

The `ddphonism` package*

Celia Rubio Madrigal[†]

August 10, 2019

Abstract

This is a music-related package which is focused on notation from the Twelve-Tone System, also called Dodecaphonism. It provides \LaTeX algorithms that produce typical T.T.S. notation based off a musical series, or row sequence, of variable length.

Keywords

twelve tone system, dodecaphonism, music, mathematics, matrix, row, series, diagram, clock diagram, notation, algorithm, schoenberg, contemporary music, 20th century

Contents

1	Introduction	1
2	Using the <code>ddphonism</code> package	2
3	The package code	5

1 Introduction

There are hundreds of music tools and software online which are able to produce different music notations. However, I have never seen a \LaTeX tool that can do the same. This package is not only about notation, but it also calculates mathematically how this notation should work.

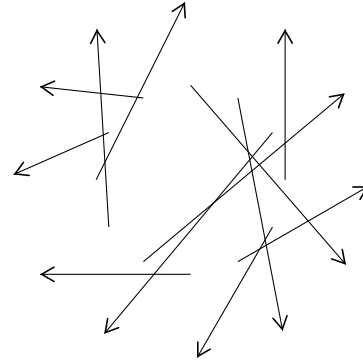
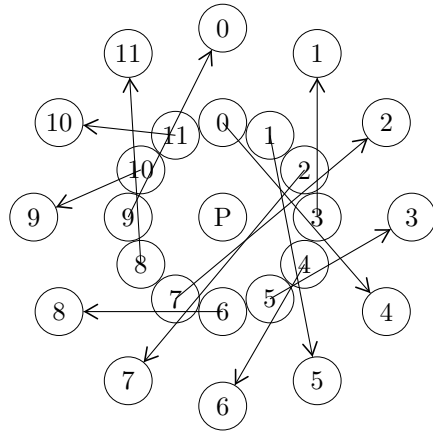
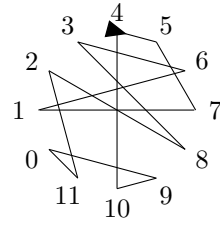
It is said that a twelve-tone matrix is the only thing a twelve-tone composer should need, because it provides the whole serial spectrum with which they may work. I wanted \LaTeX users to be able to generate them automatically.

But I also think that a twelve-tone matrix is not enough, that there exist several other notations with which they may understand their series and its potential. These are the diagrams that can be obtained with this package.

*This document corresponds to `ddphonism` v0.1, dated 2019/08/10.

[†]Email: celrubio@ucm.es

0	1	2	3	4
4	0	1	2	3
3	4	0	1	2
2	3	4	0	1
1	2	3	4	0



$$\begin{pmatrix} 0 & 1 & 2 & 3 & 4 \\ 4 & 3 & 2 & 1 & 0 \end{pmatrix}$$

2 Using the **ddphonism** package

These are the commands provided by **ddphonism**. The main parameter in every command is the row sequence.

`\dmatrix` produces a twelve-tone matrix of arbitrary length, as shown in this website. For example, `\dmatrix{0,2,1,4,3,6,5}` produces the matrix

0	2	1	4	3	6	5
5	0	6	2	1	4	3
6	1	0	3	2	5	4
3	5	4	0	6	2	1
4	6	5	1	0	3	2
1	3	2	5	4	0	6
2	4	3	6	5	1	0

The optional parameter **sep** scales the matrix. The optional parameters **vsep**, **hsep** scales the matrix vertically and horizontally.

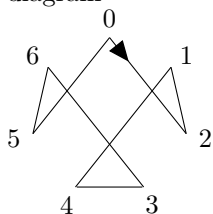
The optional parameter **lines** draws lines between rows and columns. The optional parameters **outside lines**, **inside lines** only draws the outside or inside lines. The optional parameters **vlines**, **hlines** only draws the vertical or horizontal lines.

`\dmatrix[lines,sep=0.75]{0,2,1,4,3,6,5}` produces the matrix

0	2	1	4	3	6	5
5	0	6	2	1	4	3
6	1	0	3	2	5	4
3	5	4	0	6	2	1
4	6	5	1	0	3	2
1	3	2	5	4	0	6
2	4	3	6	5	1	0

The optional parameter `no tikz` deletes the `tikz` environment and lets the user write it instead.

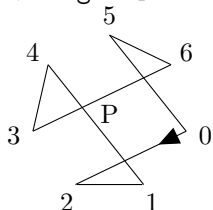
`\ddiagram` produces a twelve tone clock diagram of arbitrary length, as shown in this website. For example, `\ddiagram{0,2,1,4,3,6,5}` produces the diagram



The optional parameter `name` lets the user write a name at the center of the diagram.

The optional parameter `up` lets the user choose which number is up north. The default value is the first number in the row.

`\ddiagram[name=P, up=5]{0,2,1,4,3,6,5}` produces the diagram

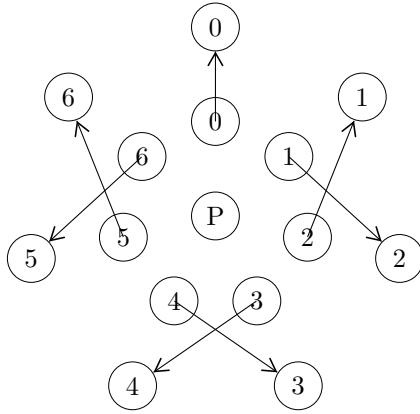


The optional parameter `no tikz` deletes the `tikz` environment and lets the user write it instead. The option `up` does not work anymore and the `up` position becomes 0. It is recommended that the user passes the option `ddiagram` to the environment:

```
\begin{tikzpicture}[ddiagram]
\ddiagram[no tikz]{0,2,1,4,3,6,5}
\end{tikzpicture}
```

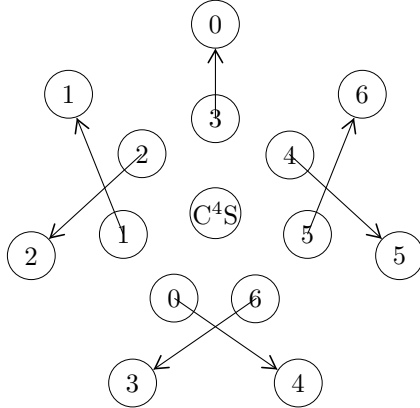
produces the same diagram as `\ddiagram{0,2,1,4,3,6,5}`.

`\ddihedral` produces a dihedral representation of a series of arbitrary length. For example, `\ddihedral{0,2,1,4,3,6,5}` produces the diagram



The optional parameters `t`, `s`, `c`, `v` let the user apply transformations to the diagram: *transposition*, *inversion*, *cyclic shift* and *retrograde*, in that order.

`\ddihedral[s=1, c=4]{0,2,1,4,3,6,5}` produces the diagram

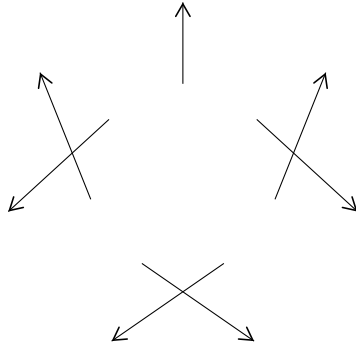


The optional parameter `no tikz` deletes the `tikz` environment and lets the user write it instead. It is recommended that the user passes the option `ddihedral` to the environment:

```
\begin{tikzpicture}[ddihedral]
\ddihedral[no tikz]{0,2,1,4,3,6,5}
\end{tikzpicture}
```

produces the same diagram as `\ddihedral{0,2,1,4,3,6,5}`.

`\darrows` produces the arrows from the `\ddihedral` diagram. For example, `\darrows{0,2,1,4,3,6,5}` produces the arrows



The optional parameter `no tikz` deletes the `tikz` environment and lets the user write it instead.

`\drow` produces a twelve-tone row sequence as a permutation in its matrix form. For example, `\drow{0,2,1,4,3,6,5}` produces the row

$$\begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 \\ 0 & 2 & 1 & 4 & 3 & 6 & 5 \end{pmatrix}$$

The optional parameter `sep` lets the user choose the column separation.

`\drow[sep=10pt]{0,2,1,4,3,6,5}` produces the row

$$\begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 \\ 0 & 2 & 1 & 4 & 3 & 6 & 5 \end{pmatrix}$$

3 The package code

```

1  % ddphonism
2  %
3  % (c) Celia Rubio Madrigal
4  %
5  %% This program can be redistributed and/or modified under the terms
6  %% of the LaTeX Project Public License Distributed from CTAN archives
7  %% in directory macros/latex/base/lppl.txt.
8
9  \NeedsTeXFormat{LaTeX2e}
10 \ProvidesPackage{ddphonism}
11 [2019/08/10 v0.1 LaTeX package for twelve-tone matrices, clock diagrams et al.]
12
13 \RequirePackage{etoolbox}
14 \RequirePackage{xparse}
15 \RequirePackage{tikz}
16 \RequirePackage{xstring}
17 \RequirePackage{pgfkeys}
18
19 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
20 % Matrices

```

```

23 \usetikzlibrary {matrix}

25 \ExplSyntaxOn
\DeclareExpandableDocumentCommand{\Evaluation}{m}{\int_eval:n {#1}}
27 \ExplSyntaxOff

29 \newcounter{Dsize}
\newcommand{\DsizeMake}[1]{%
31 \setcounter{Dsize}{0}%
\foreach \n in {#1}{%
33 \stepcounter{Dsize}%
} %
35 }

37 % Only with numbers.
\newcounter{Dfirst}
39 \newcommand{\DheadMake}[1]{%
\setcounter{Dfirst}{-1}%
41 \foreach \n in {#1}{%
\ifnum\theDfirst=-1%
43 \setcounter{Dfirst}{\n}%
\fi %
45 } %
}

47 % Only when DsizeMake is already done.
\newcounter{Dmod}
\newcommand{\Modulo}[1]{%
51 \setcounter{Dmod}{#1}
\loop%
53 \ifnum\theDmod>\Evaluation{\theDsize-1}%
\setcounter{Dmod}{\Evaluation{\theDmod-\theDsize}}%
55 \repeat%
\ifnum\theDmod<0%
57 \setcounter{Dmod}{\Evaluation{\theDmod+\theDsize}}%
\repeat%
59 \theDmod%
}

61 \newif\ifdmatrixLines
63 \newif\ifdmatrixOutside
\newif\ifdmatrixInside
65 \newif\ifdmatrixV
\newif\ifdmatrixH
67 \newif\ifdmatrixTikz
\pgfkeys{
69 /dmatrix/.is family
, /dmatrix
71 , default/.style =
{ lines = false
73 , outside lines = false
, inside lines = false
75 , sep = 1
, vsep = 1
77 , hsep = 1
, no tikz = false
79 }

```

```

, no tikz/.is if=dmatrixTikz
81 , lines/.is if=dmatrixLines
, outside lines/.is if=dmatrixOutside
83 , inside lines/.is if=dmatrixInside
, vlines/.is if=dmatrixV
85 , hlines/.is if=dmatrixH
, sep/.estore in=\dmatrixSep
87 , vsep/.estore in=\dmatrixVsep
, hsep/.estore in=\dmatrixHsep
89 }

91 \newcommand{\DLOH}{%
\draw (0.05*\dmatrixSep*\dmatrixHsep,0) --%
93 (\theDsize*\dmatrixSep*\dmatrixHsep+0.05*\dmatrixSep*\dmatrixHsep,0);%
\draw (0.05*\dmatrixSep*\dmatrixHsep,-\theDsize*0.5*\dmatrixSep*\dmatrixVsep) -- %
95 (\theDsize*\dmatrixSep*\dmatrixHsep+0.05*\dmatrixSep*\dmatrixHsep,-\theDsize*0.5*\dmatrixSep*\dmatrixVsep);%
}

97 \newcommand{\DLOV}{%
99 \draw (0.05*\dmatrixSep*\dmatrixHsep,0) -- %
(0.05*\dmatrixSep*\dmatrixHsep,-\theDsize*0.5*\dmatrixSep*\dmatrixVsep);%
101 \draw (\theDsize*\dmatrixSep*\dmatrixHsep+0.05*\dmatrixSep*\dmatrixHsep,0) -- %
(\theDsize*\dmatrixSep*\dmatrixHsep+0.05*\dmatrixSep*\dmatrixHsep,-\theDsize*0.5*\dmatrixSep*\dmatrixVsep);
103 }

105 \newcommand{\DLIH}{%
\draw (0.05*\dmatrixSep*\dmatrixHsep,-\xD*0.5*\dmatrixSep*\dmatrixVsep) -- %
107 (\theDsize*\dmatrixSep*\dmatrixHsep+0.05*\dmatrixSep*\dmatrixHsep,-\xD*0.5*\dmatrixSep*\dmatrixVsep);%
}

109 \newcommand{\DLIV}{%
111 \draw (\xD*\dmatrixSep*\dmatrixHsep+0.05*\dmatrixSep*\dmatrixHsep,0) -- %
(\xD*\dmatrixSep*\dmatrixHsep+0.05*\dmatrixSep*\dmatrixHsep,-\theDsize*0.5*\dmatrixSep*\dmatrixVsep);%
113 }

115 \newcommand{\dmatrix}[2][]{%
\DsizeMake{#2}%
117 \DheadMake{#2}%
%
119 \pgfkeys{/dmatrix, default, #1}%
%
121 \ifdmatrixTikz\else%
\begin{tikzpicture}%
123 \fi%
\foreach [count=\nj] \j in {#2} {%
125 \foreach [count=\ni] \i in {#2} {%
\draw node at
127 ( \ni*\dmatrixSep*\dmatrixHsep-0.5*\dmatrixSep*\dmatrixHsep
, -\nj*\dmatrixSep*\dmatrixVsep/2+0.25*\dmatrixSep*\dmatrixVsep) {%
129 \Modulo{\Evaluation{\i-\j+\theDfirst}}}%
};%
131 }%
}%
133 \foreach \xD in {1,...,\Evaluation{\theDsize-1}} {%
\ifdmatrixLines
135 \DLOH\DLOV\DLIH\DLIV
\fi

```

```

137         \ifdmatrixOutside
138         \DLOH\DLOV
139         \fi
140         \ifdmatrixInside
141         \DLIH\DLIV
142         \fi
143         \ifdmatrixH
144         \DLOH\DLIH
145         \fi
146         \ifdmatrixV
147         \DLOV\DLIV
148         \fi
149     }%
150 %
151 \ifdmatrixTikz\else%
152 \end{tikzpicture}%
153 \fi%
154 }
155
156 %%%%%%%%%%%
157 % Diagrams
158
159 \usetikzlibrary {shapes,arrows,decorations.markings,shapes.misc}
160
161 \tikzstyle ddiagramArrow=[decoration=
162     {markings,mark=at position 0.25 with
163         {\arrow[scale=1.25,>=triangle 45]{>}}},
164     postaction={decorate}]
165
166 \tikzstyle {ddiagram}=[minimum height=0pt,inner sep=0pt,outer sep=0pt,scale=0.65]
167
168 \newif\ifddiagramTikz
169 \pgfkeys{
170     /ddiagram/.is family
171     , /ddiagram
172     , default/.style =
173         { name =\empty%
174           , up =\empty%
175           , no tikz = false
176         }
177     , no tikz/.is if=ddiagramTikz
178     , name/.estore in=\ddiagramName
179     , up/.estore in=\ddiagramUp
180 }
181
182 \newcounter{Dprev}
183 \newcommand{\Dvar}{\theDprev}
184 \newcommand{\ddiagram}[2][\empty%
185     \DsizeMake{#2}%
186     \DheadMake{#2}%
187     %
188     \pgfkeys{/ddiagram, default, #1}%
189     %
190     \ifdefequal{\ddiagramUp}{\empty}%
191     {\renewcommand{\Dvar}{\theDfirst}}% if empty
192     {\renewcommand{\Dvar}{\ddiagramUp}}% if not empty

```



```

%
195 \ifddiagramTikz\else%
\begin{tikzpicture}[ddiagram,rotate=360*\Dvar/\theDsize]%
197 \fi%
\foreach \x in {0,...,\Evaluation{\theDsize-1}} {%
199 \node at (90-360*\x/\theDsize:2) {\x};%
\node (\x) at (90-360*\x/\theDsize:1.6) {};%
201 };%
%
203 \setcounter{Dprev}{-1}%
\foreach \x in {#2}{%
205 \ifnum \theDprev=\theDfirst%
\draw [style=ddiagramArrow] (\theDprev) -- (\x);%
207 \else \ifnum \theDprev=-1%
\else%
209 \draw (\theDprev) -- (\x);%
\fi\fi%
211 \setcounter{Dprev}{\x}%
};%
\draw (\theDprev) -- (\theDfirst);%
%
215 \ifdefequal{\ddiagramName}{\empty}%
{}% if empty
217 {\node at (0,0) [circle, fill=white] {\ddiagramName};}% if not empty
\ifddiagramTikz\else%
219 \end{tikzpicture}%
\fi%
221 }

223 %%%%%%%%%%%
225 % Dihedral diagrams

227 \tikzstyle ddiagonalArrow=[decoration=
{markings,mark=at position 1 with {\arrow[scale=1.5,>=angle 60]{>}}},
229 postaction={decorate}]

231 \tikzstyle {ddihedral}=[inner sep=0,minimum height=18pt]

233 \newif\ifddihedralTikz
\pgfkeys{
235 /ddihedral/.is family, /ddihedral,
default/.style = {t = 0, c = 0, s = 0, v = 0, no tikz=false},
237 no tikz/.is if=ddihedralTikz,
t/.estore in = \ddihedralT,
239 c/.estore in = \ddihedralC,
s/.estore in = \ddihedralS,
241 v/.estore in = \ddihedralV,
}

243 \newif\ifdarrowsTikz
245 \pgfkeys{
/darrows/.is family, /darrows,
247 default/.style = {no tikz=false},
no tikz/.is if=darrowsTikz,
249 }
\newcommand{\darrows}[2][]{%

```

```

251 \DsizeMake{#2}%
252 %
253 \pgfkeys{/darrows, default, #1}%
254 %
255 \ifdarrowsTikz\else%
256 \begin{tikzpicture}%
257 \fi%
258 \draw foreach \x in {0,...,\Evaluation{\theDsize-1}} {%
259 (90-360*\x/\theDsize:2.5) node[circle] (\x) {}%
260 };%
261 \foreach \x [count=\y] in {#2} {%
262 \draw [style=ddihedralArrow] (90-360*\Evaluation{\y-1}/\theDsize:1.25) -- (\x);%
263 };%
264 \ifdarrowsTikz\else%
265 \end{tikzpicture}%
266 \fi%
267 }

268 \newcommand\ddihedral[2][]{%
269 \DsizeMake{#2}%
270 %
271 \pgfkeys{/ddihedral, default, #1}%
272 %
273 \ifddihedralTikz\else%
274 \begin{tikzpicture}[ddihedral]%
275 \fi%
276 \draw foreach \x in {0,...,\Evaluation{\theDsize-1}} {%
277 (\Evaluation{(90+\ddihedralT*360/\theDsize)+(2*\ddihedralS-1)*\x*360/\theDsize}:2.5)%
278 node[very thin, circle, draw] (\x) {\x}%
279 };%
280 %
281 \draw foreach \x in {0,...,\Evaluation{\theDsize-1}} {%
282 (\Evaluation{(90-\ddihedralC*360/\theDsize)+(2*\ddihedralV-1)*\x*360/\theDsize}:1.25)%
283 node[very thin, circle, draw] {\x}%
284 };%
285 %
286 \darrows[no tikz]{#2}%
287 %
288 \node at (0,0) [very thin, draw, circle, fill=white] {%
289 \ifnum\ddihedralV=0%
290 \ifnum\ddihedralC=0%
291 \ifnum\ddihedralS=0%
292 \ifnum\ddihedralT=0%
293 P%
294 \fi\fi\fi%
295 \else V\fi%
296 \ifnum\ddihedralC=0%
297 \else C^{\ddihedralC}\fi%
298 \ifnum\ddihedralS=0%
299 \else S\fi%
300 \ifnum\ddihedralT=0%
301 \else T^{\ddihedralT}\fi%
302 };%
303 \ifddihedralTikz\else%
304 \end{tikzpicture}%
305 \fi%
306 }

```

```

309 %%%%%%%%%%%
311 % Rows

313 \pgfkeys{
      /drow/.is family, /drow,
315     default/.style = {sep=\arraycolsep},
      sep/.estore in = \drowSep,
317 }

319 \long\def\addto#1#2{\expandafter\def\expandafter#1\expandafter{#1#2}}
\newcounter{myDDcntr}
321 \newlength{\Dvarr}

323 \newcommand{\drow}[2][{}]{%
      \DsizeMake{#2}%
325   %
      \pgfkeys{/drow, default, #1}%
327   \setlength{\Dvarr}{\arraycolsep}
      \setlength{\arraycolsep}{\drowSep}
329   %
      \ifnum\theDsize=0%
331     \ensuremath{\left(\right)}%
      \else\ifnum\theDsize=1%
333     \ensuremath{%
          \left(\begin{array}{*{\theDsize}c}%
335             0\,%
              #2\,%
337             \end{array}\right)%
          }%
339     \else%
      \def\TableDDdata{}%
341     \setcounter{myDDcntr}{0}%
      \loop%
343     \addto\TableDDdata{\themyDDcntr\stepcounter{myDDcntr} &}%
      \stepcounter{myDDcntr}%
345     \ifnum\themyDDcntr<\Evaluation{\theDsize-1}%
      \repeat%
347     \addto\TableDDdata{\themyDDcntr \\}%
      \setcounter{myDDcntr}{0}%
349     %
      \ensuremath{%
          \left(\begin{array}{*{\theDsize}c}%
351             \TableDDdata%
353             \StrSubstitute{#2}{,}{&}\,%
              \end{array}\right)%
          }%
355     \fi\fi%
357     \setlength{\arraycolsep}{\Dvarr}
}

359 \endinput

361

363 %% End of file 'ddphonism.sty'.

```