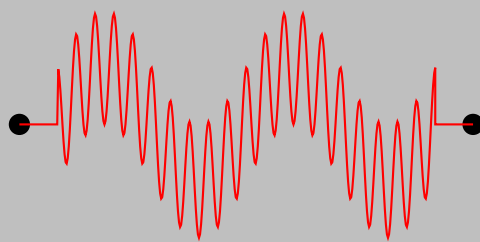


PSTricks

`pst-coil` **Special node connections** v.1.06

September 18, 2011



Documentation by
Herbert Voß

Package author(s):
Timothy Van Zandt
Herbert Voß
Michael Sharpe

Coils, springs and zigzag curves do not really count to the ordinary when creating graphics. They are of some importance though when dealing with the creation of coilish connections. They can also be used for node connections which will be shown at the end of this chapter.

The parameters especially for the coil variants are only easy to understand when keeping in mind that three dimensional helixes are dealt with, which are only projected onto the two dimensional paper plane. This will be elaborated on at the corresponding places.

Thanks to: Marco Daniel, Denis Girou, Uwe Siart.

Contents

| | | |
|----------|--|-----------|
| 1 | Parameters | 4 |
| 1.1 | coilwidth | 4 |
| 1.2 | coilheight | 5 |
| 1.3 | coilarm, coilarmA and coilarmB | 6 |
| 1.4 | coilaspect | 7 |
| 1.5 | coilinc | 7 |
| 1.6 | bow | 8 |
| 1.7 | ppoints | 8 |
| 1.8 | periods | 8 |
| 1.9 | amplitude | 9 |
| 1.10 | function | 9 |
| 2 | Makros | 10 |
| 2.1 | \pscoil | 10 |
| 2.2 | \psCoil | 11 |
| 2.3 | \pszigzag | 11 |
| 3 | Node connections | 11 |
| 4 | List of all optional arguments for pst-coil | 13 |
| | References | 13 |

1 Parameters

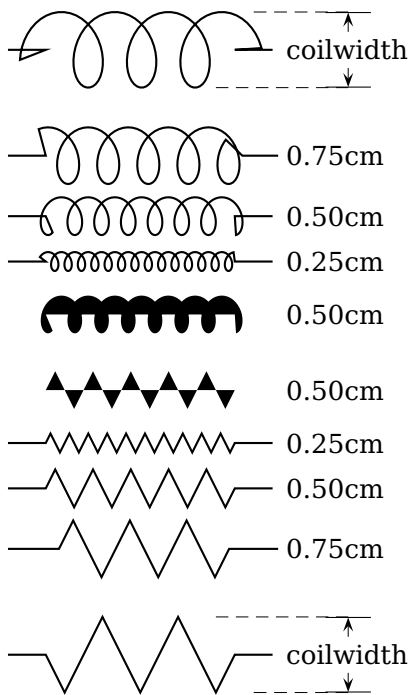
Table 1 shows a compilation of the special parameters valid for `pst-coil`.

Table 1: Compilation of all parameters for `pst-coil`

| name | value | default |
|-------------------------|---------------------------------------|---------|
| <code>coilwidth</code> | <code><value[unit]></code> | 1cm |
| <code>coilheight</code> | <code><value></code> | 1 |
| <code>coilarm</code> | <code><value[unit]></code> | 0.5cm |
| <code>coilarmA</code> | <code><value[unit]></code> | 0.5cm |
| <code>coilarmB</code> | <code><value[unit]></code> | 0.5cm |
| <code>coilaspect</code> | <code><angle></code> | 45 |
| <code>coilinc</code> | <code><angle></code> | 45 |
| <code>periods</code> | <code><value></code> | 1 |
| <code>amplitude</code> | <code><value></code> | 1 |
| <code>function</code> | <code><PS code></code> | sin |
| <code>ppoints</code> | <code><dots per periods></code> | 360 |
| <code>bow</code> | <code><value[unit]></code> | 0cm |

1.1 `coilwidth`

`coilwidth` denotes the diameter of the coil resp. the height of a zigzag line. The diameter corresponds to the height with a perpendicular parallel projection transverse to the length.



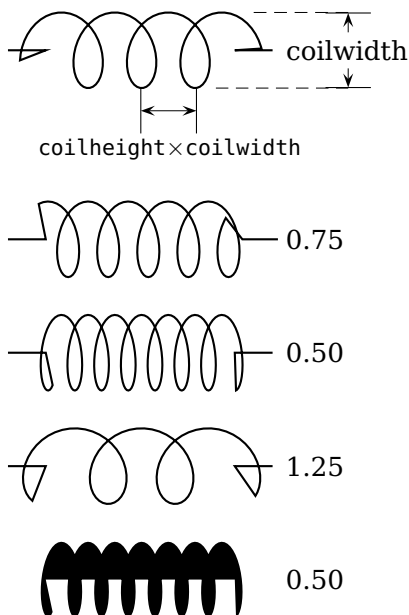
```

1 \begin{pspicture}(0,-5)(5,5)
2 \pscoil(0,4)(3.5,4)
3 {\psset{linewidth=0.2pt}
4 \psline[arrowscale=2,tbarsize=3mm
5 ]{|<->} (4.5,3.5)(4.5,4.5)
6 \psline[linestyle=dashed](3.25,4.5)(4.5,4.5)
7 \psline[linestyle=dashed](2.75,3.5)(4.5,3.5)}
8 \uput*[0](3.5,4){coilwidth}
9 \pscoil[coilwidth=0.75cm](0,2.6)(3.6,2.6)
10 \uput*[0](3.5,2.6){0.75cm}
11 \pscoil[coilwidth=0.5cm](0,1.8)(3.5,1.8)
12 \uput*[0](3.5,1.8){0.50cm}
13 \pscoil[coilwidth=0.25cm](0,1.2)(3.5,1.2)
14 \uput*[0](3.5,1.2){0.25cm}
15 \pscoil*[coilwidth=0.5cm](0,0.5)(3.5,0.5)
16 \uput*[0](3.5,0.5){0.50cm}
17 % zigzag
18 \pszigzag(0,-4)(3.5,-4)
19 [ ... ]

```

1.2 coilheight

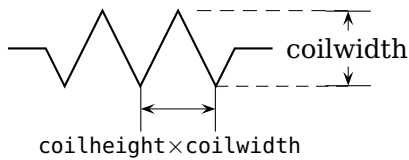
coilheight is in contrast to coilwidth no length but only a factor, wheratt the distance between two windings resp. jags is composed as follows: $dx = coilheight \cdot coilwidth$



```

1 \begin{pspicture}(0,-3)(5,4)
2 \pscoil(0,4)(3.5,4)
3 {\psset{linewidth=0.2pt}
4 \psline[arrowscale=2,tbarsize=3mm
5 ]{|<->} (4.5,3.5)(4.5,4.5)
6 \psline[linestyle=dashed](3.25,4.5)(4.5,4.5)
7 \psline[linestyle=dashed](2.75,3.5)(4.5,3.5)
8 \psline[arrowscale=2,tbarsize=3mm
9 ]{|<->} (1.75,3.2)(2.5,3.2)
10 \uput[-90](2.15,3){\footnotesize\texttt{
11 coilheight$\times$coilwidth}}
12 \uput*[0](3.5,4){coilwidth}
13 \pscoil[coilheight=0.75](0,1.5)(3.6,1.5)
14 \uput*[0](3.5,1.5){0.75}
15 \pscoil[coilheight=0.5](0,0)(3.5,0)
16 \uput*[0](3.5,0){0.50}
17 \pscoil[coilheight=1.25](0,-1.5)(3.5,-1.5)
18 \uput*[0](3.5,-1.5){1.25}
19 \pscoil*[coilheight=0.5](0,-3)(3.5,-3)
20 \uput*[0](3.5,-3){0.50}
21 \end{pspicture}

```



```

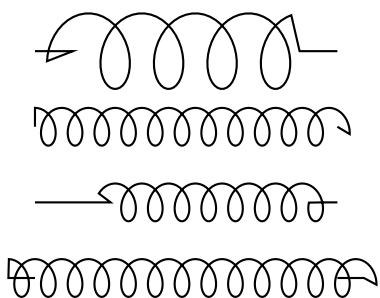
1 \begin{pspicture}[shift=2cm](0,-3)(5,4)
2 \pszigzag(0,4)(3.5,4)
3 { \psset{linewidth=0.2pt}
4 \psline[arrowscale=2,tbarsize=3mm
5 ]{|<->}(4.5,3.5)(4.5,4.5)
6 \psline[linestyle=dashed](2.5,4.5)(4.5,4.5)
7 \psline[linestyle=dashed](2.75,3.5)(4.5,3.5)
8 \psline[arrowscale=2,tbarsize=3mm
9 ]{|<->}(1.75,3.2)(2.75,3.2)
10 \uput[-90](2.15,3){\footnotesize\texttt{
11 coilheight$\times$coilwidth}} }
12 \uput*[0](3.5,4){coilwidth}
13 \pszigzag[coilheight=0.75](0,1.5)(3.6,1.5)
14 \uput*[0](3.5,1.5){0.75}
15 \pszigzag[coilheight=0.5](0,0)(3.5,0)
16 \uput*[0](3.5,0){0.50}
17 \pszigzag[coilheight=1.25](0,-1.5)(3.5,-1.5)
18 \uput*[0](3.5,-1.5){1.25}
19 \pszigzag*[coilheight=0.5](0,-3)(3.5,-3)
20 \uput*[0](3.5,-3){0.50}
21 \end{pspicture}

```

As can be seen in the above example, not the same physical distance is yielded. This is caused by the internal three dimensional representation of the coil; it is not seen under an angle of 90° but 45° . (Section 1.4)

1.3 coilarm, coilarmA and coilarmB

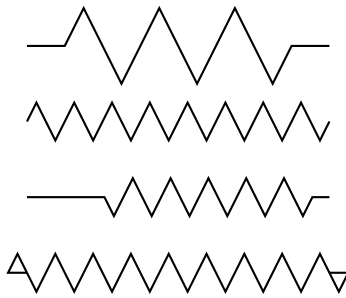
coilarm, coilarmA, and coilarmB denotes the part of a straight line made left and right. Negative values are possible, but do not make a lot of sense as a rule. As can be gathered from the example, the coil is lengthened by a negative value in this case, so that the part of a straight line is led back to the specified starting point resp. endpoint and consequently has a negative "direction". This parameter is not available for \psCoil.



```

1 \begin{pspicture}(4,4.5)
2 \pscoil(0,4)(4,4)
3 \psset{coilwidth=0.5}
4 \pscoil[coilarm=0](0,3)(4,3)
5 \pscoil[coilarmA=1cm,coilarmB=0.2cm](0,2)(4,2)
6 \pscoil[coilarm=-10pt](0,1)(4,1)
7 \end{pspicture}

```



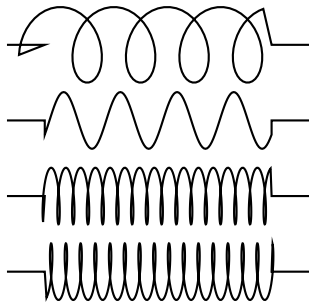
```

1 \begin{pspicture}(4,4.5)
2 \pszigzag(0,4)(4,4)
3 \psset{coilwidth=0.5}
4 \pszigzag[coilarm=0](0,3)(4,3)
5 \pszigzag[coilarmA=1cm,coilarmB=0.2cm](0,2)(4,2)
6 \pszigzag[coilarm=-10pt](0,1)(4,1)
7 \end{pspicture}

```

1.4 coilaspect

In Section 1.2 the three dimensional representation of the coil has been pointed out already. If it was regarded right-angled to its axis, the windings would not be recognizable. With `coilaspect` this “perspective” can be influenced, which is only possible for the “coil” variants of course. `coilaspect=0` yields a sine curve.



```

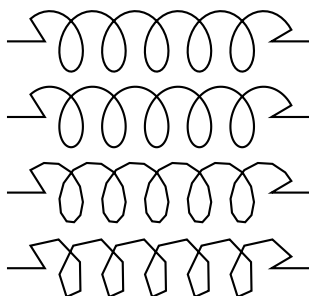
1 \begin{pspicture}(4,4.5)
2 \pscoil(0,4)(4,4)
3 \psset{coilwidth=0.75}
4 \pscoil[coilaspect=0](0,3)(4,3)
5 \pscoil[coilaspect=30,coilheight=0.3](0,2)(4,2)
6 \pscoil[coilaspect=-30,coilheight=0.3](0,1)(4,1)
7 \end{pspicture}

```

1.5 coilinc

The curve is drawn with the `lineto` procedure of PostScript, whereat `coilinc` specifies the rotation angle in the angular measure at which the next point is calculated. It has already been mentioned that the calculation is done as a three dimensional coil and is projected onto the two dimensional plane only in the end.

`coilinc` does also not make sense for zigzag lines, so that this parameter again only is available for the coil variant. Large angles result in a polygon line, small angles in harmonic progressions with increased calculation effort.



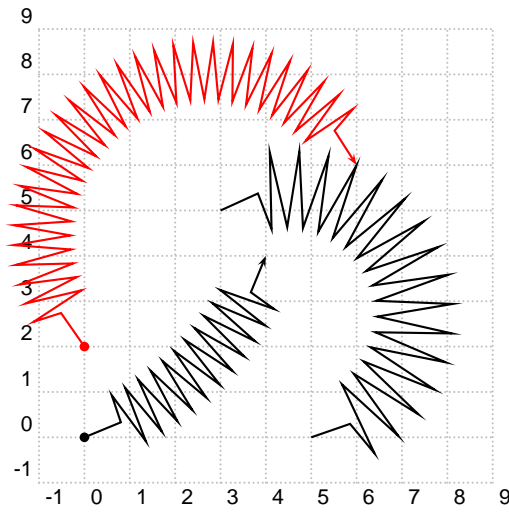
```

1 \begin{pspicture}(4,4.5)
2 \psset{coilwidth=0.8}
3 \pscoil(0,4)(4,4)
4 \psset{coilinc=0}
5 \pscoil[coilinc=5](0,3)(4,3)
6 \pscoil[coilinc=30](0,2)(4,2)
7 \pscoil[coilinc=60](0,1)(4,1)
8 \end{pspicture}

```

1.6 bow

The default value for bow is 0pt which leads to the shown direct connections between the two given point. With a positive or negative value circular zigzags are possible. This option is not available for coils.



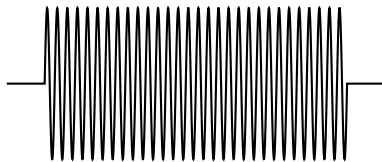
```

1 \psset{unit=0.6}
2 \begin{pspicture}[showgrid,shift=2cm](-1,-1)
   (9,9)
3 \pszigzag[bow=-0.4cm,coilheight=.3,
4   coilwidth=.75cm]{*->}(0,0)(4,4)
5 \pczigzag[linecolor=red,bow=2.5cm,
6   coilheight=0.3,
7   coilwidth=.75cm]{*->}(0,2)(6,6)
8 \pnode(3,5){A}\pnode(5,0){B}
9 \nczigzag[bow=2cm,
10  coilheight=0.3,](A){B}
11 \end{pspicture}

```

1.7 ppoints

By default a line or node connection divides the total length into 360 segments. This should be increased for long distances or a large number of periods.



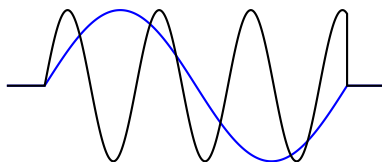
```

1 \begin{pspicture}(5,2)
2 \pssin[ppoints=2000,
3   periods=30](0,1)(5,1)
4 \end{pspicture}

```

1.8 periods

The line from A to B is plotted by default as one period, which is the same as the relative periods=1. A setting of periods=3.3 plots the function 3.3 times for the given length.

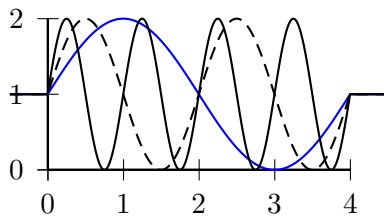


```

1 \begin{pspicture}(5,2)
2 \pssin[linecolor=blue](0,1)(5,1)
3 \pssin[periods=3.3](0,1)(5,1)
4 \end{pspicture}

```

A value without an unit for the period is always relative, with an unit it will be an absolute value.

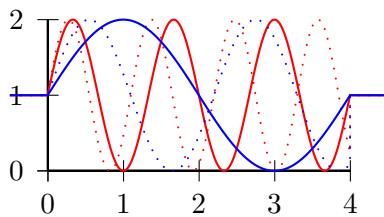


```

1 \begin{pspicture}(5,2)
2 \psaxes(5mm,0)(4.5,2)
3 \psset{coilarm=5mm}
4 \pssin[linecolor=blue](0,1)(5,1)
5 \pssin[periods=1cm](0,1)(5,1)
6 \pssin[periods=2cm,linestyle=dashed](0,1)(5,1)
7 \end{pspicture}

```

With a star as prefix the absolute real value will be truncated to get an integer number of periods.

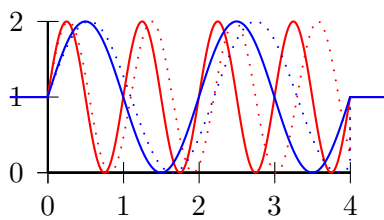


```

1 \begin{pspicture}(5,2)
2 \psset{coilarm=5mm}
3 \psaxes(5mm,0)(4.5,2)
4 \pssin[periods=1.1cm,linestyle=dotted,
5   linecolor=red](0,1)(5,1)
6 \pssin[periods=*1.1cm,linecolor=red](0,1)(5,1)
7 \pssin[periods=2.2cm,linestyle=dotted,
8   linecolor=blue](0,1)(5,1)
9 \pssin[periods=*2.2cm,linecolor=blue](0,1)(5,1)
10 \end{pspicture}

```

With a double star as prefix the absolute real value will be rounded instead of truncated to get an integer number of periods.



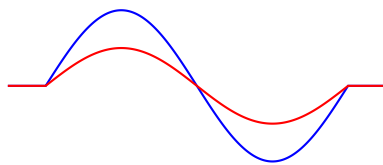
```

1 \begin{pspicture}(5,2)
2 \psset{coilarm=5mm}
3 \psaxes(5mm,0)(4.5,2)
4 \pssin[periods=1.1cm,linestyle=dotted,
5   linecolor=red](0,1)(5,1)
6 \pssin[periods>**1.1cm,linecolor=red](0,1)(5,1)
7 \pssin[periods=2.2cm,linestyle=dotted,
8   linecolor=blue](0,1)(5,1)
9 \pssin[periods>**2.2cm,linecolor=blue](0,1)
10 (5,1)
\end{pspicture}

```

1.9 amplitude

As usual the amplitude is the maximum of the plotted curve and preset to `\psyunit`, eg 1cm.



```

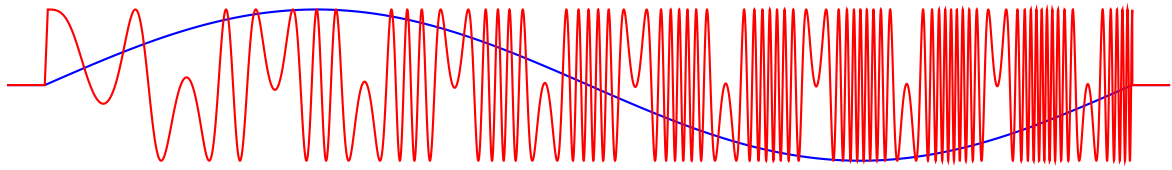
1 \begin{pspicture}(5,2)
2 \pssin[linecolor=blue](0,1)(5,1)
3 \pssin[amplitude=0.5,
4   linecolor=red](0,1)(5,1)
5 \end{pspicture}

```

1.10 function

By default the curve is plotted as $y = \sin(x)$. The optional argument `function` allows to define any other object in PostScript notation. The value of the variable x is already

on the stack! The following example plots the default curve and one with the function $y = \cos(x \cdot \sin(x))$:



```

1 \begin{pspicture}(\linewidth,2)
2 \psysin[linecolor=blue](0,1)(\linewidth,1)
3 \psysin[function=dup sin mul cos,
4   linecolor=red,ppoints=10000,
5   periods=6](0,1)(\linewidth,1)
6 \end{pspicture}

```

2 Makros

ps-coil defines three macros, which make the creation of coil or zigzag shaped lines possible.

```

\pscoil * [Options] {arrows} (x0, y0) (x1, y1)
\psCoil * [Options] {angle1}{angle2}
\pszigzag * [Options] {arrows} (x0, y0) (x1, y1)
\psysin * [Options] {arrows} (x0, y0) (x1, y1)

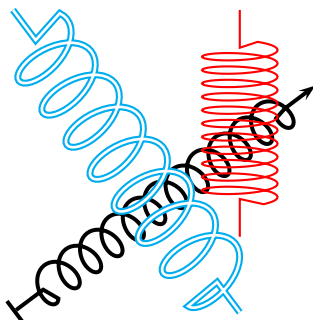
```

- If only one coordinate pair is specified, the first point is automatically set to the origin of ordinates (0,0).
- Arrows may be set with the own parameter or with the optional parameter with arrows=<...-...>.
- For \psCoil the coils are principally drawn without “arms”.

The asterisk version are in principle pointless, since they do not allow a meaningful representation. Some could be seen with the description of parameters, but will not be described further in the following.

2.1 \pscoil

Since \pscoil is nothing else than a polygon line, also all parameters are available which apply for lines.



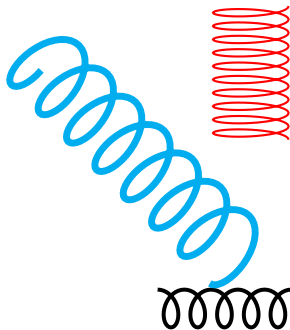
```

1 \begin{pspicture}(4,4)
2 \pscoil[coilarm=.5cm,linewidth=1.5pt,coilwidth=.5cm]
3   [|->](4,3)
4 \pscoil[linecolor=red,coilheight=0.25](3,4)(3,1)
5 \pscoil[doubleline=true,linecolor=cyan,coilheight=0.75](0,4)(3,0)
6 \end{pspicture}

```

2.2 \psCoil

\psCoil draws starting at the current point to the angle α an invisible line and then from α to β a helix. \rput has to be used if they are to be placed at a particular position. The advantage is that with \psCoil the number of helices to draw can be specified exactly.



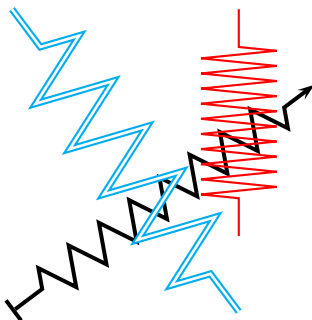
```

1 \begin{pspicture}(4,4)
2 \psCoil[linewidth=1.5pt,coilwidth=.5cm]{1800}{3600}
3 \rput{-90}(3,4){\psCoil[linecolor=red,coilheight
  =0.25]{0}{3600}}
4 \rput{-45}(0,3.5){\psCoil[doubleline=true,linecolor=
  cyan,coilheight=0.75]{100}{2700}}
5 \end{pspicture}

```

2.3 \pszigzag

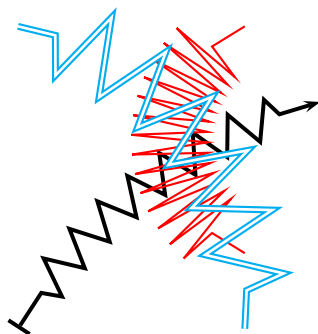
This macro represents the two dimensional variant and is correspondingly easy to use. Especially the lineararc option can bring good results for \pszigzag. The specification of the “arm length” is not absolute here, since the line is principally ended in the geometrical middle and then led to the specified arm length.



```

1 \begin{pspicture}(4,4)
2 \pszigzag[coilarm=.5cm,linewidth=1.5pt,coilwidth=.5cm
  ]{|->}(4,3)
3 \pszigzag[linecolor=red,coilheight=0.2](3,4)(3,1)
4 \pszigzag[doubleline,linecolor=cyan,coilheight
  =0.75](0,4)(3,0)
5 \end{pspicture}

```



```

1 \begin{pspicture}(4,4)
2 \pszigzag[coilarm=.5cm,
  linewidth=1.5pt,bow=5mm,
  coilwidth=.5cm]{|->}(4,3)
3 \pszigzag[linecolor=red,bow=-1cm,
  coilheight=0.2](3,4)(3,1)
4 \pszigzag[doubleline,
  linecolor=cyan,bow=1cm,
  coilheight=0.75](0,4)(3,0)
5 \end{pspicture}

```

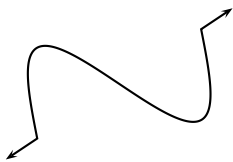
3 Node connections

Prerequisite for this macro is that pst-node is loaded, then the following node connections are available:

```

\nccoil* [Options] {arrows} {nodeA}{nodeB}
\nczigzag* [Options] {arrows} {nodeA}{nodeB}
\pccoil* [Options] {arrows} {nodeA}{nodeB}
\pczigzag* [Options] {arrows} {nodeA}{nodeB}
\ncsin* [Options] {arrows} {nodeA}{nodeB}
\pcsin* [Options] {arrows} {nodeA}{nodeB}

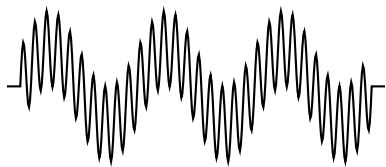
```



```

1 \begin{pspicture}(4,3)
2 \pnode(0.5,0.5){A}
3 \pnode(3.5,2.5){B}
4 \ncsin{<->}{A}{B}
5 \end{pspicture}

```



```

1 \begin{pspicture}(5,2)
2 \pnode(0,1){A}
3 \pnode(5,1){B}
4 \ncsin[amplitude=0.5,periods=30,coilarm=5pt,
5   function=dup 0.1 mul sin exch sin add]{A}{B}
6 \end{pspicture}

```

The other connections behave completely analogue to those dealt with the package `pst-node` so that no further explanations are needed here.

4 List of all optional arguments for `pst-coil`

| Key | Type | Default |
|-------------------------|----------|---------|
| <code>coilwidth</code> | ordinary | 1cm |
| <code>coilheight</code> | ordinary | 1 |
| <code>coilarmA</code> | ordinary | 0.5cm |
| <code>coilarmB</code> | ordinary | 0.5cm |
| <code>coilarm</code> | ordinary | 0.5cm |
| <code>coilaspect</code> | ordinary | 45 |
| <code>coilinc</code> | ordinary | 10 |
| <code>bow</code> | ordinary | 0 |
| <code>periods</code> | ordinary | 1 |
| <code>amplitude</code> | ordinary | 1 |
| <code>ppoints</code> | ordinary | 360 |
| <code>function</code> | ordinary | sin |

References

- [1] Denis Girou. Présentation de PSTricks. *Cahier GUTenberg*, 16:21–70, April 1994.
- [2] Michel Goosens, Frank Mittelbach, Sebastian Rahtz, Denis Roegel, and Herbert Voß. *The L^AT_EX Graphics Companion*. Addison-Wesley Publishing Company, Reading, Mass., 2007.
- [3] Nikolai G. Kollock. *PostScript richtig eingesetzt: vom Konzept zum praktischen Einsatz*. IWT, Vaterstetten, 1989.
- [4] Herbert Voß. *PSTricks – Grafik für T_EX und L^AT_EX*. DANTE – Lehmanns, Heidelberg/Hamburg, third edition, 2006.
- [5] Herbert Voß. *PSTricks – Graphics and PostScript for L^AT_EX and T_EX*. UIT, Cambridge/UK, 2011.
- [6] Timothy Van Zandt. *PSTricks - PostScript macros for generic T_EX*. <http://www.tug.org/application/PSTricks>, 1993.
- [7] Timothy Van Zandt. *multido.tex - a loop macro, that supports fixed-point addition*. CTAN:/graphics/pstricks/generic/multido.tex, 1997.
- [8] Timothy Van Zandt. *pst-coil: Coils and zigzags*. CTAN:/graphics/pstricks/generic/, 1999.
- [9] Timothy Van Zandt and Denis Girou. Inside PSTricks. *TUGboat*, 15:239–246, September 1994.

Index

- amplitude, 4, 9
- arrows, 10
- bow, 4, 8
- coil, 4
- coil, 7
- coilarm, 4, 6
- coilarmA, 4, 6
- coilarmB, 4, 6
- coilaspect, 4, 7
- coilheight, 4, 5
- coilinc, 4, 7
- coilwidth, 4, 5
- diameter, 4
- function, 4, 9
- Keyword
 - amplitude, 4, 9
 - arrows, 10
 - bow, 4, 8
 - coil, 7
 - coilarm, 4, 6
 - coilarmA, 4, 6
 - coilarmB, 4, 6
 - coilaspect, 4, 7
 - coilheight, 4, 5
 - coilinc, 4, 7
 - coilwidth, 4, 5
 - function, 4, 9
 - lineararc, 11
 - periods, 4, 8
 - ppoints, 4
- lineararc, 11
- lineto, 7
- Macro
 - `\nccoil*`, 12
 - `\ncsin*`, 12
 - `\nczigzag*`, 12
 - `\pccoil*`, 12
 - `\pcsin*`, 12
 - `\pczigzag*`, 12
 - `\psCoil*`, 10
 - `\psCoil`, 6, 10, 11
 - `\pscoil*`, 10
 - `\pscoil`, 10
 - `\pssin*`, 10
 - `\psyunit`, 9
 - `\pszigzag*`, 10
 - `\pszigzag`, 11
 - `\rput`, 11
- Package
 - pst-coil, 4, 10
 - pst-node, 11, 12
- `\pccoil*`, 12
- `\pcsin*`, 12
- `\pczigzag*`, 12
- periods, 4, 8
- PostScript
 - lineto, 7
- ppoints, 4
- `\psCoil`, 6, 10, 11
- `\pscoil`, 10
- `\psCoil*`, 10
- `\pscoil*`, 10
- `\pssin*`, 10
- pst-coil, 4, 10
- pst-node, 11, 12
- `\psyunit`, 9
- `\pszigzag`, 11
- `\pszigzag*`, 10
- `\rput`, 11
- three dimensional representation, 6
- winding, 5
- zigzag line, 4, 10