$G\sharp^{maj7}$

fr3

Chordbox

A LATEX package for drawing string instrument chord diagrams

Steven Franzen

v0.1   December 1, 2018
`https://github.com/sfranzen/chordbox`

Contents

1 Package introduction

This package is the result of a search similar to the one undertaken by Clemens Niederberger for his `leadsheets`[1] package: over the years I have collected many textual guitar tabs and chord sheets that I finally wanted to put together and typeset properly using LaTeX, including guitar chord diagrams. The first part of my requirements is now more than fulfilled by `leadsheets`, which provides all the tools for putting together chords and lyrics. For the second part I found that there were only two relevant existing packages: `guitarchordschemes`[2] and `gchords`.[3] Neither of these achieved exactly what I wanted, so this package is my own attempt at providing that tool. It is a short collection of Ti*k*Z code snippets, wrapped in two LaTeX commands that are easy to use, with several options available to customise their output.

1.1 Loading the package

`chordbox` currently does not support LaTeX options, so it is loaded simply by putting
    \usepackage{chordbox}
in your document's preamble.

1.2 License

This file is part of chordbox. Chordbox may be distributed and/or modified under the conditions of the LaTeX Project Public License, either version 1.3 of this license or (at your option) any later version. The latest version of this license is in `http://www.latex-project.org/lppl.txt` and version 1.3 or later is part of all distributions of LaTeX version 2005/12/01 or later.

Chordbox has the LPPL maintenance status 'maintained'. The Current Maintainer of chordbox is Steven Franzen.

2 Usage

The package provides two similar commands, `\chordbox` and `\bchordbox`. The latter extends the former to draw barred chords.

2.1 The `\chordbox` command

Syntax: `\chordbox`[⟨*base fret*⟩]{⟨*chord name*⟩}{⟨*fret positions*⟩}

The simplest form of the command requires you to specify only the ⟨*chord name*⟩ and ⟨*fret positions*⟩. The former can contain any text and math symbols (see below), the latter should be a comma-separated list of elements, one for each string. Each of these is interpreted as follows: a positive number as a fretted string, the number 0 as an open string and anything else as a muted string. It draws a grid resembling the (vertical) strings over four horizontal frets, with a black bar at the top figuring as the instrument's nut. The number of strings drawn is determined

---

[1]Niederberger, *leadsheets.*

[2]Niederberger, *guitarchordschemes.*

[3]Peeters, *gchords.*

by the number of ⟨*fret positions*⟩ passed to the command, so generating chord diagrams for different instruments is no problem:

Ukulele	5-string banjo	6-string guitar
\chordbox{Am}{2,0,0,0}	\chordbox{Am}{x,2,2,1,2}	\chordbox{Am}{x,0,2,2,1,0}

By default, the ⟨*chord name*⟩ is processed by a command that typesets it in math mode, in roman font (see /chordbox/name). This allows the use of math symbols like \flat and \sharp as well as the ^ (superscript) operator to typeset decorated chord names:

    \chordbox{B\flat m}{x,x,x,3,2,1}   \chordbox{F\sharp^{maj7}}{2,x,3,3,2,x}

For best results, make sure to select the same typeface for both text and math typesetting. Of course, if you are also using the leadsheets package, you could opt to use its \writechord command for typesetting chord names, see also subsection 3.2.

The ⟨*base fret*⟩ is the number that, although formally an optional argument, must be provided for chords extending past the number of frets in the chord box. It is used to position the box and fretted notes relative to this fret, for example \chordbox[6]{D\sharp m^{7\flat 5}}{x,6,7,6,7,x}:

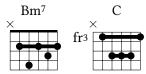As can be seen, the nut is not drawn in these cases.

2.2 The \bchordbox command

Syntax: \bchordbox[⟨*base fret*⟩]{⟨*chord name*⟩}{⟨*fret positions*⟩}{⟨*barre frets*⟩}

As mentioned, this command is for drawing barre chords, for which it requires an additional comma-separated list of fret numbers. A thick line is drawn over the string symbols at these frets. The other arguments are identical to those of \chordbox, see for example \bchordbox{Bm^7}{x,2,4,2,3,2}{2} and \bchordbox[3]{C}{x,3,5,5,5,3}{3,5}:

3 Settings

Because chordbox relies on PGF/TiKZ for drawing, it makes sense to use the powerful PGF key management system that comes with it. Therefore this package stores its code and style settings as keys, some of which may be modified to customise the output. This can be done anywhere in the preamble or document by means of the the \pgfset and \tikzset commands. Some examples of usage will be given below, but for more reading about PGF keys and their handling please refer to Section 82 of the PGF manual[4] where everything is documented in full detail. In any case, modifying a key stores its value for the LATEX group where the command is issued and its child groups, which can override it again, but it does not propagate up to parent groups.

3.1 TiKZ keys

The items described in this section are called *styles* in TiKZ terminology and contain various (lists of) keys and values that can be applied to various drawing commands. Most importantly, the chordbox style applies to the whole {tikzpicture} environment of every chord box produced. The default setting is to scale all coordinates by a quarter, because the drawing is done in PGF's "natural" units that, though convenient to use, result in an impractically large picture. Note that this scaling only influences the TiKZ components; in particular, fonts are not affected.

Because these keys are all stored in the /tikz path, it is most convenient to change them using the \tikzset command, for example \tikzset{chordbox/.style={scale=0.4, thick}}.

/tikz/chordbox   (style, initially {scale=0.25})
> This style is applied to every {tikzpicture} environment produced by this package.

The rest of the styles define the three symbol shapes used for the fretted (●), open (○) and muted (×) string positions.

/tikz/string/base (style, initially {circle, draw, inner sep=0, minimum size=20, transform shape})
> Common options for all three symbols. The transform shape option is required to correctly scale the symbols, which are implemented as TiKZ nodes, with the rest of the picture.

/tikz/string/fretted   (style, initially {string/base, fill})

/tikz/string/open   (style, initially {string/base, above})

/tikz/string/muted   (style, initially {string/open, cross out, minimum size=19})

3.2 Other keys

The following keys are not passed directly to TiKZ commands, but used to store global settings. They are all in the /chordbox path and should be set using the \pgfset command, for example \pgfset{/chordbox/name/.code=\writechord{#1}}.

---

[4]Tantau, *The TikZ and PGF Packages.*

`/chordbox/numfrets`=⟨*number*⟩   (initially 4)

The number of frets drawn in each chord box.

`/chordbox/name`   (initially `\ensuremath{\mathrm{#1}}`)

The code that is used to typeset the name of the chord.

References

Clemens Niederberger. *guitarchordschemes*. Version 0.7. Aug. 16, 2016. URL: `https://ctan.org/pkg/guitarchordschemes/`.

Clemens Niederberger. *leadsheets*. Version 0.5b. Sept. 26, 2017. URL: `https://ctan.org/pkg/leadsheets/`.

Kasper Peeters. *gchords*. Version 1.20. Feb. 3, 2008. URL: `https://ctan.org/pkg/gchords/`.

Till Tantau. *The TikZ and PGF Packages. Manual for version 3.0.1a*. Aug. 29, 2015. URL: `http://sourceforge.net/projects/pgf/`.