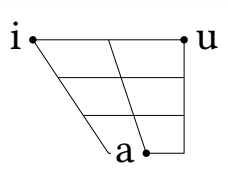


3.6 Font and font size

```
font=<commands>          fontsize=<dimension>
```

The option `font` executes the given font commands at the beginning of the chart, e.g. `font=\large`, `font=\sffamily` or `font=\itshape\small`. `fontsize` sets a fixed size such as `fontsize=12pt` (the baseline skip becomes 1.2 times the size). Both can be combined; if they set the same property, the later one wins. The font applies to all labels and to the nodes drawn inside the chart. Lengths given in `em` or `ex` (the default width, the distances around labels and the line width if you give it in `em`) refer to this font size, so the whole chart scales with it. Instead of the options, you can also put size commands such as `\Large` before the environment.

```
\begin{vowelchart}[2cm, fontsize=16pt]
  \putipavowel{i}
  \putipavowel{a'}
  \putipavowel{u}
\end{vowelchart}
```

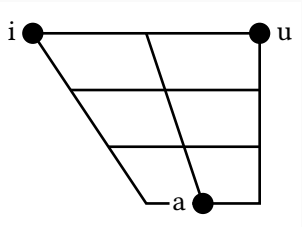


3.7 Line width and dot size

```
linewidth=<dimension>          dotsize=<dimension>
```

`linewidth` sets the width of the lines of the chart (default: `0.2pt`), `dotsize` the radius of the dots next to vowels placed with `[l]` or `[r]` (default: `1.4pt`). Both are absolute unless you give them in `em`. (The values are available for your own TikZ code as `\vowelchartline` and `\vowelchartdot`; see Section 5.3).

```
\begin{vowelchart}[width=3cm, linewidth=1pt, dotsize=4pt]
  \putipavowel{i}
  \putipavowel{u}
  \putipavowel{a'}
\end{vowelchart}
```



Further options tune the white areas around the labels and are described in Section 7, and another option is described in Section 4.6.

4 The commands: Placing vowels

4.1 Placement options

The commands `\putvowel` and `\putcvowel` take a list of options in their first optional argument. The following keys decide how the symbol is placed:

`x` (default)	symbol centered on the position, no dot
`l`	dot at the position, symbol to the left of the dot
`r`	dot at the position, symbol to the right of the dot

In addition, the keys `labelbg=<color|off>` and `contour=<color|off>` change the colors of the area cleared behind the label (Section 5.5). The keys are separated by commas (e.g. `\putvowel[l, labelbg=yellow]{e}{0}{1}`).

If you leave the *vowel label empty*, only the dot is drawn, which is useful for marking positions (see the example in Section 9).

Each command also accepts a trailing optional *node name*. The symbol becomes a TikZ node of that name (Section 5.1). Small horizontal corrections of a single symbol can be made by adding `\hspace` inside the symbol argument.

4.2 `\putvowel`

```
\putvowel[<options>]{<symbol>}{<h>}{<v>}[<node name>]
```

This command places a symbol at the abstract position (h, v). Vowels are not placed with absolute coordinates, but with relative coordinates that follow the shape of the chart, so the same code works in every shape:

- h runs from 0 (left edge) to 2 (right edge)
- v runs from 0 (top edge) to 3 (bottom edge).

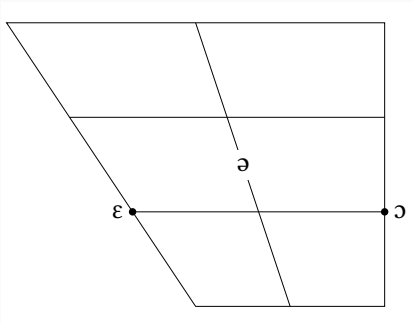
Thus $h = 1$ is always on the center line and $v = 2$ is always at the height of cardinal [ɛ] and [ɔ], whatever the shape. Values outside the ranges are allowed (e.g. $h = -0.5$ is left of the chart). The same coordinates are used by `\putdiphthong` and by the TikZ coordinate system `vowel cs` (Section 5.2).

Figure 2 marks the v values (blue, along the left and right edges) and the h values (red, along the top and bottom edges) in all five predefined chart shapes and in one user-defined shape.

If you use the shape option `triangle` and you want to place a vowel at the very bottom at $v = 3$, the h coordinate, logically, has no effect; however, by setting $h = 1$ anyway you ensure that this will still be an open *central* vowel if the chart is ever displayed in a different shape.

Example:

```
\begin{vowelchart}[width=5cm]
  \putvowel[l]{ɛ}{0}{2}[vowele]
  \putvowel[r]{ɔ}{2}{2}
  \putvowel{ə}{1}{1.5}
\end{vowelchart}
```



For charts created with `vowel`, a legacy variant with absolute coordinates exists (Section 6.2).

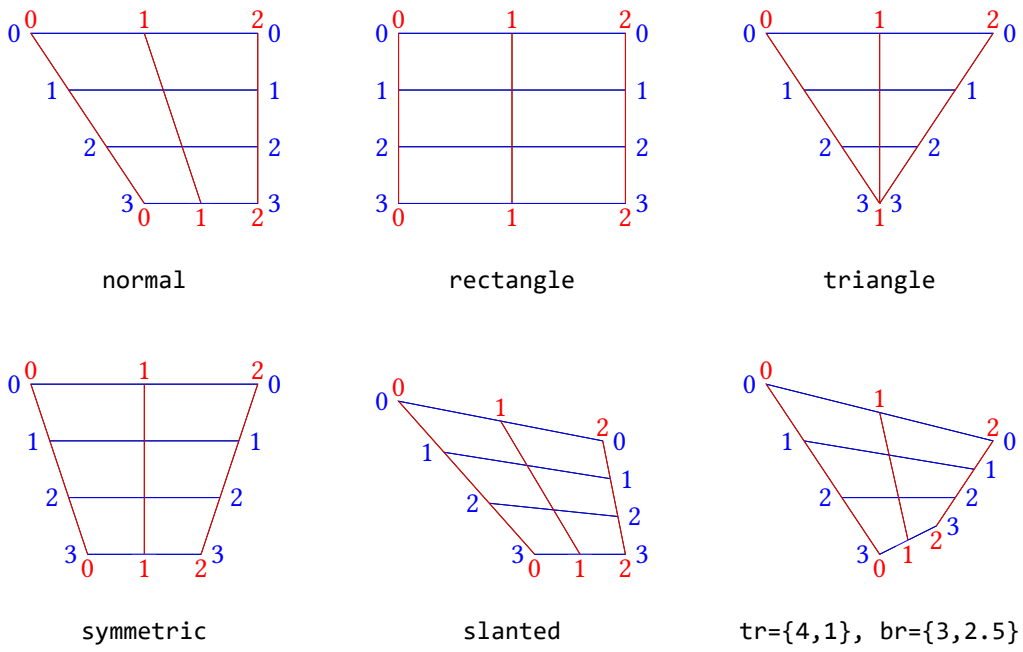
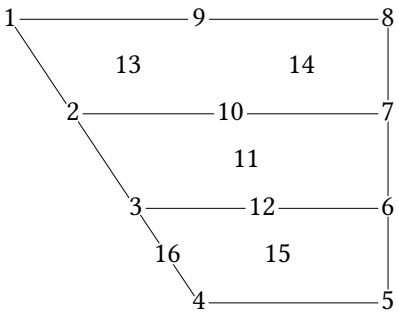


Figure 2: The relative coordinate system, independent of chart shape

4.3 \putcvowel

```
\putcvowel[<options>]{<symbol>}{<n>}[<node name>]
```

This places a symbol at cardinal position $n = 1, \ldots, 16$. Positions 1–8 are those of Daniel Jones's cardinal vowels; 9–16 are extended positions for the other vowels of the IPA chart. The default node name is vn. The positions are given in the relative coordinates of Section 4.2 and thus follow the shape of the chart.



n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
h	0	0	0	0	2	2	2	2	1	1	1	1	0.5	1.5	1	0
v	0	1	2	3	3	2	1	0	0	1	1.5	2	0.5	0.5	2.5	2.5

4.4 \putipavowel

```
\putipavowel[<options>][<label>]{<IPA or X-SAMPA symbol>}[<node name>]
```

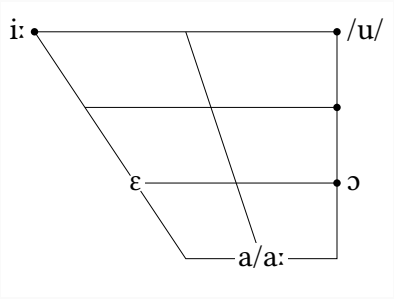
This command places a vowel at its place in the IPA chart. It also typesets the IPA symbol (even if your input is X-SAMPA); if you want something else to appear (e.g. a length mark or brackets), put it in the optional `<label>`, which replaces the default symbol. The shape of the chart does not matter. For symbols that share a position, the dot is at the same place and the symbols stand on its left (unrounded) or right (rounded) side; ə and ɐ normally have no dot.

The command has two optional arguments before the symbol. If only one is given, it is the `<label>`. If two are given, the first one holds the `<options>` for this vowel and the second one the `<label>`, which may be empty (`[]`) to draw only the dot. If you want to give options but keep the default label, repeat the symbol as the label, e.g. `\putipavowel[labelbg=yellow][e]{e}`.

The position of the symbol is determined by the input in the obligatory argument, which can be the IPA symbol in Unicode (works in XeLaTeX and LuaLaTeX) or the corresponding X-SAMPA code (with a few exceptions due to illegal characters). The input `a'`, `ae'`, `e'`, etc. selects the non-canonical but widely practiced use of [a] for an open *central* vowel and of [æ] for an *open* front vowel and places e, ɛ, o, ɔ, ø and œ as *mid* vowels. You can also place ə and ɐ on the left or right of a dot by using `'`. See Table 3 for all possible input characters. An unknown symbol in this argument raises an error.

The `<options>` in the first of the two optional arguments can contain `labelbg` and `contour` (Section 5.5), the settings listed in Table 5, and `l`, `r` or `x`, which override the position of the label that is predefined for the symbol.

```
\begin{vowelchart}[width=4cm, font=\large]
  \putipavowel[i:]{i}
  \putipavowel[/u/]{u}[vu]
  \putipavowel{O}
  \putipavowel[a/a:]{'a'}
  \putipavowel[]{o} % empty label → only dot
  \putipavowel[x][ɛ]{E} % options, then label
\end{vowelchart}
```



4.5 \putdiphthong

```
\putdiphthong[<TikZ options>]{<label>}{<h1>}{<v1>}{<h2>}{<v2>}[<node name>]
```

This draws an arrow from (h_1, v_1) to (h_2, v_2) in the relative coordinates (see Section 4.2). The optional label runs along the arrow, centered, always readable: TikZ turns the text by $180°$ if it would be upside down, so it stands to the right of an arrow pointing to the upper left and to the left of an arrow pointing to the upper right. The label is protected from the lines of the chart by a contour (Section 7). Use `{}` as the label for an arrow without label.

The first optional argument holds TikZ options for this one arrow; besides the usual TikZ keys, it accepts `labelbg` and `contour` (Section 5.5). Furthermore, the position of the label relative to the arrow is set with the option `diphlabelpos`, which can be given here, in the properties of the chart, with `\vowelchartset`, or when loading the package (Section 3.1):

```
diphlabelpos=above|below|on
```

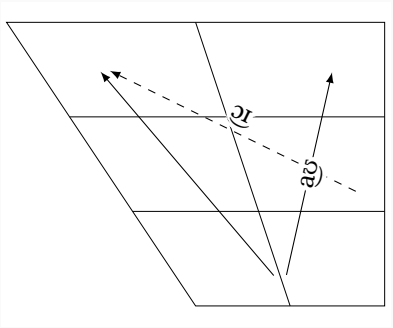
IPA	Accepted input	h	v	dot
i	`i`	0	0	l
y	`y`	0	0	r
ɨ	`ɨ, 1`	1	0	l
ʉ	`ʉ, u-, \}`	1	0	r
ɯ	`ɯ, M`	2	0	l
u	`u`	2	0	r
ɪ	`ɪ, I`	0.33	0.5	l
ʏ	`ʏ, Y`	0.33	0.5	r
ʊ	`ʊ, U`	1.67	0.5	r
e	`e`	0	1	l
e (mid)	`e'`	0	1.5	l
ø	`ø, 2`	0	1	r
ø (mid)	`ø', 2'`	0	1.5	r
ɘ	`ɘ, @-`	1	1	l
ɵ	`ɵ, 8`	1	1	r
ɤ	`ɤ, 7`	2	1	l
o	`o`	2	1	r
o (mid)	`o'`	2	1.5	r
ə	`ə, @`	1	1.5	–
ə (unrounded)	`ə', @'`	1	1.5	l
ə (rounded)	`'ə, '@`	1	1.5	r
ɛ	`ɛ, E`	0	2	l
ɛ (mid)	`ɛ', E'`	0	1.5	l
œ	`œ, oe, 9`	0	2	r
œ (mid)	`œ', oe', 9'`	0	1.5	r
ɜ	`ɜ, 3`	1	2	l
ɞ	`ɞ, 3-`	1	2	r
ʌ	`ʌ, V`	2	2	l
ɔ	`ɔ, O`	2	2	r
ɔ (mid)	`ɔ', O'`	2	1.5	r
æ	`æ, ae, \{`	0	2.5	l
æ (open)	`æ', ae', \{'`	0	3	l
ɐ	`ɐ, 6`	1	2.5	–
ɐ (unrounded)	`ɐ', 6'`	1	2.5	l
ɐ (rounded)	`'ɐ, '6`	1	2.5	r
a	`a`	0	3	l
a (central)	`a'`	1	3	l
a (central, no dot)	`'a'`	1	3	–
ɶ	`ɶ, &, OE`	0	3	r
ɑ	`ɑ, A`	2	3	l
ɒ	`ɒ, Q`	2	3	r

Table 3: Predefined IPA vowels

`above` (the default) puts the label on the upper side of the arrow, `below` on the lower side, `on` puts the label on the arrow so that the arrow runs through the middle of the x-height (the arrow is interrupted by the label's contour, Section 7).

The `<node name>` belongs to the label node in the middle of the arrow. If you give a name but no label, an empty node of that name is placed in the middle of the arrow; with neither, no node is created.

```
\begin{vowelchart}[width=5cm]
  \putdiphthong{}{0.95}{2.7}{0.35}{0.5} % no label
  \putdiphthong[dashed]{ɔ͡ɪ}{1.8}{1.8}{0.4}{0.5}
  \putdiphthong[diphlabelpos=on]{a͡ʊ}{1.05}{2.7}{1.7}{0.5}
\end{vowelchart}
```



The appearance of arrow and label is controlled by these TikZ styles:

```
\tikzset{
  vc diph label/.style={sloped, midway, inner sep=0pt},
  vc diph above/.style={above=0.5pt},   % used by diphlabelpos=above
  vc diph below/.style={below=0.15em},  % used by diphlabelpos=below
  vc diph on/.style={anchor=base},      % used by diphlabelpos=on}
}
```

You can redefine them with `\tikzset`, e.g. `\tikzset{vc diph label/.append style={pos=0.35}}` (label closer to the start). `vc diph` can also be used for your own arrows between nodes.

4.6 `\putipadiphthong`

```
\putipadiphthong[<TikZ options>]{<label>}{<start>}{<end>}[<node name>]
```

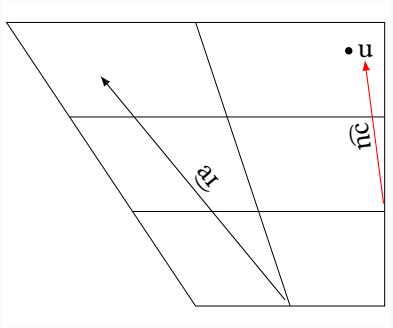
This command works like `\putdiphthong`, but start and end of the arrow are not given as numbers. Each of them is either

- an IPA or X-SAMPA symbol from Table 3 (e.g. `a'`, `I`, `ʊ`): the arrow starts or ends at the position of this vowel in the IPA chart, even if no vowel has been placed there; or
- a TikZ coordinate *in parentheses*: a node such as `(vu)`, a node with anchor such as `(vu.south)`, `(vowel cs:h=1,v=2)` or `(2,-1)`.

The two kinds can be mixed, e.g. to draw an arrow from a node of your own to an IPA vowel. The parentheses decide which kind is meant, so a node named `i` cannot be confused with the vowel i.

The label * is replaced by the two IPA symbols of the start and the end, joined by the tie bar (U+035C) that is used in the transcription of diphthongs, as in *a͡ɪ*. This works only if both ends are symbols. The tie is stored in the macro `\vowelchartdiphtie`; `\renewcommand\vowelchartdiphtie{}` removes it.

```
\begin{vowelchart}[width=5cm]
  \putipadiphthong{*}{a'}{I}
  \putvowel[r]{u}{1.8}{0.3}[vu]
  \putipadiphthong[red,
  ↪  diphlabelpos=below]{ɔʊ}{O}{(vu.south)}
\end{vowelchart}
```



Arrows that end at an IPA position are shortened at both ends by the length given in the option

```
ipashorten=<dimension>          % default: 3pt
```

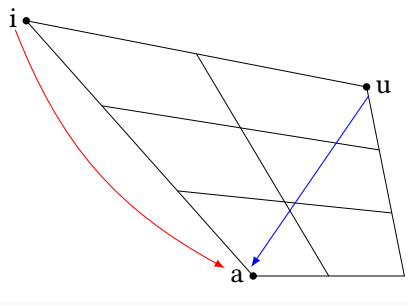
so that they do not touch the dot or the vowel at this position. Ends at nodes or coordinates are not shortened beyond the 1 pt of the style `vc diph`. For a single arrow, you can override this with TikZ options, e.g. `\putipadiphthong[shorten >=0pt]{...}{a'}{I}`.

5 Using TikZ in the vowel chart

Since the chart is a `tikzpicture`, everything TikZ offers is available inside the environment. This section explains how to refer to positions and nodes in the chart and which settings of the chart you can use in your own code.

5.1 Nodes

```
\begin{vowelchart}[slanted, width=5cm]
  \putipavowel{i}[vi]
  \putipavowel{u}[vu]
  \putipavowel{a}[va]
  \draw[vc diph, red]  (vi.south) to[bend right=20] (va.north
  ↪  west);
  \draw[vc diph, blue] (vu.south west) -- (va.north east);
\end{vowelchart}
```



Without a name, the symbol gets an automatic node name (`vcauto1`, `vcauto2`, ...); `\putcvowel` names its nodes `v1`, ..., `v16` by default. These names should not be used for nodes of your own. Nodes are local to their picture, so the same names may be used in different charts. Use only ASCII letters and digits in node names (no dot, colon or comma). A node is the symbol's *label* (a box one `ex` high, with its baseline half an `ex` below the dot), not the dot. If a vowel should be known under a second name, use TikZ's `\pgfnodealias{<new name>}{<name>}` after placing it.

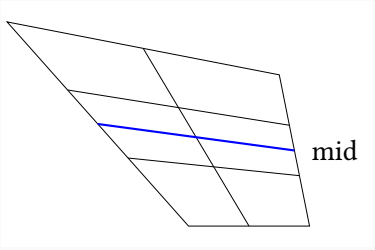
5.2 The coordinate system `vowel cs`

```
(vowel cs:h=<h>, v=<v>)          (vowel cs:x=<h>, y=<v>)
```

This is a TikZ coordinate system with the relative coordinates of Section 4.2 (h from 0 to 2 across, v from 0 to 3 down, see Figure 2). It can be used wherever TikZ expects a coordinate, and it follows the

shape and size of the chart. `x` and `y` are synonyms of `h` and `v`; missing values are 0. It works only inside `vowelchart`. With `\usetikzlibrary{calc}`, offsets are possible: `($(vowel cs:h=1,v=1)+(0,-.3)$)`.

```
\begin{vowelchart}[slanted, width=4cm]
  \draw[blue, thick]
  (vowel cs:h=0,v=1.5) -- (vowel cs:h=2,v=1.5);
  \node[anchor=west, xshift=1mm]
  at (vowel cs:h=2,v=1.5) {mid};
\end{vowelchart}
```



Plain TikZ coordinates (x, y) also work inside the chart, but they do not follow the shape. Their unit is one quarter of the width of the default shape, and the y axis points *upwards*, so that the chart lies at negative y values: a corner given as `br={4,3}` is the point `(4,-3)`. Figure 3 shows this absolute coordinate system for the predefined shapes. Use `vowel cs` for everything that should stay in place if the shape changes.

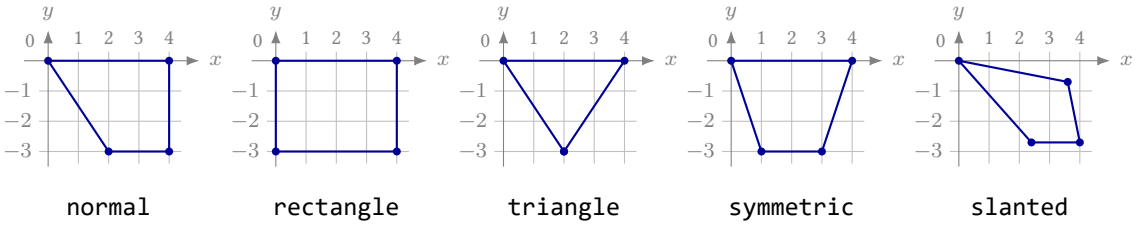


Figure 3: The absolute coordinate system for TikZ commands inside a `vowelchart`. The y axis points upwards, so the chart lies at negative y values: a corner given as `br={4,3}` is the point `(4,-3)`.

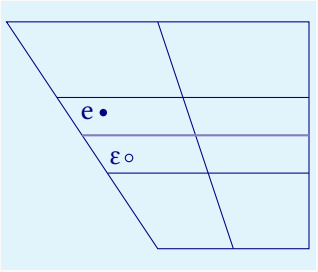
5.3 Using the chart's settings in your own code

Inside the environment, the settings of the chart are available under the following names. TikZ commands that use them follow any change of design automatically.

Name	Kind	Meaning (set by)
`\vowelchartline`	length	line width of the chart (`linewidth`)
`\vowelchartdot`	length	radius of the dots (`dotsize`)
`vowelchartfg`	color	foreground color (`fg`; default: the text color at the beginning of the chart)
`vowelchartbg`	color	background color (`bg`)
`vc bg`	style	fills with the background color: `\node[vc bg]{...}`
`vc diph`, `vc diph label`	styles	arrows and arrow labels (Section 4.5)
`\vowelchartdiphtie`	macro	tie bar of automatic diphthong labels (Section 4.6)
`\vowelhunit`, `\vowelvunit`	lengths	the unit of the chart (for legacy charts, see Section 6.2)

The colors and lengths hold the values set by the options (or their defaults), so they are only meaningful inside the environment. After you have changed the color with `\color`, `\color{vowelchartfg}` restores the chart's foreground color.

```
\begin{vowelchart}[fg=blue!50!black, bg=cyan!10,
↪  linewidth=.4pt, width=4cm]
  \draw[line width=2\vowelchartline, vowelchartfg!50]
  (vowel cs:h=0,v=1.5) -- (vowel cs:h=2,v=1.5);
  \putvowel[l]{e}{0.3}{1.2}
  \draw[vowelchartfg] (vowel cs:h=0.3,v=1.8) circle
  ↪  (\vowelchartdot);
  \putvowel[x]{ɛ}{0.17}{1.8}
\end{vowelchart}
```



5.4 Further possibilities

Here are just a few more ideas what you can do with a vowel chart and TikZ:

- **Colour**: put `\color` or a group before the chart, (e.g. `{\color{gray}\begin{vowelchart} ... \end{vowelchart}}`), to color lines, dots and symbols (or use the option `fg`). Inside the chart, `\color` or a `scope` colors what follows.
- **Shading and highlighting**: fill regions with `\fill[blue, opacity=.15] (vowel cs:h=0,v=0) -- ... -- cycle;`. Labels that are placed on a colored area can be given the same background (Section 5.5).
- **Annotations**: `\node` with `vowel cs`, or placed relative to the vowel nodes, e.g. `(voweli.north)`.
- **Arrows and curves**: `\draw` between nodes or coordinates, with `to[bend left]`, `dashed`, arrow tips from `arrows.meta`, etc.
- **Offsets**: with the `calc` library, use `($(vowel cs:h=1,v=1)+(0,-.3)$)`.
- **Styles**: change `vc diph` and `vc diph label` globally (Section 4.5).

Note that `\begin{vowelchart}` takes no TikZ options; use `\tikzset` or scopes for global settings. The size of the chart cannot be changed with `scale`; use `width` or `height`, which keep the size of the symbols. The bounding box of the picture includes all nodes, including labels outside the chart.

5.5 Changing the white areas for colored backgrounds

If you draw colored areas in the chart with TikZ, the white areas behind the labels would stand out. For this case, the color of the cleared areas can be changed or switched off with two keys:

Key	Effect
`labelbg=<color>`	color of the box behind the label, of the strip between a dot and its label and of the field behind central vowels (default: the chart background `bg`)
`labelbg=off`	none of these three is drawn
`contour=<color>`	color of the contour around the label (default: the same color as `labelbg`; with `labelbg=off`, the chart background)
`contour=off`	no contour is drawn

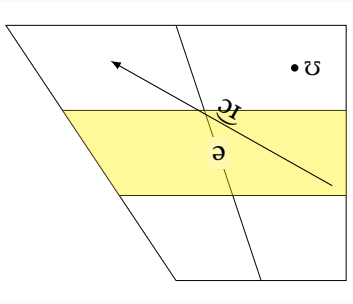
With `labelbg=off` (and `contour=off`), lines run through the label; this makes sense for labels that stand outside the lines or on areas where you take care of the clearing yourself.

The keys can be given on every level:

```
% for an individual label:
  \putvowel[l, labelbg=yellow]{e}{0}{1}
  \putcvowel[l, labelbg=yellow]{e}{2}
  \putipavowel[labelbg=yellow][e]{e}
  \putdiphthong[contour=off]{...}
  \putipadiphthong[contour=off]{...}
% for all following labels in a chart:
  \vowelchartlabels{labelbg=yellow}
% for all labels in a chart:
  \begin{vowelchart}[contour=off]
% for all charts:
  \usepackage{vowelchart}[labelbg=off]
  \vowelchartset{labelbg=yellow}
```

A setting for a single label overrides the settings of the higher levels. An empty value (`labelbg=`) restores the default.

```
\begin{vowelchart}[width=4.5cm, font=\large]
  \fill[yellow, opacity=.4] (vowel cs:h=0,v=1) -- (vowel
  ↪  cs:h=2,v=1) -- (vowel cs:h=2,v=2) -- (vowel cs:h=0,v=2)
  ↪  -- cycle;
  \putipavowel[labelbg=yellow!40!vowelchartbg][ə]{ə}
  \putdiphthong[contour=off]{ɔ͡ɪ}{1.9}{1.9}{0.5}{0.4}
  \putipavowel{ʊ}
\end{vowelchart}
```



Note that a translucent area (`opacity=.4`) produces a mixed color. To make the cleared areas match, use the mixture itself, as in the example: yellow at 40 % on the chart background is `yellow!40!vowelchartbg`.

6 Compatibility with `vowel` and `pst-vowel`

6.1 Moving from `(pst-)vowel` to `vowelchart`

The package `vowelchart` neither requires or changes `vowel` or `pst-vowel`. The environment is called `vowelchart` instead of `vowel`. The homonymous commands `\putvowel` and `\putcvowel` as well as the new commands `\putipavowel`, `\putdiphthong` and `\putipadiphthong` are defined *only inside* the `vowelchart` environment. In `vowel` environments the commands of `vowel` and `pst-vowel` keep working, so old and new charts can coexist in the same document.

For an overview of the differences between the three packages, see Table 4.

If you already have vowel charts created with `vowel` (with or without `pst-vowel`), you can transfer them to this package in the following steps:

1. Rename the environment: `\begin{vowel}` → `\begin{vowelchart}`. The options `plain`, `simple`, `standard`, `ipanew`, `rectangle`, `triangle` and `three` keep their names.

2. `\putcvowel` needs no change, including the optional `[l]` and `[r]`.

3. `\putvowel` changes: the coordinates of the new command are no longer absolute lengths but relative coordinates, see Section 4.2. The easiest way to convert the coordinates is to read the position off the old chart and estimate its relative position by eye. For a more mathematical approach, see Sections 8.2 and 8.3. If you do not want to convert the coordinates at all, the legacy command `\putvowel*` (Section 6.2) accepts the old absolute coordinates unchanged.

	vowel	pst-vowel	vowelchart
Output	built from boxes and tiny squares	= vowel	TikZ vector paths
Engines	any	LaTeX+dvips (PSTricks)	LuaLaTeX, XeLaTeX, pdfLaTeX
Size	\vowelhunit, \vowelvunit, font size	= vowel	width=, height=, font size (\vowelhunit and \vowelvunit still work for legacy charts)
Chart shapes	quadrangle, rectangle, triangle	= vowel	+ slanted, symmetric, freely defined corners
Grid options	plain, simple, standard, ipanew; three	= vowel	+ centralzone, centralcol (+ TikZ)
Vowel placement	\putvowel with absolute coordinates	= vowel	\putvowel with relative coordinates (and \putvowel* with absolute coordinates for legacy charts)
Cardinal vowels	\putcvowel{...}{n}	= vowel	= vowel
IPA vowels	—	—	\putipavowel with Unicode or X-SAMPA input
Nodes	none	obligatory (PSTricks)	optional (TikZ)
Arrows	none	PSTricks	\putdiphthong, \putipadiphthong, TikZ

Table 4: Comparison of packages displaying vowel charts

4. The size of the chart is set with width= or height=. For compatibility, font size commands can still be used to change the chart's size, and the default width of 8em equals vowel's default of 2em per unit. (\vowelhunit and \vowelvunit are also still understood, see Section 6.2.)

5. The options noerase, t and b are not supported. The chart can be aligned with \raisebox.

6. vowel allows different values for \vowelhunit and \vowelvunit, which distorts the chart. This is not supported. If both are set, \vowelhunit wins. (A distorted chart can be created by moving the corners, see Section 3.2.)

6.2 Legacy: absolute coordinates with \putvowel*

```
\putvowel*[l|r|x]{<symbol>}{<x>}{<y>}[<node name>]
```

This command exists only to make old charts work with a minimum of changes. It is not recommended for new charts. Use the relative coordinates of \putvowel (Section 4.2) or the IPA positions of \putipavowel (Section 4.4) instead.

With the star, \putvowel* takes the same absolute coordinates as in vowel: lengths measured from the top left corner of the default chart, with the y axis pointing downwards, usually multiples of \vowelhunit and \vowelvunit. A chart created with vowel can therefore be converted by renaming the environment and adding a star to every \putvowel. If the chart was created with pst-vowel, the node name, which was the last obligatory argument there, must in addition be moved into square brackets:

```
% vowel.sty
\putvowel[l]{ɛː/ɛ}{1.33\vowelhunit}{2\vowelvunit}
% vowelchart
\putvowel*[l]{ɛː/ɛ}{1.33\vowelhunit}{2\vowelvunit}

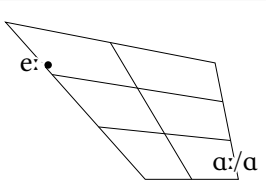
% pst-vowel.sty
\putvowel[l]{ɛː/ɛ}{1.33\vowelhunit}{2\vowelvunit}{Elongshort}
% vowelchart
\putvowel*[l]{ɛː/ɛ}{1.33\vowelhunit}{2\vowelvunit}[Elongshort]
```

The starred command `\putvowel*` accepts the same options as the normal `\putvowel` (Section 4.2), and the new options of `vowelchart` for changing size, colors, grid styles, etc., and for adding TikZ commands are available. Note, however, that the chart *shape* cannot be changed without the vowels placed with `\putvowel*` ending up in the wrong places. For technical reasons, the additional white field that clears the center line behind central vowels (see Section 7) is not drawn with `\putvowel*` (which is not a problem in most cases; where it is, just use the unstarred version to place the vowel instead.)

7 A known problem and some solutions

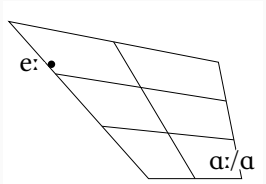
The space that the lines leave for vowel symbols written over them is created by white boxes and contours (or boxes and contours in the background color of the chart) drawn around the symbols, which are drawn after the lines and thus erase them. The size of these boxes and contours has been tweaked as much as possible to erase only those lines that should be erased. However, vowel symbols placed near diagonal lines can occasionally cause ugly missing bits in those diagonals, see the diagram below. (This is a problem that `vowel` sometimes had as well.)

```
\begin{vowelchart}[slanted] % with tricky labels
  \putvowel[l]{eː}{0.06}{0.8}
  \putvowel[x]{ɑː/ɑ}{2}{2.6}
\end{vowelchart}
```



To prevent this, you have three options. **First**, you can use `\hspace` or `\vspace` with positive or negative values within the label argument of `\putvowel` (e.g. `\putvowel[l]{\hspace{1.5pt}i}{0.1}{0.1}`) to move the label into a position where the problem does not occur; see the following diagram.

```
  \begin{vowelchart}[slanted] % corrected by moving the
  ↪  labels
  \putvowel[l]{\hspace{-1.1em}eː}{0.06}{0.8}
  \putvowel[x]{\hspace{-5pt}ɑː/ɑ}{2}{2.6}
\end{vowelchart}
```

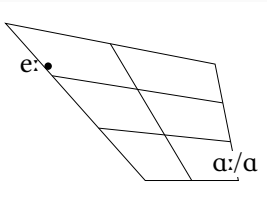


Second, you might want to tweak the size of the areas that are cleared for the labels. It can be changed with the options listed in Table 5. They are given like all other options (Section 3.1), i.e. after `\begin{vowelchart}` for a single chart, with `\vowelchartset` for all following charts, as a package option, or (except for `diphlabelpos`) in the options of a single vowel command, for example `\putvowel[l,pad=6pt]{e}{0.06}{0.8}` or `\putipavowel[pad=6pt][u]{u}`. An option given in a vowel command only applies to this vowel, so that you can make more space for one label without affecting any others. For `\putdiphthong` and `\putipadiphthong`, `diphmargin`, `labelbg`, and `contour` can be given in the TikZ options. Lengths given in `em` or `ex` refer to the font size of the chart, so the cleared areas grow and shrink together with the symbols.

Option	Default	Effect
`margin`	`0.06em`	Width of the contour around the glyphs. It clears the lines right at the outline of each symbol and is the main protection above and below the symbols.
`pad`	`0.10em`	The white box behind each symbol is this much wider than the symbol on each side.
`boxh`, `boxd`	`1ex`, `0pt`	Height above and depth below the baseline of this box. By default it reaches from the baseline to the x-height; the contour takes care of ascenders and descenders.
`gap`	`0.18em`	Distance between the symbol and the dot next to it (labels placed with `[l]` or `[r]`).
`lift`	`0.5ex`	The baseline of a label lies this far below the dot (or the position of a label placed with `[x]`).
`stripe`	`1.5pt`	Thickness of the white strip between a dot and its label that removes the piece of the horizontal line that would otherwise remain there.
`zonew`, `zoneh`	`0.7em`, `1em`	Size of the white field that is drawn behind central vowels (see below). The field also covers a piece of any horizontal line close to the vowel; for a vowel at the very top or bottom of the chart (e.g. `\putcvowel[x]` at position 9), choose a smaller `zoneh`, or switch the field off with `zoneh=0pt`.
`labelbg`	`bg`	Color of the white box and of the strip behind the labels. `labelbg=off` switches the box off (Section 5.5).
`contour`	as `labelbg`	Color of the contour around the glyphs. `contour=off` switches the contour off (Section 5.5).
`diphmargin`	`0.15em`	Width of the contour around the label of a diphthong arrow.

Table 5: Variables determining the white space around vowel labels

```
\begin{vowelchart}[slanted] % with adjusted options
  \putvowel[l, labelbg=off, contour=off]{eː}{0.06}{0.8}
  \putvowel[x, boxh=.7em, boxd=2pt]{ɑː/ɑ}{2}{2.6}
\end{vowelchart}
```



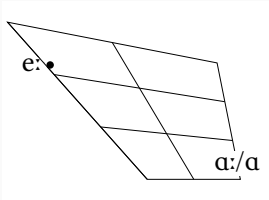
Remedies for typical problems:

- A line runs too close to a symbol, or remains between the parts of a symbol such as "(i̯)": Increase `pad`. Increase `margin` if the line touches the outline of the glyph.
- A diagonal line has a notch next to a symbol: The white box or the contour is too large for this position. Decrease `pad` and, if necessary, `margin`, or move the label with `\hspace` as described above. This happens most often in strongly slanted shapes.
- Descenders cut into a diagonal line, or a line runs through the descender: A positive `boxd` (e.g. `0.2ex`) clears the area below the baseline but also removes more of the line there.
- A bit of a horizontal line remains between a dot and its label, especially in slanted charts: Increase `stripe`. In slanted vowel charts the horizontal lines are slightly tilted, so a thicker strip is needed. Do not choose a thickness larger than necessary, because it also removes parts of neighboring lines.
- The label sits too close to or too far from its dot: Change `gap`.
- All labels sit too high or too low relative to their dots: Change `lift`.
- The central vowels ([ə], [ɐ], and central [a], and more generally any symbol placed with `[x]` at `h=1`) stand on the center line, which is nearly vertical and runs through the symbol. To keep the

line away from them, a white rectangle of size `zonew` × `zoneh` is drawn in the background color, centered on the position of the symbol, before the symbol is set. Increase `zonew` if a piece of the center line remains left or right of the symbol, and decrease `zoneh` if the rectangle removes too much of a horizontal line near the symbol. With `zoneh=0pt` the field is switched off. It is not drawn for symbols placed next to a dot (`[l]`, `[r]`) or for empty symbols.

Third, you can add TikZ commands, e.g. to draw a white rectangle over a line to make more space for a vowel label or to redraw a line that should not have been erased, as in the following example.

```
% With corrections using TikZ
\begin{vowelchart}[slanted]
  \putvowel[l]{eː}{0.06}{0.8}
  % Redraw line AFTER adding the vowel label for [eː]:
  \draw[line width=\vowelchartline] (vowel cs:h=0,v=0) --
  ↪  (vowel cs:h=0,v=3);
  % Make space BEFORE adding the vowel label for [aː/a]:
  \fill[vowelchartbg] (vowel cs:h=1.75,v=2.3) rectangle
  ↪  (vowel cs:h=2.25,v=2.9);
  \putvowel[x]{aː/a}{2}{2.6}
\end{vowelchart}
```



8 The mathematics behind it

This chapter is not needed to use the package. The formulas given below are only useful if (a) you want to know exactly what `\putvowel` does with its arguments (8.1), (b) you want to convert a diagram from `vowel` to the new relative coordinates and the positions of the vowels have to be accurate to the millimetre (8.2), or (c) you have to convert a large amount of old data (8.3).

8.1 How `vowelchart` turns relative into real coordinates

A chart is defined by four corners P_{tl}, P_{tr}, P_{bl} and P_{br} (top left, top right, bottom left, bottom right; see Section 3.2). Each corner is a pair (x, y), where y increases *downwards*. A position in the chart is given by the relative coordinates $h \in [0, 2]$ (from left to right) and $v \in [0, 3]$ (from top to bottom). For the calculation, it is more convenient to express both as fractions of the way across the chart. We call them s and t:

$$s = \frac{h}{2}, \qquad t = \frac{v}{3}.$$

So s is 0 at the left edge and 1 at the right edge, and t is 0 at the top edge and 1 at the bottom edge. A point with $h = 0.5$ and $v = 1$, for example, has $s = 0.25$ and $t = \frac{1}{3}$: it lies a quarter of the way from the left to the right and a third of the way from the top to the bottom.

The package then works in two steps. First, it determines the two points on the left and on the right edge of the chart at the height v, namely the points that are the fraction t of the way from the top corner to the bottom corner:

$$L = P_{\text{tl}} + t\,(P_{\text{bl}} - P_{\text{tl}}), \qquad R = P_{\text{tr}} + t\,(P_{\text{br}} - P_{\text{tr}}).$$

Second, it goes the fraction s of the way from L to R: $P = L + s\,(R - L)$. Written as one formula, this is

$$P(h, v) = (1 - s)(1 - t)\,P_{\text{tl}} + s(1 - t)\,P_{\text{tr}} + (1 - s)\,t\,P_{\text{bl}} + s\,t\,P_{\text{br}}.$$

Each corner is weighted by how close the point is to it. This is a bilinear interpolation between the four corners. This formula is used for all shapes, for the coordinate system `vowel cs` (Section 5.2) and for all vowel placement commands.

The corners are given in the unit of the chart. Its length u is derived from the option `width` or `height` (Section 3) and the extent of the four corners:

$$u = \frac{\text{width}}{x_{\max} - x_{\min}} \qquad \text{or} \qquad u = \frac{\text{height}}{y_{\max} - y_{\min}},$$

where $x_{\max}$, $x_{\min}$, $y_{\max}$, and $y_{\min}$ are the largest and smallest x and y values of the four corners. The point $P = (X, Y)$ is drawn at the TikZ position $(u \cdot X,\ -u \cdot Y)$; the minus sign is needed because TikZ's y axis points upwards.

Example. For the default shape `normal`, the corners are $(0, 0)$, $(4, 0)$, $(2, 3)$ and $(4, 3)$, so that

$$X = \frac{2v}{3} + 2h - \frac{hv}{3}, \qquad Y = v.$$

The vowel at $h = 0.5$, $v = 1$ is therefore at $X = \frac{2}{3} + 1 - \frac{1}{6} = 1.5$ and $Y = 1$. In a chart with `width=4cm` this is $u = 1$ cm, i.e. 1.5 cm to the right of and 1 cm below the top left corner.

8.2 From absolute `vowel` coordinates to relative `vowelchart` coordinates

In `vowel`, the position of a vowel is a pair of lengths (x, y), measured from the top left corner of the chart. To convert such a pair to the relative coordinates (h, v) of `\putvowel`, divide both lengths by the units of the old chart:

$$a = \frac{x}{\texttt{\textbackslash vowelhunit}}, \qquad b = \frac{y}{\texttt{\textbackslash vowelvunit}}.$$

If the old code gave the coordinates in multiples of the units (such as `1.5\vowelhunit` and `1\vowelvunit`), a and b are simply these numbers (here $a = 1.5$ and $b = 1$). Then the new coordinates are calculated with the formulas in the following table. The vertical coordinate v only depends on b. The horizontal coordinate h depends on a and, except for the rectangle, also on b, because the left and right edge of the chart are slanted. The formulas for `three` differ because `vowel` draws such a chart only two units of `\vowelvunit` high (and uses a correspondingly larger unit), so that the overall height stays the same.

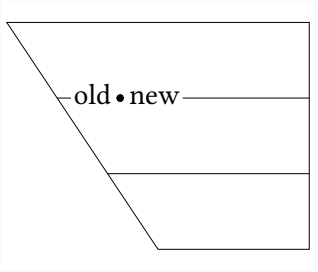
	four heights (default)	`three`
v for all shapes	b	$1.5\,b$
h for `normal`	$\dfrac{3a - 2b}{6 - b}$	$\dfrac{2a - 2b}{4 - b}$
h for `rectangle`	$\dfrac{a}{2}$	$\dfrac{a}{2}$
h for `triangle`	$\dfrac{3a - 2b}{6 - 2b}$	$\dfrac{a - b}{2 - b}$

For the triangle, the formulas are undefined at the apex (the bottom of the chart), where every h describes the same point. Set $h = 1$ in this case.

The formulas also describe the position correctly if the new chart is larger or smaller than the old one, because h and v only state where a point lies *within* the quadrilateral. But note that the new chart always keeps the proportions of its shape, whereas `vowel` allowed `\vowelhunit` and `\vowelvunit` to be set independently of each other. (If you want to reproduce this effect, you have to adjust the corners of the chart; see Section 3.2.)

Check. In a chart with four heights, the point $a = 1.5$, $b = 1$ gives $v = 1$ and, for the shape `normal`, $h = \frac{4.5-2}{5} = 0.5$. The following chart draws this point twice, once with the legacy command `\putvowel*` (Section 6.2) with absolute coordinates and once with `\putvowel` (Section 4.2) with relative coordinates; the two dots coincide:

```
\begin{vowelchart}[width=4cm, simple]
  \putvowel*[l]{old}{1.5\vowelhunit}{1\vowelvunit}
  \putvowel[r]{new}{0.5}{1}
\end{vowelchart}
```



8.3 Converting many coordinates in a spreadsheet

If a large amount of data has to be converted from `vowel` to `vowelchart`, a spreadsheet application such as Excel, LibreOffice Calc, Google Sheets, OnlyOffice Docs, or Numbers can be used:

1. Enter the old x coordinates in column A and the old y coordinates in column B, starting in row 2. They have to be plain numbers in multiples of `\vowelhunit` and `\vowelvunit`. If you have lengths (for example in centimetres), divide them by the values of `\vowelhunit` and `\vowelvunit` first.
2. Enter the formula for v from the following table in cell C2 and the formula for h for your shape in cell D2.
3. Copy both cells down to the last row. Column C then contains the new v values and column D the new h values, which you can use in `\putvowel`.

Cell	four heights (default)	`three`
C2 (v)	`=B2`	`=3/2*B2`
D2 (h), `normal`	`=(3*A2-2*B2)/(6-B2)`	`=(2*A2-2*B2)/(4-B2)`
D2 (h), `rectangle`	`=A2/2`	`=A2/2`
D2 (h), `triangle`	`=IF(B2>=3,1,(3*A2-2*B2)/(6-2*B2))`	`=IF(B2>=2,1,(A2-B2)/(2-B2))`

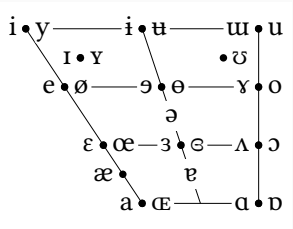
The formulas for the vertical coordinates and for the horizontal ones in `normal` and `rectangle` contain only the basic operations, so they work in every spreadsheet.

The formulas for the horizontal coordinate in `triangle` use the function `IF` to deal with the apex of the triangle (the bottom of the chart): there, every h describes the same point, and the formula would end with a division by zero. `IF` returns the value 1 (the middle of the chart) when $B2$ is at least 3 (four heights) or 2 (`three`), i.e. when the old y coordinate is the bottom end of the chart, and calculates h with the formula from the table otherwise. These formulas are written for spreadsheets in English. In other language versions, two things can differ. First, many versions translate the names of functions; `IF` is, for example, `WENN` in German, `SI` in French and Spanish, `SE` in Italian, `ALS` in Dutch, and `ЕСЛИ` in Russian (whereas other versions use only English function names, e.g. `IF` in Ukrainian Excel). Second, the arguments of a function can be separated by a semicolon instead of a comma. For example, the formula for the triangle with four heights in the German version of Excel reads `=WENN(B2>=3;1;(3*A2-2*B2)/(6-2*B2))`. Instead of translating the formula, you can also switch the program to English function names: in LibreOffice Calc with the option *Use English function names* in the formula options, in Google Sheets with the option *Always use English function names* in the spreadsheet settings. You might also simply accept a “division by zero” error and use `=(3*A2-2*B2)/(6-2*B2)` (for four heights) or `=(A2-B2)/(2-B2)` (for the option `three`).

9 Examples

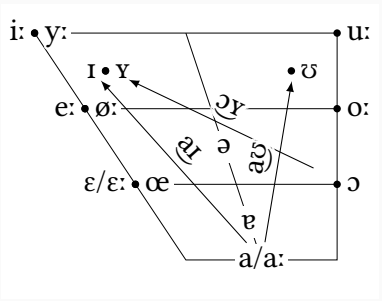
Full IPA vowel chart

```
\begin{vowelchart}
  \putipavowel{i} \putipavowel{y} \putipavowel{1}
  \putipavowel{u-} \putipavowel{M} \putipavowel{u}
  \putipavowel{I} \putipavowel{Y} \putipavowel{U}
  \putipavowel{e} \putipavowel{2} \putipavowel{@-}
  \putipavowel{8} \putipavowel{7} \putipavowel{o}
  \putipavowel{@} \putipavowel{E} \putipavowel{9}
  \putipavowel{3} \putipavowel{3-} \putipavowel{V}
  \putipavowel{O} \putipavowel{ae} \putipavowel{6}
  \putipavowel{a} \putipavowel{OE} \putipavowel{A}
  \putipavowel{Q}
\end{vowelchart}
```



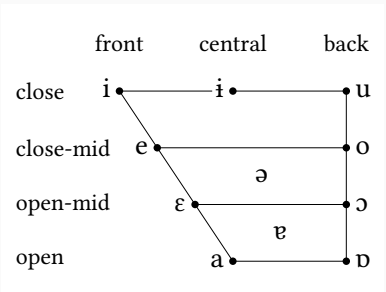
Standard German monophthongs and diphthongs

```
\begin{vowelchart}[width=4cm]
  \large
  \putipavowel[i:]{i} \putipavowel[y:]{y} \putipavowel[u:]{u}
  \putipavowel{I} \putipavowel{Y} \putipavowel{U}
  \putipavowel[e:]{e} \putipavowel[ø:]{ø} \putipavowel[o:]{o}
  \putipavowel[ɛ/ɛ:]{ɛ} \putipavowel{œ} \putipavowel{O}
  \putipavowel{@} \putipavowel{6}
  \putipavowel[a/a:]{'a'}
  \putdiphthong{ɔy}{1.8}{1.8}{0.45}{0.6}
  \putdiphthong{aʊ}{1.1}{2.8}{1.67}{0.6}
  \putdiphthong{aɪ}{0.9}{2.8}{0.25}{0.6}
\end{vowelchart}
```



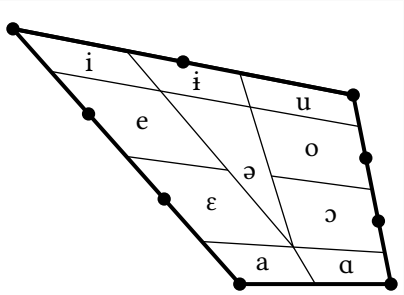
A chart with labels on the axes

```
\begin{vowelchart}[width=3cm, simple]
  \begin{scope}[font=\footnotesize]
    \node[anchor=south, yshift=4mm] at (vowel cs:h=0, v=0)
    ↪  {front};
    \node[anchor=south, yshift=4mm] at (vowel cs:h=1, v=0)
    ↪  {central};
    \node[anchor=south, yshift=4mm] at (vowel cs:h=2, v=0)
    ↪  {back};
    \node[anchor=west] at (-2, 0) {close};
    \node[anchor=west] at (-2, -1) {close-mid};
    \node[anchor=west] at (-2, -2) {open-mid};
    \node[anchor=west] at (-2, -3) {open};
  \end{scope}
  \putipavowel{i} \putipavowel{ɨ} \putipavowel{u}
  \putipavowel{e} \putipavowel{o} \putipavowel{ə}
  \putipavowel{ɛ} \putipavowel{ɔ} \putipavowel{ɐ}
  \putipavowel{a} \putipavowel{ɒ}
\end{vowelchart}
```



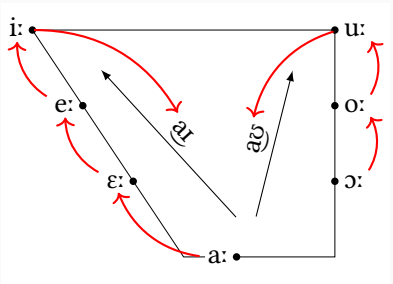
Vowel quadrilateral according to Daniel Jones with ten cardinal vowel zones

```
\begin{vowelchart}[slanted, plain, linewidth=1.5pt,
↪  dotsize=2.5pt, width=5cm]
  % Draw internal lines; the triangle erases the line drawn
  ↪  previously
  \draw[line width=0.5pt] (vowel cs:h=0, v=1.5) -- (vowel
  ↪  cs:h=2, v=1.5);
  \draw[line width=0.5pt, fill=vowelchartbg] (vowel
  ↪  cs:h=0.667, v=0) -- (vowel cs:h=1, v=2.5) -- (vowel
  ↪  cs:h=1.333, v=0) -- cycle;
  \draw[line width=0.5pt] (vowel cs:h=0, v=0.5) -- (vowel
  ↪  cs:h=2, v=0.5);
  \draw[line width=0.5pt] (vowel cs:h=0, v=2.5) -- (vowel
  ↪  cs:h=2, v=2.5);
  \draw[line width=0.5pt] (vowel cs:h=1, v=2.5) -- (vowel
  ↪  cs:h=1, v=3);
  % Place vowel symbols in the middle of the zones
  \putvowel{i}{0.333}{0.3} \putvowel{u}{1.667}{0.3}
  \putvowel{e}{0.388}{1} \putvowel{o}{1.611}{1}
  \putvowel{ɛ}{0.444}{2} \putvowel{ɔ}{1.556}{2}
  \putvowel{a}{0.5}{2.75} \putvowel{ɑ}{1.5}{2.75}
  \putvowel{ɨ}{1.001}{0.3} \putvowel{ə}{1.001}{1.5}
  % Place dots for cardinal vowels 1 to 9
  \putcvowel[x]{}{1} \putcvowel[x]{}{2} \putcvowel[x]{}{3}
  ↪  \putcvowel[x]{}{4} \putcvowel[x]{}{5}
  ↪  \putcvowel[x]{}{6} \putcvowel[x]{}{7}
  ↪  \putcvowel[x]{}{8} \putcvowel[x]{}{9}
  % Redraw top outer line
  \draw[line width=\vowelchartline] (vowel cs:h=0, v=0) --
  ↪  (vowel cs:h=2, v=0);
\end{vowelchart}
```



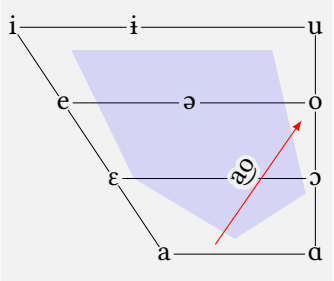
The Great Vowel Shift

```
\begin{vowelchart}[width=4cm, plain]
  \tikzset{GVC/.style={red, thick, ->, shorten >=2pt, shorten
  ↪  <=2pt}}
  \putvowel[l]{aː}{0.7}{3}[a] \putipavowel[ɛː]{ɛ}[E]
  \putipavowel[eː]{e}[e] \putipavowel[iː]{i}[i]
  \putdiphthong{a͡ɪ}{0.9}{2.5}{0.3}{0.5}[ai]
  \putipavowel[ɔː]{ɔ}[O] \putipavowel[oː]{o}[o]
  \putipavowel[uː]{u}[u]
  \putdiphthong{a͡ʊ}{1.1}{2.5}{1.7}{0.5}[au]
  \draw[GVC] (a.west) to[bend left=30] (E.south);
  \draw[GVC] (E.north west) to[bend left=30] (e.south);
  \draw[GVC] (e.north west) to[bend left=30] (i.south);
  \draw[GVC] (i.east) to[bend left=30] (ai.north west);
  \draw[GVC] (O.north east) to[bend right=30] (o.south east);
  \draw[GVC] (o.north east) to[bend right=30] (u.south east);
  \draw[GVC] (u.west) to[bend right=30] (au.north east);
\end{vowelchart}
```



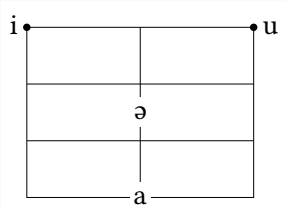
Shaded vowel space

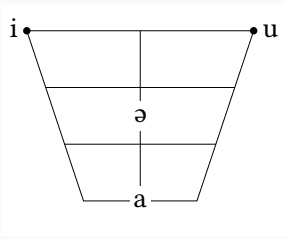
```
\begin{vowelchart}[width=4cm, simple, bg=black!5!white]
  \putcvowel{i}{1} \putcvowel{e}{2} \putcvowel{ɛ}{3}
  ↪  \putcvowel{a}{4}
  \putcvowel{ɑ}{5} \putcvowel{ɔ}{6} \putcvowel{o}{7}
  ↪  \putcvowel{u}{8}
  \fill[blue, opacity=.12]
  (vowel cs:h=0.3,v=0.3) -- (vowel cs:h=1.7,v=0.3) -- (vowel
  ↪  cs:h=1.9,v=2.2)
  -- (vowel cs:h=1,v=2.8) -- (vowel cs:h=0.2,v=2) -- cycle;
  {\vowelchartlabels{contour=blue!12!vowelchartbg,
  ↪  labelbg=blue!12!vowelchartbg}
    \putvowel{ə}{1}{1}
    \putdiphthong[red]{ao͡}{0.7}{2.9}{1.9}{1.2}
  }
  \putvowel[x]{ɨ}{0.8}{0}
\end{vowelchart}
```

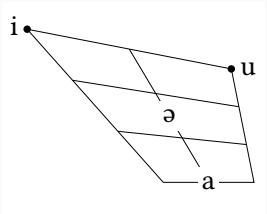


The same vowels in different shapes

```
\def\demo#1{%
  \begin{vowelchart}[#1, width=3cm]
    \putipavowel{i}
    \putipavowel{u}
    \putipavowel{'a'}
    \putipavowel{ə}
  \end{vowelchart}%
}
\demo{rectangle}\\[1em]
\demo{symmetric}\\[1em]
\demo{slanted}
```

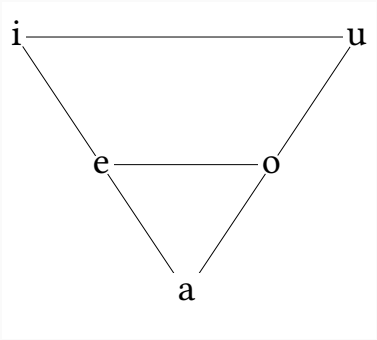






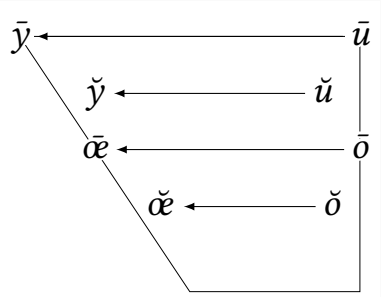
A three-height system on a triangular chart

```
\begin{vowelchart}[triangle, simple, three, width=4.5cm,
↪  font=\Large]
  \putvowel{i}{0}{0}   \putvowel{u}{2}{0}
  \putvowel{e}{0}{1.5} \putvowel{o}{2}{1.5}
  \putvowel{a}{1}{3}
\end{vowelchart}
```



Diphthong arrows used for illustrating sound change

```
\begin{vowelchart}[plain, width=4.5cm, font=\Large]
  \putvowel[x]{\textit{ū}}{2}{0}[ulong]
  \putvowel[x]{\textit{ŭ}}{1.75}{0.67}[ushort]
  \putvowel[x]{\textit{ō}}{2}{1.33}[olong]
  \putvowel[x]{\textit{ŏ}}{1.75}{2}[oshort]
  \putvowel[x]{\textit{ȳ}}{0}{0}[ylong]
  \putvowel[x]{\textit{y̆}}{0.25}{0.67}[yshort]
  \putvowel[x]{\textit{œ̄}}{0}{1.33}[oelong]
  \putvowel[x]{\textit{œ̆}}{0.25}{2}[oeshort]
  \putipadiphthong{}{(ulong.west)}{(ylong.east)}
  \putipadiphthong{}{(ushort.west)}{(yshort.east)}
  \putipadiphthong{}{(olong.west)}{(oelong.east)}
  \putipadiphthong{}{(oshort.west)}{(oeshort.east)}
\end{vowelchart}
```



Appendix: Short reference of commands and options

Syntax

```
\usepackage[<options>]{vowelchart}
\vowelchartset{<options>}
\vowelchartlabels{<label options>}

\begin{vowelchart}[<options>]
         \putvowel[<vowel options>]{<label>}{<h>}{<v>}[<node name>]
        \putvowel*[<vowel options>]{<label>}{<x>}{<y>}[<node name>]
        \putcvowel[<vowel options>]{<label>}{<number>}[<node name>]
      \putipavowel[<vowel options>][<label>]{<symbol>}[<node name>]
     \putdiphthong[<TikZ options>] {<label>}{<h1>}{<v1>}{<h2>}{<v2>}[<node name>]
  \putipadiphthong[<TikZ options>] {<label>}{<start>}{<end>}[<node name>]
\end{vowelchart}
```

The arguments in square brackets are optional.

- `<options>` are the chart options of the table below. `\vowelchartset` and the package options set defaults for all charts; options in the environment apply to one chart (Section 3.1).
- `<vowel options>` are `l`, `r` or `x` (position of the label relative to the dot) and all options marked *vowel* in the last column of the table. They apply to this vowel only.
- `\putipavowel` has two optional arguments before the symbol. If only one is given, it is the `<label>`; with two, the first one holds the `<vowel options>` and the second one the `<label>` (which can be empty for a dot without label).
- `<label options>` are `labelbg=` and `contour=`; they are set for the labels that follow, up to the end of the current group (Section 5.5).
- `<h>`, `<v>` are relative coordinates ($0 \ldots 2$ and $0 \ldots 3$, Section 4.2), `<x>`, `<y>` are lengths (legacy, Section 6.2), `<number>` is the number of a cardinal vowel (1–16, Section 4.3), `<symbol>` an IPA or X-SAMPA vowel symbol (Table 3).
- `<start>` and `<end>` are IPA symbols or TikZ coordinates in parentheses, such as `(vowel cs:h=1,v=2)` or `(name.west)` (Section 4.6).
- `<TikZ options>` are passed to the arrow; the options `labelbg=`, `contour=`, `diphmargin=` and `diphlabelpos=` can be given there as well.

Options

All options can be given for a single chart, with `\vowelchartset` or as package options (Section 3.1). The column *Single* shows which options can also be given in the options of an individual command: *vowel* stands for the vowel commands (`\put…vowel`) and *diphthong* for `\putdiphthong` and `\putipadiphthong` (in their TikZ options). An option marked — has to be set at least for the whole chart.

Option	Default	Effect	Command	Section
Shape				
`normal`, `rectangle`, `triangle`, `symmetric`, `slanted`	`normal`	predefined shapes (set all four corners)	—	3.2
`tl={x,y}`, `tr`, `bl`, `br`	from shape	individual corners	—	3.2
Grid lines and dots				
`plain`, `simple`, `ipanew`, `centralzone`, `standard`, `centralcol`	`ipanew`	line pattern	—	3.3
`three`, `four`	`four`	number of heights	—	3.3
`linewidth=`⟨dim⟩	`0.2pt`	line width	—	3.7
`dotsize=`⟨dim⟩	`1.4pt`	dot radius	—	3.7
Size				
`width=`⟨dim⟩	`8em`	exact width of the quadrilateral	—	3.4
`height=`⟨dim⟩	—	exact height of the quadrilateral	—	3.4
⟨dim⟩	—	shorthand for `width`	—	3.4
Colors				
`fg=`⟨color⟩	text color	foreground color	—	3.5
`bg=`⟨color⟩	`white`	background color	—	3.5
`labelbg=`⟨color|off⟩	`bg`	color of the areas cleared behind labels	vowel, diphthong	5.5
`contour=`⟨color|off⟩	as `labelbg`	color of the contour around labels	vowel, diphthong	5.5
Font				
`font=`⟨commands⟩	—	font commands for the chart	—	3.6
`fontsize=`⟨dim⟩	—	fixed font size	—	3.6
Position of a label (only in the options of a vowel command)				
`l`, `r`, `x`	`x`	label left of, right of, or centered on the dot	*only* vowel	4.1
White areas around labels				
`margin=`⟨dim⟩	`0.06em`	width of the contour	vowel	7
`pad=`⟨dim⟩	`0.10em`	extra width of the white box	vowel	7
`boxh=`, `boxd=`	`1ex`, `0pt`	height and depth of the white box	vowel	7
`gap=`⟨dim⟩	`0.18em`	distance between label and dot	vowel	7
`lift=`⟨dim⟩	`0.5ex`	height of the baseline below the dot	vowel	7
`stripe=`⟨dim⟩	`1.5pt`	thickness of the strip next to dots	vowel	7
`zonew=`, `zoneh=`	`0.7em`, `1em`	size of the field behind central vowels	vowel	7
Diphthongs				
`diphmargin=`⟨dim⟩	`0.15em`	width of the contour around diphthong labels	diphthong	7
`diphlabelpos=`⟨above|below|on⟩	`above`	position of diphthong labels	diphthong	4.5
`ipashorten=`⟨dim⟩	`3pt`	shortening of arrows ending at IPA positions	—	4.6